ISSN: 0798-1015

REVISTA ESPACIOS

Vol. 41 (27) 2020 • Art. 16

Recibido/Received: 23/04/2020 • **Aprobado**/Approved: 30/06/2020 • **Publicado**/Published: 23/07/2020

Constructive matheuristic algorithms for solving the multi-depot vehicle scheduling problem for public transportation

Algoritmos mateheurísticos para solucionar el problema de programación de vehículos multidepósito para transporte público

MARÍN, Cesar A [1]
BOLAÑOS, Rubén I [2]
ESCOBAR, Luis M [3]
ESCOBAR, John W. [4]

Abstract
This paper considers the Vehicle Scheduling Problem of public transportation by considering Multi-depots (MDVSP). We propose three hybrid constructive algorithms combining heuristic and exact methods. The proposed approaches are validated by using 90 benchmark instances, having between two and five depots, and between 100 and 500 trips. Also, the efficiency of the algorithms has tested on real instances obtained from the Mass Transit System of the Centro Occidente de Centro Metropolitana de Colombia (AMCO), whose operation consists of about 5000 trips daily.
key words: multi depot vehicle scheduling problem, matheuristic algorithm, public transportation.

Resumen
Este documento considera el problema de programación de vehículos del transporte público al considerar los depósitos múltiples (MDVSP). Proponemos tres algoritmos híbridos constructivos que combinan métodos heurísticos y exactos. Los enfoques propuestos se validan mediante el uso de 90 instancias de referencia, que tienen entre dos y cinco depósitos, y entre 100 y 500 viajes. Además, la eficacia de los algoritmos se ha probado en instancias reales obtenidas del Sistema de Tránsito Masivo del Centro Occidente de Centro Metropolitano de Colombia (AMCO), cuya operación consiste en aproximadamente 5000 viajes diarios.
Palabras clave: problema de programación de vehículos de depósito múltiple, algoritmo matemático, transporte público

1. Introduction

The rate at which economic growth and human development are transforming cities, significant changes are required to be at the forefront of a globalized economy. The obstacles arising from industrialization are increased individual motorization and per-capita travel, increased traffic congestion, inequality, and social segregation (Escobar, 2009; Escobar et al. 2012). The process of climate change aggravates these problems, by the air

[1] R+D+i, Integra S.A., Universidad Tecnológica de Pereira cmarin@integra.com.co
[2] R+D+i, Integra S.A., Universidad Tecnológica de Pereira ribolanos@utp.edu.co
[3] R+D+i, Integra S.A., Universidad Libre Seccional Pereira luism.escobarf@unilibre.edu.co
[4] Associate Professor. Department of Accounting and Finance, Universidad del Valle, Cali. john.wilmer.escobar@correounivalle.edu.co

pollution and deaths in traffic accidents; thus, different transport models are required to supplement the mobilization of people (Escobar et al. 2013; Escobar et al., 2014a; Escobar et al., 2014b).

The complications arising from mobility could be even more significant, according to recent data obtained from ONU (2016). 95 % of urban expansion in the next decades will occur in the developing world. This fact leads to an increase in people who need to mobilize; thus, from today, cities are forced to think of plans to mitigate these problems. One of the issues is the restructuring of public transport systems, in such a way that it is possible to move people quickly, practical and economical.

This fact has led to the establishment of different transport models to meet the needs of mobilization of people worldwide; a clear example is the implementation of mass transport models types BRT (Bus Rapid Transit), as a measure to reduce vehicular congestion and improve transport conditions. Currently, 165 cities in the world have Passenger Public Transportation Systems, which transport more than 32 million passengers per day on a road infrastructure of 4.862 kilometers (BRTData, 2017).

Despite the restructuring of public passenger transport performed during the last decades and the implementation of public transport policies in each country, the reality of mass transport operators shows that not all cases have been the success from the economic and social point of view. This fact due that the enormous efforts of the states in favor of the implementation of Integrated Systems of Massive Transport (SITM), at present this type of systems is having significant problems of sustainability, since that there are different unconventional and, in some cases, illegal ways of mobilizing, reducing the use of public transport in BRT. Therefore, this model of transportation becomes unviable, arising as main challenge the improvement of the efficiency of the public transportation system, which is not only the daily operation and complies with the travels planned, but also, is the implementation of administrative strategies based on technical concepts and applied research that helps reduce operating costs.

From the technical point of view, the operational planning of public passenger transport systems covers different aspects such as the scheduling of work shifts for bus operators, the schedule of preventive maintenance work and the scheduling of the buses needed to carry out the trips. The scheduling of buses is stipulated in the tactical planning of the system, as well as the assignment of the personnel to each work shift recently involved at this stage, and the control in real-time of the system fleet. Each of the problems mentioned above has been widely studied in the specialized literature and due to its mathematical and computational complexity are classified as NP-hard type problems, which has led to each of them, be solved sequentially by generally using approximate approaches.

The problem of scheduling of vehicles with multiple depots (MDVSP), considers the determination of a set of vehicles that must carry out a set of trips of a set of routes with a given frequency at each moment of the day. The reality of public passenger transport companies makes the MDVSP problem of great importance, and are the source of motivation for this research, considering new variants fitting the particular environment of each company dedicated to the operation of public passenger transport services. However, regardless of the variables present in each reality, the objective will always be framed in the total fulfillment of the itineraries and the reduction of costs related to the operation of the system through optimization processes. Each plan is a description of the trips that must be executed in a specific time and sections called routes, obeying a frequency according to the conditions of the service and the public service needs of the mass transport determined by the tactical planning defined by the managing entity of the SITM. Thus, the combination of route and time of departure is called service, and a group of services of the same section is defined as a table. The routes of public transportation systems are identified from their strategic planning without any substantial changes in the short or medium term.

Note each route must be served with a specific frequency, at a given average speed, defined in the tactical planning of the transport systems. Indeed, all these requirements of the routes are determined from the design of the service network (network route design), and they are performed precisely to meet the needs identified for the strategic planning.

Colombia, in particular, has been in the process of restructuring its public transportation system since 1993 through the development of plans and strategies for the country to use the SITM concept. Features include reliability, efficiency, increased quantity of buses, and greater coverage in population areas, among others.

The system used for mass transportation in Colombia and Latin America has been the Integrated Mass Transit System of the Third Millennium "Transmilenio." It was Inaugurated in the year 2000, and its policy is outlined in Document CONPES 2999, System of Urban Public Service of Mass Passenger Transportation for the city of Santa Fe de Bogotá, Colombia. This policy led to an environment in which a regulatory framework for public transportation was required, resulting in CONPES 3167, in which the National Planning Department established the Policy to improve the Public Transport Service Urban Passenger. In 2003, the CONPES 3260 established the National Policy for Urban and Mass Transportation.

However, the reality of the companies operating Mass Transit in Colombia reflects that not all enterprises are undertaking the execution of this policy to be successful from the economic and social point of view. Despite the enormous efforts of the Colombian government for the implementation of SITM, these enterprises are having major sustainability problems. The different subcontracting enterprises have competition from unconventional and in some cases, illegal methods of transportation. Integra S.A., which is the Mass transportation operator of the bus system for the Downtown West Metropolitan of Colombia, has implemented strategies that include the incorporation of advanced models and techniques to improve the efficiency and sustainability of the system. These approaches consider the implementation of hardware and software that allow an optimal operation and guarantees conditions of accessibility, comfort, and efficiency to the customer.

This paper proposes three matheuristic constructive algorithms to solve the MDVSP. The first algorithm considers the assignment of trips taking into account their chronological order and their cost from the deposit. The second algorithm deals with the attention to the sequence with all the services combining the chronological order and the nearest neighbor for each of the trips. Finally, the third algorithm contemplates a graph theory to construct minimum cost itineraries by using a particular formulation of the minimum flow of a network.

The paper is organized as follows. Section 2 reviews the literature related to the MDVSP. Section 3 proposes a mathematical formulation of the problem, while section 4 describes the proposed algorithms. Finally, in sections 5 and 6, the computational results and conclusions are shown, respectively.

2. Literature Review

The MDVSP is a well-known problem seeking the determination of the best schedules for vehicles assigned to several depots by considering that each task is performed exactly once by a vehicle. An optimal plan is characterized by minimal fleet size and minimal operational costs. An extensive review of vehicle scheduling problems has been proposed by Bunte and Kliewer (2009).

