SEPARABLE CROSS DECOMPOSITION FOR THE ALLOCATION-DISTRIBUTION PROBLEM IN THE IRP

Mayra Elizondo-Cortés, Facultad de Ingeniería, Universidad Nacional Autónoma de México, mareli@avantel.net

Ricardo Aceves-García, Facultad de Ingeniería, Universidad Nacional Autónoma de México, aceves@unam.mx

ABSTRACT

The Inventory-Routing Problem (IRP) involves a central warehouse, a fleet of trucks with finite capacity, a set of customers, and a known storage capacity. The objective is to determine when to service each customer, as well as what route each truck should take, with the least expense. IRP is a NP-hard problem this means that searching for solutions can take a very long time. A three-phase strategy is used to solve the problem. In the idealization of the strategy, the need to define algorithms that would solve more manageable problems in an easy and implementable way was every present. Thus in the second phase we uses Separable Cross Decomposition [1] to solve an Allocation-Distribution Problem, and a algorithm is obtained that produces a solution for two transport subproblems and their solution algorithms have a low order of complexity, $O(n^3)$. And the result is a very efficient algorithm for large cases of the IRP.

Keywords: Cross Decomposition, Inventory, Routing.

INTRODUCTION

Many private and public organizations have realized that goods or services can be more valuable for a customer if there is good logistics management. The important concept of customer value can be created through the availability of the product in place and time by allocating orders, among other logistics services as part of the supply chain management.

A combination of elements that can create logistic value is when the supplier manage inventory, because he saves on distribution costs with the possibility of improving the coordination of deliveries, while customers do not have to use resources on inventory management, since the transportation of the products and inventory management is the most expensive aspect of supply chains.

In spite of this, this combination of elements has not been more widely applied because the fact that it is extremely complicated to device a distribution strategy to minimize transport and inventory costs. This is called the Inventory Routing Problem (IRP). These kinds of problems are called NP-hard because the optimal solution in a reasonable running time is almost impossible.

This paper deals with the distribution problem in the IRP and proposes a strategy to resolve this problem, by a solution of two transportation subproblems.

LITERATURE REVIEW

Over time different methods have been developed to deal with this problem of distribution, though in distinct scenarios or formulations.

Baita *et al.* [2] developed a classification of solution methods into two main types: 1. frequency domain methods, and 2. time domain methods. In the former the decision variables are the frequencies of replenishment or time between deliveries. In the latter, the delivery schedule is decided with discrete time models and the amounts and routes are decided, using fixed time intervals. Time domain methods are the focus of the present study.

The time domain methods are used in a number of papers by Dror and Ball [5], Dror and Levy [6], Dror and Trudeau [7], and Trudeau and Dror [8]. In each time interval only customers who have reached their inventory safety level are attended to. Just one product should be delivered from one warehouse to several customers, whose demands, different in each time period, are deterministic in [5] and [6]; stochastic in [8]; stochastic or deterministic in [7].

Dror and Ball [5] proposed a mixed integer program (MIP) model where the effect of current decisions on subsequent periods is taken into account as a factor of penalties and incentives. Trudeau and Dror [8] developed some heuristics which use linear MIP submodels to solve the problem. On the basis of these ideas, Dror and Trudeau [7] present a computerized distribution system for one product (heating oil), evaluated using real data for more than 2000 customers.

Chandra and Fisher [9] follow two methods: In the first one a production planning and routing vehicle problem are solved separately, and in the second one the two problems are coordinated within one model. Some computational experiments show that such coordination can reduce costs.

Hwang [10] demonstrates an interesting application for food distribution to deal with the famine in North Korea that independently solves the problems of inventory replenishment, distribution to sectors and routing scheduling using modified heuristics (Clarke and Wright and Sweep).

Another interesting application, by Campbell *et al.* [11], demonstrates a solution method that individually solves the problems of customer attention and route scheduling, for a period of one day, using a combination of the GRASP heuristic and integer programming, for a real application of industrial gas distribution for the company PRAXAIR.

Ouimet [12] applies the well-known metaheuristic Taboo Search to the IRP. Taboo Search has demonstrated very good results for VRP [13]. The application for randomly generated items showed efficiency for multiple-customer problems and fleets of different sizes. However, it was limited in respect of the length of the planning period.

The time domain method is ideal for situations where the conditions can vary significantly in time. However, this flexibility is counteracted by the need to recalculate different solutions in each time interval.