Pepin et al. (2009) propose five different heuristics for solving the MDVSP. Some of them are adaptions of existing methods, while two are novel heuristics proposed for the considered problem. In the review of the state of the art of scheduling of vehicles, it was identified that there is different research applied to the improvement of the operation of the public transport system of passengers by optimizing the MDVSP Problem. All these approaches have been of sufficient importance for companies due to their results contribute to the development of an efficient transport system capable of meeting the mobility needs to be required in cities (Ibarra-Rojas et al., 2015; Muñoz and Paget-Seekins, 2016).

A dynamic model is introduced by (Huisman et al., 2004) to solve the problem of vehicle scheduling (VS). This approach attempts to explain a set of optimization problems in a sequential way. It takes into account different scenarios in future travel times. The first phase initially assigns trips to the various depots (clustering). The second stage solves a simple problem of dynamically scheduling vehicles.

Gintner et al. (2005) propose MDVSP with multiple vehicles types. This two-phase method provides results very close to optimal solutions. The mathematical formulation of the problem is based on a space-time network. A vehicle is allowed to return to a different depot, which seeks to minimize empty travel times and downtime. In reality, the number of trips exceeds one thousand, which is why the authors combine the model of a space-time network with a heuristic approach to solve significant problems and to be able to add new practical considerations.

Hadjar et al. (2006) propose a Branch and Bound Algorithm to solve the MDVSP. This model combines the generation of Columns (CG), Fixed Variables, and Cutting Plans. The authors review two mathematical formulations based on CG schemes to solve the Lagrange relaxation of the linear programming problem. The algorithm is validated in randomly generated instances, case studies, and a set of real data from the Montreal Transport Society (STM). The STM operates a network that includes seven depots, 665 bus lines with 380 completion points, and 17,037 trips.

A review of the literature reveals several methodologies applied to the Vehicle Scheduling Problem (VSP) in academic test cases. These methods are less successful than real facts from the computational point of view; since in practical situations, the quantity of services grows considerably compared to test instances. This fact makes solving these problems with exact methods cumbersome. The efficiency of assignment of trips is paramount as vehicles constitute the highest costs within the budget of the operation of public transport systems (Ceder, 2007), either by their acquisition or by use.

Wang and Shen (2007) propose a new version of the problem of scheduling of vehicles VSP called VSPRFTC, examining electric buses. This approach considers two new constraints related to the length of route and vehicle recharge time. The authors propose a new mathematical model and Ant Colony algorithm to solve large instances.

A new neighborhood scheme called block moves (Laurent and Ha, 2009), suggests an iterative local search algorithm (ILS) to solve the MDVSP. The methodology uses an efficient auction algorithm to generate the initial vehicle schedule. The algorithm then integrates a two-step perturbation mechanism, which allows a search with controlled diversification. The methodology was validated in a set of 30 instances of the MDVSP from the literature.

Shui et al. (2015) present a new VSP approach based on a cloning algorithm, which achieves good quality solutions efficiently. This new method can also solve problems of large-scale vehicle scheduling, for which two heuristics are applied. It allows the readjustment of departure times of each trip to improve the solutions found in previous procedures. The methodology is validated in the programming of vehicles of the bus company of Nanjing China, finding satisfactory solutions in less than a minute.

A heuristic framework that combines a space-time network is proposed by Guedes and Borenstein, (2015). This approach addresses the problem of scheduling vehicles with multiple depots and a mixed fleet (MDVTSP). Using truncated column generation and reduction of State-space solves the problem of scheduling for large-scale

MDVTSP. The development of the algorithms is measured by using randomly generated episodes of up to 3000 trips, 32 depots, and eight types of vehicles. The results obtained are promising and constitute a viable alternative to solve MDVTSP efficiently.

Hassold and Ceder (2014) use a methodology based on a low-cost network flow model for the problem of scheduling vehicles with mixed fleets (MVT-VSP). The method uses a set of timetables organized on an optimal Pareto front for each bus line and allows for the stipulation of a particular type of vehicle for a trip and in turn, allows replacement of the vehicles. The authors apply this methodology in New Zealand, and the results show an improvement of 15%, regarding the cost of the fleet of vehicles.

Kliever et al. (2006) discuss the multi-depot, multi-vehicle-type bus scheduling problem (MDVSP). A time–space-based network formulation is used for modeling MDVSP. This formulation allows a reduction of the size of the problem in comparison with other formulations. A new formulation for the MDVSP using assignment arcs in a multi-commodity time-space network flow is proposed by Kulkarni et al. (2018). A Dantzig–Wolfe decomposition to the formulation by decomposing it for each trip, is applied. Besides, three different heuristics are proposed based on the solution framework. The computational experiments show that the former algorithms provide better quality solutions than the existing heuristics.

A two-phase fast heuristic approach for the MDVSP is introduced by Guedes et al. (2016). The first phase applies two state-space procedures reducing the complexity of the problem. Then, in the second phase, the reduced problem is solved by a truncated column generation approach. The performance of the former algorithm has been tested on a series of benchmark problems. A local search approach using pruning and deepening techniques in a variable depth search framework for the MDVSP is proposed by Otsuki and Aihara, (2016).

Different variants of the MDVSP have been proposed by Desaulniers et al. (1998), Semedo et al. (2015), Uçar et al. (2017), Guedes and Borenstein (2018) and Xu et al. (2018). Desaulniers et al. (1998) consider the MDVSP with time windows called MDVSPTW. In particular, each task is restricted to begin within a prescribed time interval, and different depots supply vehicles. A nonlinear model has been proposed by considering costs on exact waiting times. The Multi-Depot Vehicle Scheduling Problem with Line Ex-changes is introduced by Semedo et al. (2015). A parallel Ant-Colony Optimization (ACO) metaheuristic has been proposed to solve the considered problem. Uçar et al. (2017) discuss two disruptions for the MDVSP (delays and extra trips). The mathematical model for the considered problem includes robustness aspects such as Polo et al. (2018). Exact column and row generation algorithm has been proposed to validate a lower bound. Guedes and Borenstein (2018) discuss the multiple-depot vehicle type rescheduling problem (MDVTRSP), which is a dynamic extension of the MDVSP. A new formulation and a heuristic solution approach for the MDVTRSP have been proposed. Computational experiments on randomly generated instances were performed to evaluate the performance of the former algorithms. Finally, Xu et al. (2018) suggested a model and algorithm for the MDVSP with departure-duration constraints. The former approach is applied to a real-life case in China and several test instances.

3. Methodology

3.1. Mathematical Formulation

According to the mathematical formulation presented by Fischetti et al. (1999), the MDVSP considers a set of n trips $T = \{T_1, T_2, \dots, T_n\}$, where each trip T_j $(j = 1, 2, \dots, n)$ has a starting time s_j and an ending time e_j, a set of m depots $D = \{D_1, D_2, \dots, D_m\}$, where each depot has $r_k \leq n$ homogeneous available vehicles and it is assumed that $m \leq n$.

Consider τ_{ij} as the time that a vehicle needs in order to travel from the location where the trip i ends, to the location where the trip j starts, this way, a pair of consecutive trips $(T_{i;}T_j)$ are feasible if the same vehicle can service the trip T_j immediately after completing the trip T_i, implying the fulfillment of the condition in (1).

$$e_i + \tau_{ij} \leq s_j \tag{1}$$

For each pair of feasible trips, a cost $\gamma_{ij} \geq 0$ is associated; also, for each infeasible pair and for $i = j$, a cost $\gamma_{ij} = +\infty$ is associated. For each trip T_j and each depot D_k exist a non-negative cost $\bar{\gamma}_{kj}$ when a vehicle starts its itinerary with the service T_j from the depot D_k (in the same way, there is a cost $\bar{\gamma}_{jk}$ when a vehicle finishes its itinerary with the trip T_j from the depot D_k). Indeed, the total cost of an itinerary $(T_{i_1}, T_{i_2}, \ldots, T_{i_h})$ associated with a vehicle from the depot D_k is calculated like the following expression $\bar{\gamma}_{ki_1} + \gamma_{i_1 i_2} + \cdots + \gamma_{i_{h-1} i_h} + \bar{\gamma}_{i_h k}$.

Consider a graph $G = (V, A)$, where the set of nodes $V = \{1, \ldots, n + m\}$ is divided into two subsets, $W = \{1, \ldots, m\}$ which contains a node k for each depot D_k and $N = \{m + 1, \ldots, m + n\}$ which is associated to each node $m + j$ to a different trip T_j, to simplify the notation without loss generalization G is considered a complete graph, where the set of edges is given by $= \{(i, j): i, j \in V\}$. Therefore, the associated costs for each edge (i, j) is defined by (2).

$$c_{ij} = \begin{cases} \gamma_{i-m,j-m}; & \forall\, i, j \in N. \\ \bar{\gamma}_{i,j-m}; & \forall\, i \in W, \forall j \in N. \\ \bar{\gamma}_{i,j-m}; & \forall i \in N, \forall j \in W. \\ 0; & \forall\, i, j \in W, i = j. \\ +\infty; & \forall\, i, j \in W, i \neq j. \end{cases} \tag{2}$$

The MDVSP could be defined as the problem of finding the minimum number of subtours with minimum cost by the linear programming problem proposed by Dell'Amico et al. (1993) described by (3)-(7).

$$MDVSP = min \textstyle\sum_{i \in V} \sum_{j \in V} c_{ij} x_{ij} \tag{3}$$

$$\sum_{i \in V} x_{ij} = r_j, j \in V \tag{4}$$

$$\sum_{j \in V} x_{ij} = r_i, i \in V \tag{5}$$

$$\sum_{(i,j) \in P} x_{ij} \leq |P| - 1, P \in \Pi \tag{6}$$

$$x_{ij} \geq 0 \ \ integer, j \in V \tag{7}$$

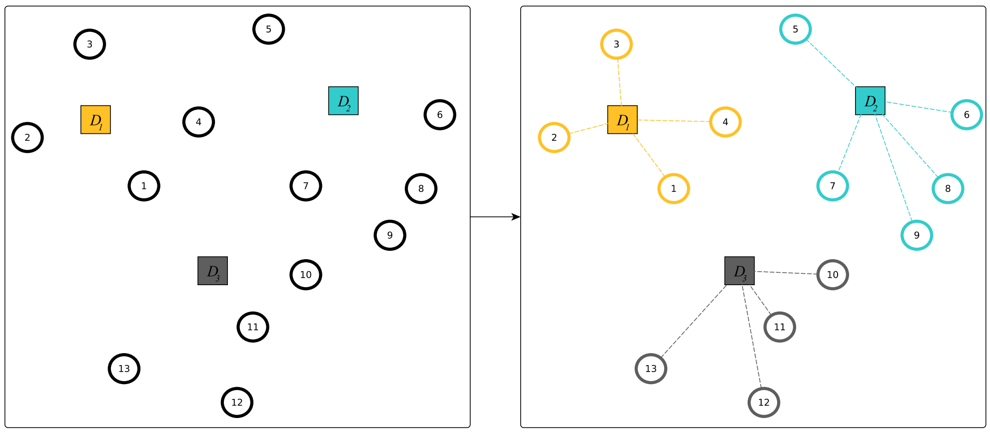
Equations (3) correspond to the objective function considering the cost of the itineraries (selected edges) into a solution. Constraints (4) and (5) impose that each node (trip) $k \in V$ must be visited (serviced) exactly r_k times. For this problem the node must be visited only once. Equations (6) forbid the generation of infeasible subtours, i.e., subtours presenting more than one node from the set W (nodes representing depots).

3.2 Proposed Methods

3.2.1. Concurrent Clustered Scheduler (CSC)

This procedure proposes the application of the well-lnown Concurrent Scheduler method by adding a first stage of clustering. The algorithm starts by determining from which depot D_k each of the trips must be served. Each trip T_j is allocated by a heuristic way taking into account the lowest value $\bar{\gamma}_{kj}$, as illustrated in Figure 1.

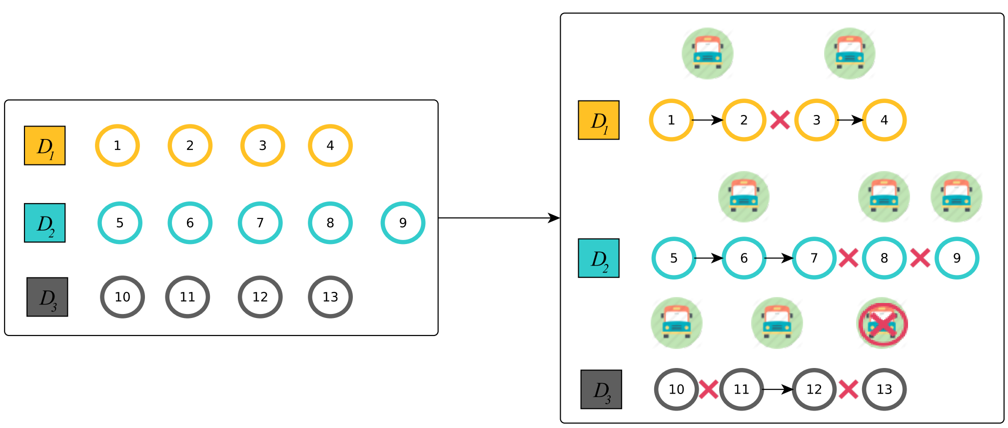
Figure 1
Clustering of trips for each depot $\boldsymbol{D_k}$



Source: Owner

Then, the trips are assigned to each depot are sorted chronologically according to the start time s_j. Later, each itinerary is created taking into account the established order and the condition described in (1). When (1) fails, the itinerary is completed and assigned to a different vehicle of the depot D_k. The process continues according to the order of the remaining trips, and is repeated until there are no trips to be assigned in each of the clusters. Figure 2 shows this process for the cluster associated to D_3 in which there is a lack of available vehicles for trip number 13, therefore, a reassignment to the next nearest available cluster is performed.

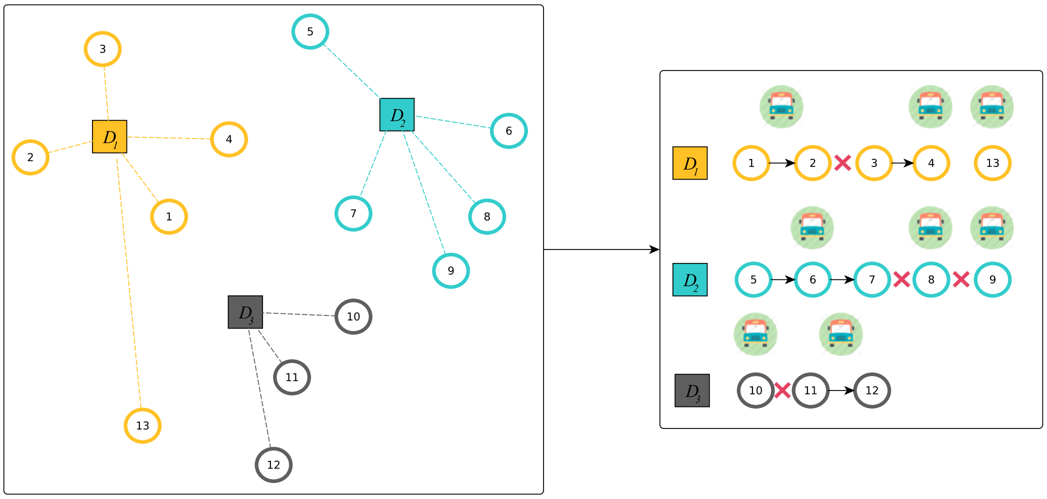
Figure 2
Construction of itineraries for the fleet of each depot $\boldsymbol{D_k}$



Source: Owner

Figure 3 illustrates the reassignment of trip 13 to the depot D_1, generating an itinerary with a single trip or service from this depot. At the end of the process, there is a stage of intensification of the plans that have a single trip assigned, in the attempt to insert them in the existing itineraries or constructing itineraries between them.