DECOMPOSITION TECHNIQUES

As part of the time domain methods, the Decomposition Technique has given good results. This technique, which is used to solve large-scale problems as well as problems of linear programming with special structure, is characterized by a decomposition of the original system into subsystems, each one with a minor or separate subproblem. The subproblems are considered manageable because of their size, unlike the case of the complete problem that exceeds the available computer capacity.

The principles of decomposition, including the primal and dual forms are known as Benders Decomposition and Lagrangean Decomposition (Dantzig-Wolfe), respectively. These techniques make it possible to take advantage of the special structure of the problem by solving a sequence of simpler problems. Thus, the primal or dual substructure of the problem are exploited. In fact, such strategies are considered dual for each other and in each one of them the cycling occurs between the master problem and the subproblem. In particular, the decomposition techniques were proposed and used to solve the IRP, by authors such as Federgruen and Zipkin [1]; Chien, Balakrishnan and Wong [14] and Christiansen [15].

However, the major disadvantage in using only one of these techniques is the need to solve in all iterations a master problem, which is highly complex in a mixed-integer problem or large linear problem. To avoid this, Van Roy [16] proposes a new decomposition method called Cross Decomposition.

This strategy is based on the relationship between Benders' and Dantzig-Wolfe's principles of decomposition. In fact, it is possible to establish that the Lagrangean dual subproblem is a relaxed master problem in the Benders' decomposition and, at the same time, Benders' subproblem can be considered a relaxed master problem for the dual decomposition. It is also known that both methods are dual pairs, i.e., if Benders' algorithm is applied to an LP problem, it coincides with the Dantzig-Wolfe decomposition algorithm applied to the same LP problem [17].

The basic idea of Cross Decomposition is to use a "ping pong" method for both subproblems in a single solution procedure, exploiting the fact that in some cases both subproblems are simple to solve. Van Roy [16], applied Cross Decomposition to a set of standard MIP test problems and obtained a 20% (average) reduction in solution time compared to the best algorithms known at the time.

DEFINITION OF THE PROBLEM

A single product (boxes, containers) must be distributed from a warehouse to several customers in different geographic locations. The geographic position of the customers and the warehouse is known, as well as the distances between the warehouse and the customers and the distances between all the customers.

The data cost (storage and organization) are used to characterize the customers. It is assumed that every customer's demand for the product is known and constant in time. The deliveries begin at the start of every day, using vehicles characterized by a given capacity, fixed costs and variable costs in regards to covered distance. Time is the main decision for the problem as it defines fixed time periods of one day. Moreover, it is assumed that the customer's daily demand does not exceed their storage capacity.

There is no limitation in the central warehouse's supplying capacity. A mixed fleet of vehicles is available to take care of the customers in the time period. However, there are limitations in the capacity of the vehicles and their size is a decision variable that will be involved in the optimization process.

The aim purpose is to serve the customers' demand at the lowest cost, incorporate inventory-related costs (storage costs, supply costs and distribution costs), including fixed costs plus the costs involved in the distanced covered by the vehicles used.

So, the problem is to find the defined supply and distribution strategies, based on the answers to the four following questions that at the start of each day:

1. Which customers must be supplied?
2. What vehicle will be used to supply each customer?
3. What volume of the product is to be delivered to the customers when visited?
4. What delivery route will each vehicle take?

PROPOSED STRATEGY

The IRP is NP-hard for a context of NP-completeness, however, as the vehicle-routing component of the problem now includes inventory restrictions, is NP-hard [12], and generalizes the TSP [18]. The fact is, all the non-trivial vehicle-routing problems are NP-hard [19], so it is unlikely that an algorithm of polinomial time can obtain the optimal solution. Considering the previous, the revision of existing literature is developed with a three-phase strategy, which responds to the four questions for the solution of the treated IRP, these phases are presented as follows:

Phase I answers the question: *which customers must be supplied that day?*

Applying an Optional Replenishment System (ORS) will force the review of the inventory level and replenishment orders at a fixed time frequency if the level has fallen below a certain amount. Thus, this is basically a fixed time period model. A demand level for day D is forecast from historical data from daily demand, as supplying stock involves time and economic resources, a minimum order size Q can be established. So, a good option for calculating Q is to use the Economic Order Quantity (EOQ) equation. Therefore, when the product is checked, the inventory position I is subtracted from the required level of restocking D and the result is called q.