Figure 3
Reassignment of trip 13 for fleet availability

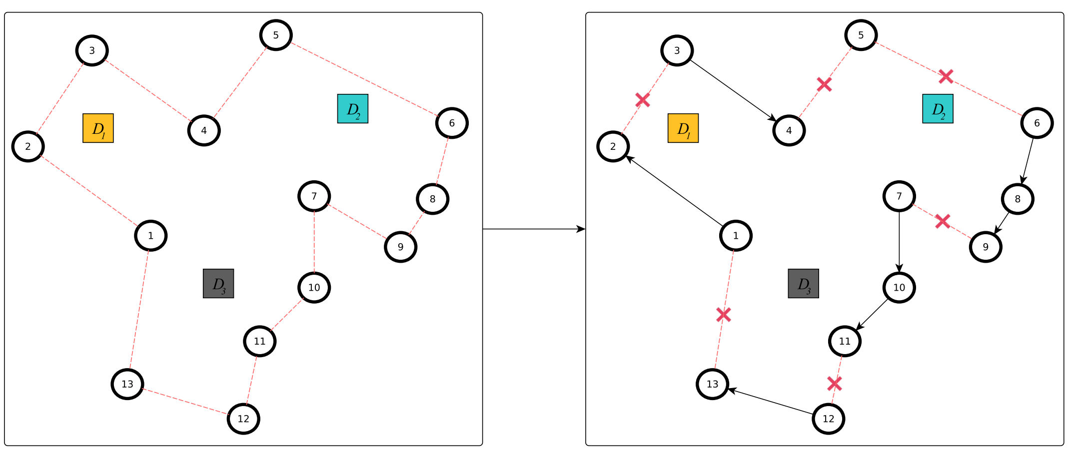


Source: Owner

3.2.2. Minimum Cost Attention Sequence (MCAS)

This method is based on the construction of a general sequence with all the trips T_j. Only the nodes of the set N are taken into account. The first trip in the sequence corresponds to the trip whose start time s_j indicates that it is the first to be performed. The subsequent trips in the sequence are assigned according to the cost of the transition between a pair (i,j) $\gamma_{i-m,j-m}$ of lower value that satisfies (1). When (1) or the minimum grade requirement is not fulfilled, the algorithm finishes and a new itinerary is generated. The construction of the general sequence continues with the following trips that have yet to been assigned by grouping trips based on the highest feasible output (according with 1). The process is repeated until all the trips has been assigned based on the overall sequence. Figure 4 shows how the sequence is constructed. This method attempts to the well-known Traveling Salesman Problem (Lin and Kernighan, 1973), however, the MCAS uses an incomplete version.

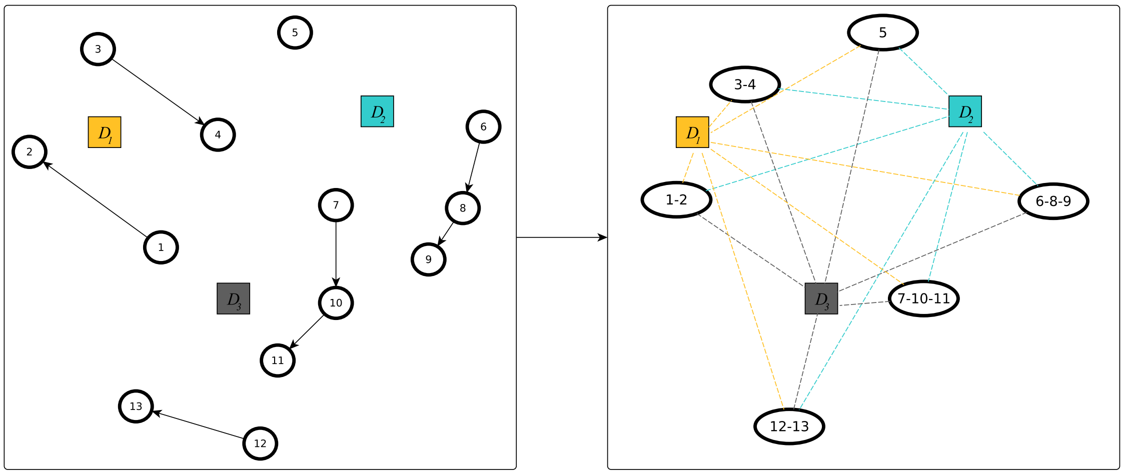
Figure 4
Sequence generated connecting routes
with the criterion of the nearest neighbour



Source: Owner

After finishing the general sequence, some of the itineraries have a single trip. A permutation is then performed to try to insert these trips into another itinerary. This fact allows a savings of a vehicle and thus another vehicle for use on other itineraries. In the second stage of intensification, the pairing of itineraries is sought. The itineraries are ordered ascending according to the start time of the first trip s_j. To pair two routes, equation (1) must be satisfied, taking into account the time of completion e_j of the last trip of itinerary 1 and the start time s_j of the first trip of itinerary 2 and the travel time between these two trips. The resulting itineraries are modelled as super nodes as shown in Figure 5.

Figure 5
Sequence generated connecting routes
with the criterion of the nearest neighbour



Source: Owner

Finally, the assignment of each of the tours to the depots is executed, solving the mathematical model of the generalized allocation problem GAP, which is given by (8)-(11).

$$(GAP) = min \sum_{i \in A^*} \sum_{j \in W} c_{ij} x_{ij} \quad (8)$$

$$\sum_{j \in W} x_{ij} = 1, \forall\, i \in A^* \quad (9)$$

$$\sum_{i \in A^*} a_{ij}\, x_{ij} \leq b_j, \forall\, j \in W \quad (10)$$

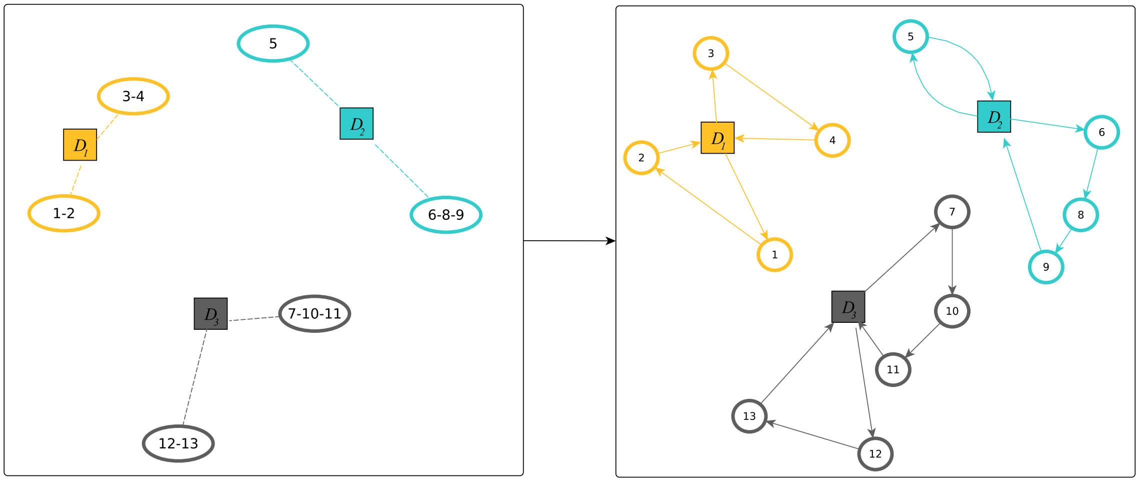
$$x_{ij} \in \{0, 1\}, \forall\, i \in A^*, \forall\, j \in W \quad (11)$$

The equation (objective function) of (8) represents the total cost of assigning an itinerary $\boldsymbol{i}$, to a depot $\boldsymbol{j}$, where $\boldsymbol{A}^*$ is the set of all the itineraries constructed in the first part of the algorithm. The cost $\boldsymbol{c_{ij}}$ is given by the sum of each of the terms of (12). Equations (9) indicates that each itinerary must start from a single depot. Constraints (10) refer to the capacity of each depot in terms of fleet, where $\boldsymbol{a_{ij}}$ is a constant that is equal to 1 and represents the need for a vehicle for each itinerary. $\boldsymbol{b_j}$ represents the capacity of each of the depots. Finally, the expressions (11) correspond to the set of binary variables $\boldsymbol{x_{ij}}$, where is equal 1 if the itinerary $\boldsymbol{i}$ is fulfilled by the depot $\boldsymbol{j}$, otherwise it equals zero.

$$c_{ij} = \begin{cases} \bar{\gamma}_{i,j}\ ; & \forall\ i \in W, \forall j \in A^*. \\ \bar{\gamma}_{j,i}\ ; & \forall i \in A^*, \forall j \in W. \\ 0; & \forall\ i,j \in W, i = j. \\ +\infty; & \forall\ i,j \in W, i \neq j. \end{cases} \quad (12)$$

The resulting allocation of the routes to the various depots will be established by the MDVSP solution as illustrated in Figure 6.

Figure 6
Assignment of resolved supernodes

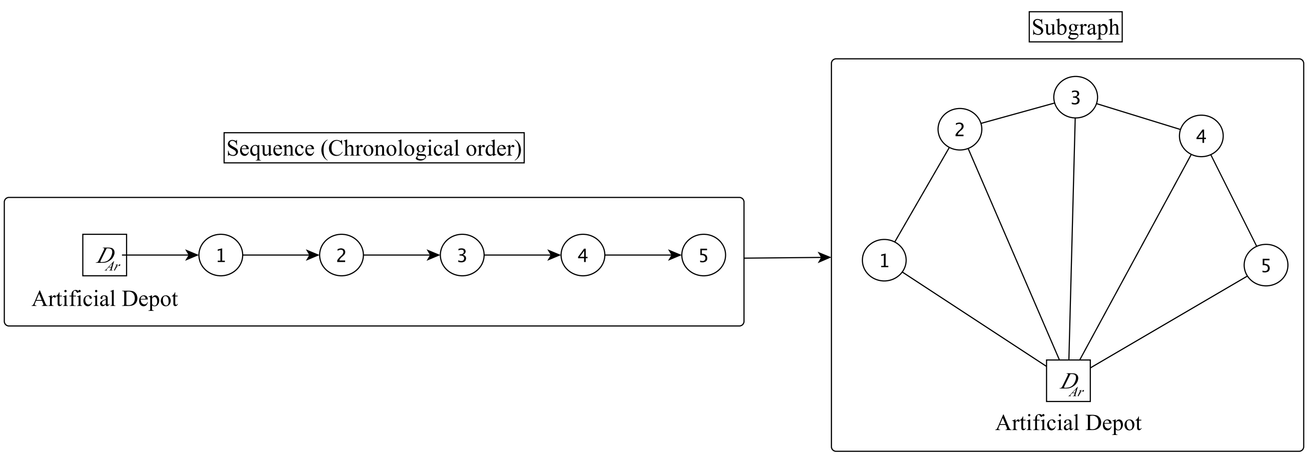


Source: Owner

3.2.3. Division of Attention Sequence (DSA)

This algorithm is an adaptation of the sequence division methodology proposed by Prins (2004). The method starts with a sequence of all trips, whose order is given by the start time s_j of each trip $T_j (j = 1, \dots, n)$.

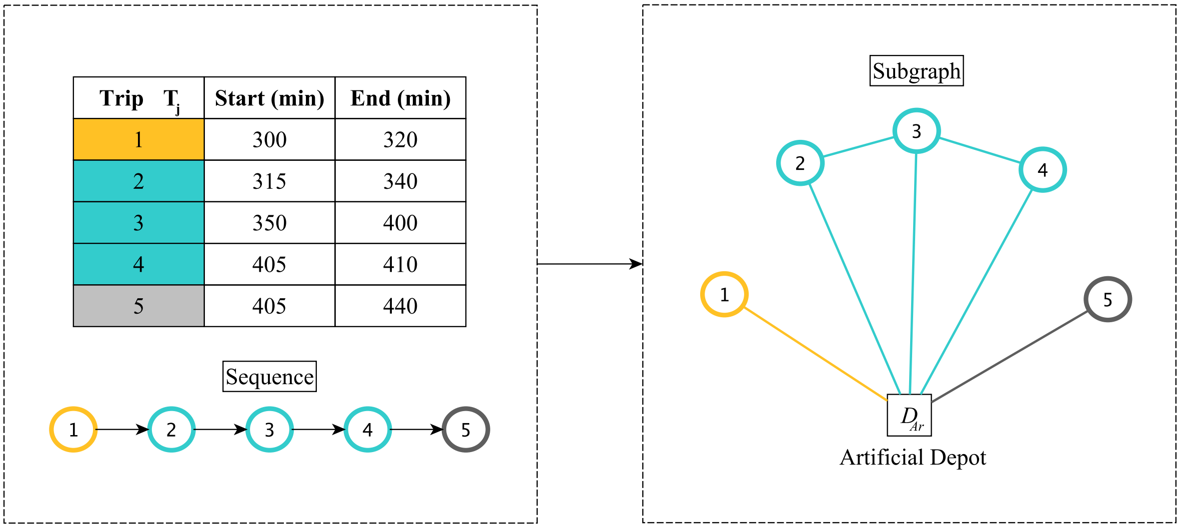
Figure 7
Chronologically ordered sequence (or any combination of criteria) and its corresponding subgraph



Source: Owner

The proposed approach uses a subgraph of the problem instead of the original graph (see Figure 7) in order to reduce considerably the solution space. As each sequence presents infeasibilities, customers must be removed from subgraph due to violations of Equation (1). An illustrative example is given in Figure 8.

Figure 8
Subgraph (or auxiliary graph) after removing infeasibilities

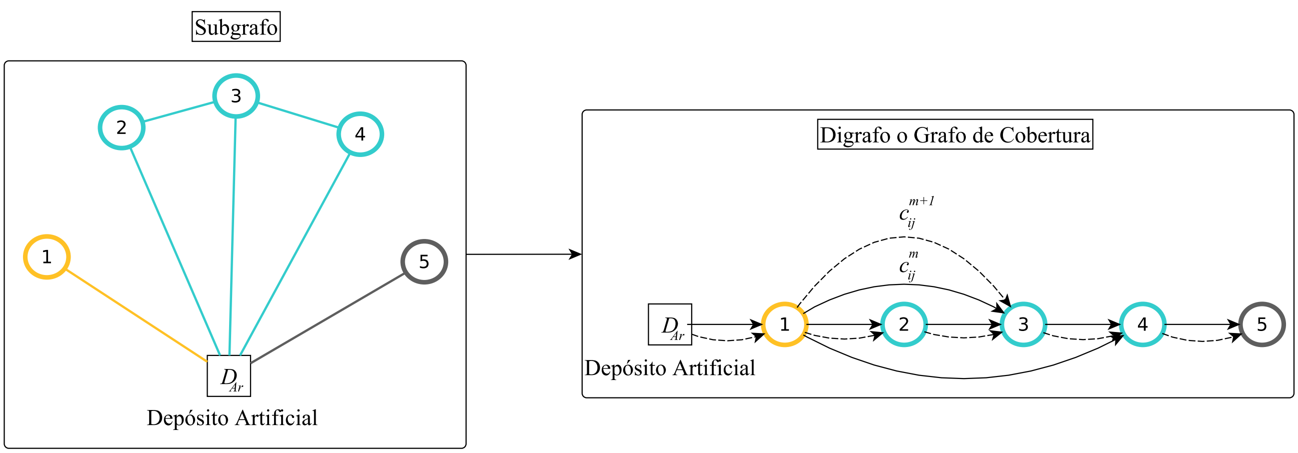


Trip T_j	Start (min)	End (min)
1	300	320
2	315	340
3	350	400
4	405	410
5	405	440

Source: Owner

For an exhaustive exploration of the auxiliary graph, a digraph is constructed with all feasible routes and sub-routes. Each itinerary is repeated as many times as number of depots. Figure 9 illustrates a feasible set of paths, represented in a digraph to model the MDVSP as a minimal flow problem: $\{1\},\{2\},\{3\},\{4\},\{5\},\{2{,}3\},\{2{,}3{,}4\},\{3{,}4\}$

Figure 9
Digraph with all feasible routes and subroutes
for each one of the depots



Source: Owner

The contributions proposed of this algorithm with respect to the methodology presented by Prins (2004), suggests the extension to multiple depots, reflected in the constraints (18). Since the original methodology is proposed for the division of itinerary sequences from a single depot. Additionally, we have extended the use of the proposed approach from vehicle routing problems (VRP) to the MDVSP. The MDVSP presents a more complex problem with respect to the resulting auxiliary graph, since it presents a high amount of infeasibility.

To model the MDVSP as a problem of minimum flows, a digraph $G^* = (V^*, A^*)$, where $G^* = V^*$ represents the set of trips that must be completed and the set of edges, A^* represents all possible combinations of itineraries that result from a determined sequence of trips and each itinerary has a cost c_{ij}^{m}. A binary variable x_{ij}^{m} is defined that takes the value of one if the itinerary $(i,j) \in A^*$ is completed from the depot $m \in W$ and is part of the final solution, otherwise it takes the value of zero. Additionally, the parameters of Equations (13) and (14) are defined.

$$e_r = \begin{cases} -1; & \text{if } r \in V^* \text{ is the initial node of digraph } G^* \\ 0; & \text{if } r \in V^* \text{ is a transit node of digraph } G^* \\ +1; & \text{if } r \in V^* \text{ is the final node of digraph } G^* \end{cases} \tag{13}$$

$$a_{ij}^r = \begin{cases} -1; & \text{if } (i,j) \in A^* \text{ exits of the node } r \in V^* \\ 0; & \text{if } (i,j) \in A^* \text{ does not have realtion with } r \in V^* \\ -1; & \text{if } (i,j) \in A^* \text{ arrives at node } r \in V^* \end{cases} \tag{14}$$

The mathematical model representing the MDVSP as a model of minimum cost flows is given by Equations (15)-(19).

$$Z = min \sum_{m \in W} \sum_{(ij) \in A^*} c_{ij}^m x_{ij}^m \tag{15}$$

$$\textstyle\sum_{m \in W} \sum_{(ij) \in A^*} a_{ij}^r x_{ij}^m = e_r, \forall\, r \in V^* \tag{16}$$

$$\sum_{(ij) \in A^*} x_{ij}^m \leq b_m, \forall\, m \in W \tag{17}$$

$$\textstyle\sum_{m \in W} x_{ij}^m \leq 1, \forall\, (i,j) \in A^* \tag{18}$$

$$x_{ij}^m \in \{0,1\}, \forall\, (i,j) \in A^*, \forall\, m \in W \tag{19}$$

The set of Equations (16) ensures the conservation of flow in both the nodes and the digraph. The constraints (17) control the number of routes served from each depot; the capacity of each depot. The set of inequalities (18) ensure each itinerary must be served from a single depot. Finally, the set of constraints (19) ensure the integrality of the decision variables.