Formally establishing,

$$q = D - I$$

where

D : level of forecast demand
I : current inventory level

q : shortfall for achieving the maximum level of inventory

Q : amount of the acceptable minimum order

And the rule is that only if $I \leq Q$, q is sent; if $q = 0$ the customer is not served. Using an ORS for each customer, depending on their specific circumstances, the supplier decides if they should be supplied on that day or wait for the next review. We will obtain a subgroup of customers who will be visited that day.

The information required during this phase consists in the specific D, I, q and Q for each customer depending on their demand, capacity, inventory levels, restocking costs, shortfall and storage. Once the question is answered, you go on to the next phase.

Phase II responds to the questions: 2. what vehicle will be used to supply each customer? and 3. What volume of the product is to be delivered to the customers when visited?.

These questions are answered by applying a decomposition technique called Separable Cross Decomposition [20], to a Facility Location (FL) problem that, by analogy, assumes that the services are vehicles that will be delivered to the customers. The use of Separable Cross Decomposition contains a result that is fundamental to the study of the problem's structure, solving a problem that uses binary variables to allocate vehicles to customers.

Therefore, the problem of Allocation of Vehicles and Distribution (AVD) with the following structure:

$$
\begin{aligned}
& \textit{Minimize} \quad \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} + \sum_{i=1}^{m} f_i y_i \\
& \textit{subject to} \\
& \sum_{i=1}^{m} x_{ij} = 1, \qquad \forall j \qquad (1) \\
& x_{ij} \leq y_i, \qquad \forall i, j \qquad (2) \qquad (\mathrm{P}) \\
& \sum_{i=1}^{m} a_i y_i \geq \sum_{j=1}^{n} d_j \qquad (3) \\
& \sum_{j=1}^{n} d_j x_{ij} \leq a_i y_i, \quad \forall i \qquad (4) \\
& x_{ij} \geq 0, \quad y_i = 0,1, \quad \forall i, j.
\end{aligned}
$$

where

m : number of available vehicles

n : number of customers

d_j : customer's demand j

f_i : fixed cost of the route i

a_i : vehicle capacity i

c_{ij} : cost of distribution to customer j vehicle i, the cost is determined as a function of distance

x_{ij} : fraction of the customer's total demand that has been supplied j using vehicle i

y_i = 1 if vehicle i is used, 0 if not.

The restriction (1) ensures that the demand is totally supplied, (2) establishes distribution only with active vehicles, (3) considers the use of enough vehicles to attend to the demand and (4) avoids exceeding the capacity of the vehicle.

There are some difficulties in optimizing these types of problem because of their size and combinatorial structure. In fact, logistic-type problems such as the IRP and AVD are usually very big considering their large number of variables and restrictions. The AVD is very complicated, since the single basic decision to use or not a vehicle becomes a problem with a complex combinatorial structure.

This problem, in particular, has two types of inherent decisions: choosing the vehicles to be used and the best way of distributing the supply to the customers. This makes decomposition techniques an attractive option for dealing with them, if the discrete decision of choosing the vehicle has been done, the next distribution problem is, in general, easier to solve.

Even if it is not possible to take advantage of this characteristic in the design of solution algorithms, decomposition can still be a very attractive option. If the AVD problem wasn't affected by the discrete decision of choosing services and was formulated as a linear programming problem (relaxing the integrality restrictions for the problem's variables), it would still be very big and difficult to solve. Fortunately this problem has a special structure that can be exploited by decomposition techniques [20].

As a result of Van Roy's research [21] and the structure of the AVD, we can confirm that when the cross decomposition strategy is applied to this problem the consequence is a solution produced for two subproblems, incorporating a process among them, reducing the number of master problems to be solved. Three facts are fundamental for the Cross:

1. The relationship between the primal (Benders) and the dual (through Lagrangean relaxation) decomposition.
2. The subproblems ($SP\,x_2$) and ($SD\,u_2$) can be considered master problems for each other.
3. Considerations under which *(P)* can only be solved by iterating between both subproblems.

The research developed by Aceves [20], incorporates into this cross process the strategy of Lagrangean Separable Relaxation, a special case that is very advantageous because, with this scheme, none of the original restrictions disappear and it isn't necessary to choose between the quality of the bound obtained and the degree of difficulty of the problem that remains.

When this strategy is incorporated, it establishes that it is not necessary to use the master problem in the solution, i.e., it can be solved just by iterating between subproblems, completely avoiding the master problem. This procedure is called Separable Cross Decomposition and is the procedure applied to the AVD problem in this phase.

Two subproblems are obtained when Separable Cross Decomposition is applied to the AVD problem.