4. Results

The proposed approaches have been coded to expedite minimum computing time, in the actual programming of the company's fleet INTEGRA SA, operator of the AMCO Mass Transportation System that coordinates an average of 5000 trips daily. These methods will be the basis for the development of a population algorithm to find solutions for the MDVSP, taking advantage of the multiple strengths provided by each of the implemented constructive algorithms.

The proposed algorithms have been validated using test cases from the established literature. The results are compared to that of Fischetti et al. (1999). The test case data encompass 120 instances described by two depots and between 100 and 500 trips, three depots and between 100 and 400 trips and those of 5 depots vary between 100 and 300 trips. The algorithms were implemented in C ++ ® under the Linux operating system Ubuntu 14.04 64-bit LTS along with the CPLEX® Version 12.06 64-bit solver, an Intel® Core ™ i7-3630QM CPU @ 2.40GHz × 8 processor and 8 GB.

In Table 1, the first four columns correspond to the reference values regarding the case, solution, number of vehicles used (V) and computational time (t). The last nine columns correspond to the results obtained by the proposed constructions CSC, SAMC, and DSA.

Table 1
Results for the instances with 2 depots.

Instance	(Fischetti, Lodi, & Toth, 1999)			CSC			SAMC			DSA		
	Sol.	V	t(s)	Sol.	V	t(s)	Sol.	V	t(s)	Sol.	V	t(s)
2-100-1	279463	25	0,35	319649	28	0	319288	28	0,015	319288	28	0,108
2-100-2	301808	27	0,38	330841	29	0	334693	29	0,27	334693	29	0,111
2-100-3	341528	31	0,45	373141	33	0	364312	32	0,103	364312	32	0,096
2-100-4	289864	26	0,67	328709	29	0	320422	28	0,095	320422	28	0,141
2-100-5	328815	30	1,45	370867	33	0	380735	34	0,092	380735	34	0,097
2-100-6	360466	33	0,28	391506	35	0	383712	34	0,102	383712	34	0,143
2-100-7	290865	26	0,42	322480	28	0	308985	27	0,067	308985	27	0,154
2-100-8	337923	31	0,5	368594	33	0	359374	32	0,102	359374	32	0,133
2-100-9	270452	24	0,9	342358	30	0	300175	26	0,068	300175	26	0,189
2-100-10	291400	26	0,63	390584	35	0	320275	28	0,067	320275	28	0,096
Average	309258	27	0,6	353873	31	0	339197	30	0,07	339197	29	0,127
2-200-01	545188	49	5,23	618159	54	0	660802	58	0,304	660802	58	0,48
2-200-02	617417	56	13,58	660855	58	0	675250	59	0,2914	675250	59	0,479
2-200-03	666698	61	26,73	702664	62	0	739245	66	0,336	739237	66	0,496
2-200-04	599404	54	4,17	694244	61	0	654025	57	0,266	654025	57	0,513
2-200-05	626991	56	27,73	701208	61	0	704203	61	0,3	704203	61	0,484
2-200-06	592535	54	5,15	676355	60	0	624568	55	0,273	624568	55	0,486
2-200-07	611231	55	77,43	672587	59	0	699273	61	0,317	699273	61	0,506
2-200-08	586297	53	61,02	670180	59	0	648139	57	0,261	648139	57	0,501
2-200-09	596192	54	9,1	627349	55	0	697392	61	0,315	697392	61	0,5
2-200-10	618328	56	2,88	709511	63	0	679830	60	0,312	679830	60	0,508
Average	606028	55	23,3	673311	59	0	678273	59	0,3	678271	59	0,495
2-300-01	907049	83	349,38	1045166	93	0	979298	87	0,581	989962	88	1,253
2-300-02	789658	71	46,3	900715	79	0	906356	79	0,453	916456	80	1,172
2-300-03	813357	74	61,12	913846	81	0	966706	85	0,571	966706	85	1,274
2-300-04	777526	70	51,37	878221	77	0	923910	81	0,482	923910	81	1,222
2-300-05	840724	76	19,25	972836	86	0	942604	82	0,513	942604	82	1,206
2-300-06	828200	75	66,55	916390	80	0	957692	85	0,51	957692	85	1,227
2-300-07	817914	74	30,67	867390	76	0	882812	78	0,457	882812	78	1,196
2-300-08	858820	78	33,02	960522	85	0	967834	85	0,532	967834	85	1,191
2-300-09	902568	82	77,2	936435	82	0	1053320	92	0,604	1053321	92	1,334
2-300-10	797371	72	106,72	890118	78	0	897633	78	0,454	897633	78	1,183
Average	833319	75	84	928164	82	0	947816	83	0,52	949893	83	1,226
2-400-01	1084141	98	431,2	1174619	103	0	1394780	123	1,1	1394781	123	2,346
2-400-02	1028509	93	171,4	1201027	106	0	1204640	105	0,863	1204638	105	2,266
2-400-03	1152954	105	137,8	1268889	112	0	1265370	110	0,91	1265371	110	2,275
2-400-04	1112589	101	412,7	1183893	104	0	1288680	113	0,967	1288678	113	2,365
2-400-05	1141217	104	670,7	1257922	111	0	1213000	106	0,917	1223924	107	2,28
2-400-06	1100988	100	61,57	1261390	112	0	1217380	108	0,822	1217381	108	2,329
2-400-07	1237205	113	398,3	1300285	115	0	1301510	114	1,075	1301508	114	2,483
2-400-08	1111077	101	158,9	1207654	106	0	1273730	113	0,948	1284614	114	2,407
2-400-09	1104559	100	410,7	1216153	107	0	1203330	105	0,828	1203328	105	2,396
2-400-10	1086040	99	125,8	1192299	105	0	1213630	107	0,881	1213627	107	2,352
Average	1115927,9	101	297,9	1226413	108	0	1257605	110	0,93	1259785	110	2,35
2-500-01	1296920	118	1222,1	1455685	128	0	1785500	159	1,84	1785500	159	4,032
2-500-02	1490681	136	2667,5	1628068	144	0	1666580	148	1,645	1677092	149	3,864
2-500-03	1328290	121	854,8	1456659	129	0	1545960	136	1,481	1545959	136	3,686
2-500-04	1373993	125	1351,4	1499704	132	0	1613560	141	1,585	1624500	142	3,733
2-500-05	1315829	119	807,7	1417931	124	0	1743130	154	1,831	1743128	154	3,771
2-500-06	1358140	124	1155,5	1480216	131	0	1663780	148	1,838	1663780	148	3,619
2-500-07	1436202	131	1025,7	1647110	147	0	1559000	139	1,568	1559002	139	3,489
2-500-08	1279768	116	356,9	1451866	128	0	1437040	127	1,295	1437036	127	3,446
2-500-09	1462176	134	588,9	1564237	139	0	1865820	166	2,068	1877004	167	3,862
2-500-10	1390435	127	1576,8	1506591	133	0	1481030	131	1,365	1481032	131	3,473
Average	1373243	125	1160,7	1510807	133	0	1636140	145	1,65	1639403,3	145	3,698

Source: Owner

Tables 2 and 3 correspond to the results obtained with the cases of 3 and 5 depots respectively.

Table 2
Results for the instances with 3 depots.

Instance	(Fischetti, Lodi, & Toth, 1999)			CSC			SAMC			DSA		
	Sol.	V	t(s)	Sol.	V	t(s)	Sol.	V	t(s)	Sol.	V	t(s)
3-100-01	307705	28	9,22	351806	31	0	369770	33	0,016	369770	33	0,14
3-100-02	300505	27	1,05	363854	32	0	332911	29	0,036	332390	29	0,156
3-100-03	316867	29	2,22	453416	41	0	326361	29	0,03	325417	29	0,227
3-100-04	336026	31	2,37	401782	36	0	353034	32	0,014	353034	32	0,139
3-100-05	278896	25	1,25	373345	33	0	316188	28	0,013	315848	28	0,146
3-100-06	368925	34	2,35	457438	41	0	399720	36	0,014	399004	36	0,189
3-100-07	287190	26	2,78	379028	34	0	317545	28	0,014	317251	28	0,147
3-100-08	338436	31	3,55	438800	40	0	381328	34	0,018	381328	34	0,164
3-100-09	275943	25	1,13	337510	30	0	315845	28	0,016	315470	28	0,158
Average	309642	28	2,8	391643	35	0	341837	30	0,018	341517,9	30	0,163
3-200-01	551657	50	151,05	635341	56	0	610763	54	0,129	610613	54	0,672
3-200-02	543805	50	124,93	725958	64	0	605991	54	0,122	603220	54	0,645
3-200-03	615675	57	7,18	754187	68	0	670359	60	0,218	669550	60	0,655
3-200-04	557339	51	112,22	669271	60	0	619914	55	0,148	619382	55	0,639
3-200-05	626364	57	55,12	801852	71	0	754295	67	0,276	753969	67	0,713
3-200-06	558414	51	6,65	619840	55	0	646211	58	0,246	645778	58	0,644
3-200-07	595605	55	33,48	722203	65	0	695352	62	0,191	693385	62	0,659
3-200-08	562311	51	15,22	656190	58	0	709807	63	0,188	709764	63	0,672
3-200-09	671037	62	196,08	726214	65	0	713616	63	0,301	713614	63	0,704
3-200-10	565053	52	25,5	638540	57	0	629342	56	0,263	627032	56	0,638
Average	584726	54	72,74	694960	62	0	665565	59	0,208	341517,9	59	0,163
3-300-01	834240	77	87,43	1078853	97	0	913867	82	0,41	922669	83	1,644
3-300-02	830089	76	706,75	968949	86	0	914023	81	0,418	924605	82	1,613
3-300-03	799803	74	286,57	985586	89	0	940813	85	0,43	938206	85	1,595
3-300-04	850929	78	166,17	1167801	104	0	1005270	90	0,537	1001778	90	1,709
3-300-05	837460	77	576,2	986954	88	0	1078580	97	0,594	1089265	98	1,784
3-300-06	795110	73	142,05	1005917	90	0	906530	81	0,409	906530	81	1,692
3-300-07	774873	70	138,1	945060	83	0	891506	78	0,418	891183	78	1,686
3-300-08	916484	85	261,42	1128445	101	0	1030770	93	0,601	1037943	94	1,637
3-300-09	830364	77	560,77	1078629	96	0	938469	85	0,489	956599	87	1,596
3-300-10	850515	78	472,95	947914	84	0	938061	83	0,452	938061	83	1,623
Average	831987	76	339,84	1029411	92	0	955789	85	0,476	960683,9	86	1,658
3-400-01	1141067	106	3188,92	1248434	112	0	1209380	110	0,895	1219824	111	2,863
3-400-02	1059717	97	1617,23	1227357	109	0	1225600	109	0,855	1235660	110	3,258
3-400-03	1124169	103	2205,48	1302319	116	0	1439350	127	1,269	1438250	128	3,254
3-400-04	1091238	101	5142,95	1239273	111	0	1200600	108	0,8644	1199581	108	2,909
3-400-05	1159027	107	429,15	1261988	113	0	1254050	112	0,923	1263611	113	2,828
3-400-06	1042121	96	4476,55	1343667	120	0	1121320	100	0,744	1119620	100	2,827
3-400-07	1104156	101	4144,12	1283541	114	0	1301830	116	0,95	1301723	116	2,882
3-400-08	1050490	97	5480,95	1360377	122	0	1155560	104	0,857	1152848	104	2,915
3-400-09	1007810	93	775,32	1206448	108	0	1074970	96	0,7	1094872	98	2,918
3-400-10	1063571	98	4315,67	1218363	109	0	1283230	115	0,958	1282035	115	3,049
Average	1084337	100	3177,63	1269177	113	0	1226589	110	0,902	1230802,4	110	2,97

Source: Owner

Table 3
Results for the instances with 5 depots.

Instance	(Fischetti, Lodi, & Toth, 1999)			CSC			SAMC			DSA		
	Sol.	V	t(s)	Sol.	V	t(s)	Sol.	V	t(s)	Sol.	V	t(s)
5-100-01	365591	34	6,87	561255	51	0	383427	34	0,019	382959	34	0,225
5-100-02	295568	27	2,95	412302	37	0	338048	30	0,015	347823	31	0,233
5-100-03	314117	29	58,02	502385	46	0	336861	30	0,014	336823	30	0,241
5-100-04	340785	31	25,18	479689	43	0	353054	31	0,015	352665	31	0,247
5-100-05	306369	28	1,25	500939	45	0	337430	30	0,482	337430	30	0,238
5-100-06	333833	31	11,32	397227	36	0	348203	31	0,08	359029	32	0,211
5-100-07	296816	27	30,07	441027	39	0	319359	28	0,016	319292	28	0,237
5-100-08	355657	33	34,18	594980	54	0	386480	35	0,119	385655	35	0,221
5-100-09	306721	28	4,58	399974	36	0	339674	30	0,112	339670	30	0,261
5-100-10	291832	27	50,48	409052	37	0	322283	29	0,108	322074	29	0,222
Average	320728,9	29,5	22,49	469883	42,4	0	346481,9	30,8	0,098	348342	31	0,234
5-200-01	619511	58	603,5	1034677	94	0	686484	62	0,277	685622	62	0,952
5-200-02	601049	56	123,45	697076	63	0	698010	63	0,198	697028	63	1,015
5-200-03	623685	58	247,73	1018154	91	0	686854	61	0,229	684748	61	0,989
5-200-04	622408	58	883,22	828638	75	0	671081	60	0,161	669809	60	0,958
5-200-05	597086	55	221,12	774487	70	0	626297	56	0,142	626284	56	0,987
5-200-06	479571	44	160,57	661053	59	0	660133	59	0,257	658708	59	1,012
5-200-07	553880	51	128,22	780993	70	0	588656	52	0,15	588656	52	1,011
5-200-08	595291	55	594,38	1031595	91	0	632350	56	0,135	641329	57	0,994
5-200-09	588537	54	220,32	938616	83	0	707193	63	0,204	706420	63	1,016
5-200-10	593183	54	231,77	729178	65	0	692785	61	0,198	691624	61	1,023
Average	587420,1	54,3	341,43	849446,7	76,1	0	664984,3	59	0,195	665022,8	59	0,996
5-300-03	900205	84	3040,72	1606539	144	0	991831	88	0,513	985524	88	2,249
5-300-04	815586	76	847,63	1251207	114	0	925640	84	0,408	935134	85	2,212
5-300-05	868503	81	4506,17	1133577	102	0	933346	84	0,438	930690	84	2,256
5-300-06	787059	73	4863,87	988419	89	0	913226	82	0,408	922316	83	2,463
5-300-07	811301	75	2799,87	1124489	101	0	913880	81	0,417	912500	81	2,264
5-300-08	780788	72	5796,38	1030059	93	0	844289	75	0,352	854214	76	2,159
5-300-09	850934	79	3148,93	1143755	103	0	1060000	95	0,573	1067517	96	2,271
5-300-10	819068	76	2395,4	1019353	92	0	929966	83	0,441	929414	83	2,247
Average	827447	77	3130,49	1179758	106	0	934590	84	0,447	936837,8	84	2,278
5-300-03	900205	84	3040,72	1606539	144	0	991831	88	0,513	985524	88	2,249
5-300-04	815586	76	847,63	1251207	114	0	925640	84	0,408	935134	85	2,212

Source: Owner

Although the proposed constructive algorithms do not equal the solutions presented in Fischetti, Lodi, & Toth, (1999) where the test cases are solved with specific techniques, this study achieves response times in the real case of the operation of companies operating mass transport systems. It is much more important to offer a solution that reprograms fleet itineraries quickly since the failure to comply with the scheduled trips represents economic sanctions for BRT operators. Having zero execution times means that combine methods are allowing fast solution and high quality, are the most appropriate methodology for handling failures or interruptions of the actual operation of public transport.

Finally, the algorithms presented in this work were applied to the operation of INTEGRA S.A. Specifically, the daily services provided by the articulated buses. This case study consists of 719 trips that must be performed by a fleet of 36 vehicles distributed between two depots located in opposing points of the city of Pereira (depot Cuba and depot Dosquebradas). Dosquebradas has a fleet of 20 vehicles and Cuba 16 vehicles. Reducing the number of vehicles, in addition to lowering operating costs, provides more reserve vehicles to alleviate contingencies or to facilitate fleet maintenance. In Table 4 the performance of the algorithms regarding the number of vehicles used and the computational time required for the real instance is described.

Table 4
Comparative table examining daily case operation.

Instance	Manual Programming		CSC		SAMC		DSA	
	V	t(s)	V	T(s)	V	T(s)	V	T(s)
719 trips -	36	14400	66	1,25	35	7,33	36	8,95

Source: Owner

The impact on the way travel is currently programmed is inefficient since this procedure is performed manually. Using the CSC algorithm offers quick response, but is unfeasible because the required fleet exceeds the 36 available vehicles. The SAMC algorithm saves a vehicle from the operation. The next most efficient algorithm is DSA, equaling the manual programming. The excess vehicle offered by the CSC algorithm has a significant impact on costs, reduction of gas emissions and increases the flexibility of the operator to handle contingencies.

5. Concluding remarks

In this paper, we present three constructive algorithms to solve the problem of programming of Passenger Public Transport Vehicles, known in the literature as MDVSP. Although numerous solutions have been suggested, the NP-hard problem remains. No exact algorithm can solve the computational times appropriate for the daily scheduling of transport companies. For this reason, from the practical point of view, it is necessary to have alternative solutions, based on heuristic or mathematical methods offering good quality, efficient solutions. The results obtained in the test cases used to reflect that the first construct (CSC) performs satisfactorily in the instances with two deposits, whereas the construct SAMC and DSA present better development in larger test cases (5 deposits and 500 trips).

The validation performed with the test cases in the literature is concordant with the behavior of the implementations with the real case in the programming of vehicles of the Mass Transportation System of the AMCO. Using only constructive algorithms, not only does time and effort of fleet assignment decrease, but also the use of the number of the articulated vehicle is decreased in the 719 trips that are mandatory. The algorithms presented in this work suggest a more efficient method to construct population metaheuristics, where the improvements obtained could be increased. These constructs provide a tool that is adjusted to the minimum response times required when reprogramming the services when contingencies arise.

References

BRTData (2017). Global brt data. https://brtdata.org/. Accessed in 08-11-2017.

Bunte, S., & Kliewer, N. (2009). An overview on vehicle scheduling models. *Public Transport*, 1(4), 299-317.

Ceder, A. (2007). Public transit planning and operation: Modeling, practice and behavior. *CRC press*. 1-615.

Dell'Amico, M., Fischetti, M., & Toth, P. (1993). Heuristic algorithms for the multiple depot vehicle scheduling problem. *Management Science*, 39(1), 115-125.

Desaulniers, G., Lavigne, J., & Soumis, F. (1998). Multi-depot vehicle scheduling problems with time windows and waiting costs. *European Journal of Operational Research*, 111(3), 479-494.

Escobar, J. W. (2009). Modelación y optimización de redes de distribución de productos de consumo masivo con elementos estocásticos. In *Proceedings of XIV Latin American Summer Workshop on Operations Research (ELAVIO)*, El Fuerte, Mexico.

Escobar, J. W., Bravo, J. J., & Vidal, C. J. (2012). Optimización de redes de distribución de productos de consumo masivo en condiciones de riesgo. In *Proceedings of XXXIII Congreso Nacional de Estadística e Investigación Operativa (SEIO)*, Madrid, Spain.

Escobar, J. W., Linfati, R., & Toth, P. (2013). A two-phase hybrid heuristic algorithm for the capacitated location-routing problem. *Computers & Operations Research*, 40(1), 70-79.

Escobar, J. W., Linfati, R., Toth, P., & Baldoquin, M. G. (2014a). A hybrid granular tabu search algorithm for the multi-depot vehicle routing problem. *Journal of Heuristics*, 20(5), 483-509.

Escobar, J. W., Linfati, R., Baldoquin, M. G., & Toth, P. (2014b). A Granular Variable Tabu Neighborhood Search for the capacitated location-routing problem. *Transportation Research Part B: Methodological*, 67, 344-356.

Fischetti, M., Lodi, A., & Toth, P. (1999). A branch-and-cut algorithm for the multiple depot vehicle scheduling problem. *Dipartimento di Elittronica e Informatica*, Università di Padova, Italy.

Gintner, V., Kliewer, N., & Suhl, L. (2005). Solving large multiple-depot multiple-vehicle-type bus scheduling problems in practice. *OR Spectrum*, 27(4), 507-523.

Guedes, P. C., & Borenstein, D. (2015). Column generation based heuristic framework for the multiple-depot vehicle type scheduling problem. *Computers & Industrial Engineering*, 90, 361-370.

Guedes, P. C., Lopes, W. P., Rohde, L. R., & Borenstein, D. (2016). Simple and efficient heuristic approach for the multiple-depot vehicle scheduling problem. *Optimization Letters*, 10(7), 1449-1461.

Guedes, P. C., & Borenstein, D. (2018). Real-time multi-depot vehicle type rescheduling problem. *Transportation Research Part B: Methodological*, 108, 217-234.

Hadjar, A., Marcotte, O., & Soumis, F. (2006). A branch-and-cut algorithm for the multiple depot vehicle scheduling problem. *Operations Research*, 54(1), 130-149.

Hassold, S., & Ceder, A. A. (2014). Public transport vehicle scheduling featuring multiple vehicle types. *Transportation Research Part B: Methodological*, 67, 129-143.

Huisman, D., Freling, R., & Wagelmans, A. P. (2004). A robust solution approach to the dynamic vehicle scheduling problem. *Transportation Science*, 38(4), 447-458.

Ibarra-Rojas, O. J., Delgado, F., Giesen, R., & Muñoz, J. C. (2015). Planning, operation, and control of bus transport systems: A literature review. *Transportation Research Part B: Methodological*, 77, 38-75.

Kliewer, N., Mellouli, T., & Suhl, L. (2006). A time–space network based exact optimization model for multi-depot bus scheduling. *European Journal of Operational Research*, 175(3), 1616-1627.

Kulkarni, S., Krishnamoorthy, M., Ranade, A., Ernst, A. T., & Patil, R. (2018). A new formulation and a column generation-based heuristic for the multiple depot vehicle scheduling problem. *Transportation Research Part B: Methodological*, 118, 457-487.

Laurent, B., & Hao, J. K. (2009). Iterated local search for the multiple depot vehicle scheduling problem. *Computers & Industrial Engineering*, 57(1), 277-286.

Lin, S., & Kernighan, B. W. (1973). An effective heuristic algorithm for the traveling-salesman problem. *Operations Research*, 21(2), 498-516.

Muñoz, J. C., & Paget-Seekins, L. (Eds.). (2016). Restructuring public transport through Bus Rapid Transit: An international and interdisciplinary perspective. *Policy Press*.

Otsuki, T., & Aihara, K. (2016). New variable depth local search for multiple depot vehicle scheduling problems. *Journal of Heuristics*, 22(4), 567-585.

Pepin, A. S., Desaulniers, G., Hertz, A., & Huisman, D. (2009). A comparison of five heuristics for the multiple depot vehicle scheduling problem. *Journal of Scheduling*, 12(1), 17.

Polo, A., Peña, N., Muñoz, D., Cañón, A., & Escobar, J. W. (2018). Robust design of a closed-loop supply chain under uncertainty conditions integrating financial criteria. *Omega,* forthcoming.

Prins, C. (2004). A simple and effective evolutionary algorithm for the vehicle routing problem. *Computers & Operations Research*, 31(12), 1985-2002.

Semedo, D., Barahona, P., & Medeiros, P. (2015). Asynchronous Parallel Ant-Colony Optimization Strategies: Application to the Multi-Depot Vehicle Scheduling Problem with Line Exchanges. *In INForum*, Technical Report, 1 – 16.

Shui, X., Zuo, X., Chen, C., & Smith, A. E. (2015). A clonal selection algorithm for urban bus vehicle scheduling. *Applied Soft Computing*, 36, 36-44.

Uçar, E., Birbil, Ş. İ., & Muter, İ. (2017). Managing disruptions in the multi-depot vehicle scheduling problem. *Transportation Research Part B: Methodological*, 105, 249-269.

Wang, H., & Shen, J. (2007). Heuristic approaches for solving transit vehicle scheduling problem with route and fueling time constraints. *Applied Mathematics and Computation*, 190(2), 1237-1249.

Xu, X., Ye, Z., Li, J., & Wang, C. (2018). Solving a Large-Scale Multi-Depot Vehicle Scheduling Problem in Urban Bus Systems. *Mathematical Problems in Engineering*, 2018, 1 – 13.