Using the Benders Decomposition to the (P) problem gives us:

$$\begin{aligned} &Minimize \quad \sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij} + \sum_{i=1}^{m} f_i y_i \\ &subject\ \text{to} \\ &\sum_{i=1}^{m} x_{ij} = 1, \qquad \forall j \qquad (\mathrm{SP_y}) \\ &\sum_{j=1}^{n} d_j x_{ij} \le a_i y_i, \quad \forall i \\ &0 \le x_{ij} \le y_i, \qquad \forall i, j \end{aligned}$$

This is a transportation problem, for which there are very efficient solution methods.

Therefore, the Benders master problem or primal can be expressed like this:

$$\begin{aligned} &\underset{y\in\{0,1\},\rho}{Min} \ \rho \\ &s.a. \\ &\sum_j v_j^t + \sum_i \left(f_i - a_i u_i^t - \sum_j w_{ij}^t \right) y_i \le \rho, \\ &t \in T_P, \end{aligned}$$

Where T_P is the index set of all basic feasible solutions of the dual constraint, ie, they are Benders cuts.

Now apply the following process for the Lagrangean Separable Relaxation schemes, for the problem AVD we have:

- Copy $\sum_j d_j\ x_{ij} = \sum_j d_j\ x'_{ij}$ and $y_i = y'_i$ on the restriction (4).
- Duplicate the restriction (3).
- Dualize equality constraints.
- Separate the problem to obtain two subproblems, each with the following sets of constraints: ((1), (2), (3)) and ((2), (3), (4)).

The problem AVD can be formulated as:

$$
\begin{aligned}
& Min. \sum_i \sum_j C_{ij} x_{ij} + \sum_i f_i y_i \\
& s.a. \\
& \quad \sum_i x_{ij} = 1, \qquad \forall j, \qquad (1), \\
& \quad x_{ij} \le y_i, \qquad \forall i, j, \qquad (2), \\
& \quad \sum_i a_i y_i \ge \sum_j d_j, \quad \forall j, \qquad (3), \qquad (D) \\
& \quad \sum_j d_j x'_{ij} \le a_i y'_i, \quad \forall i, \qquad (4), \\
& \quad \sum_j d_j x_{ij} = \sum_j d_j x'_{ij}, \quad \forall i, \quad (5), \\
& \quad y_i = y'_i, \qquad \forall i, \qquad (6). \\
& \quad x_{ij} \ge 0, y_i = 1,0 \qquad \forall i, j.
\end{aligned}
$$

Relaxing (5) and (6), making algebraic simplifications and separating them, we have two subproblems:
.

$$
\begin{aligned}
& Q = \underset{y}{Min}. \sum_i (f_i + \nu_i) y_i \\
& s.a. \\
& \quad \sum_i a_i y_i \ge D, \quad \forall j, \\
& \quad y_i = 0,1, \quad \forall i.
\end{aligned}
$$

$$
\begin{aligned}
& S = \underset{x}{Min}. \sum_i \sum_j \left(C_{ij} - d_j \lambda_i \right) x_{ij} \\
& s.a. \\
& \quad \sum_i x_{ij} = 1, \quad \forall j, \\
& \quad 0 \le x_{ij} \le y_i, \quad \forall i, j, \\
& \quad \lambda_i \, unrestricted, \quad y_i = 0,1, \; \forall i.
\end{aligned}
$$

When we incorporate restriction 4 to the subproblem S to reinforce it, it can be formulated as:

$$S = Min.\sum_i \sum_j \left(C_{ij} - d_j \lambda_i\right) x_{ij}$$
$$s.a.$$
$$\sum_i x_{ij} = 1, \qquad \forall j, \qquad\qquad (SD_\lambda)$$
$$\sum_j d_j x_{ij} \le a_i, \quad \forall i,$$
$$x_{ij} \ge 0, \quad \forall i, j, \quad \lambda_i \, unrestricted, \quad \forall i,$$

This is a transportation problem, for which there are very efficient algorithms to solve it.

Since Q and S problems are function of λ_i, for i = 1,..., m, we need to change in the Q problem $v_i = a_i \lambda_i$ and $\lambda_i = -f_i / a_i$,or, more generally $\lambda_i = -\frac{f_i}{\sum_j d_j x_{ij}}$ when $\sum_j d_j x_{ij} = a_i \; , \forall i.$

The Q problem may be formulated as:

$$Q = Min.\sum_i \left[f_i - \left(\frac{f_i}{a_i}\right) a_i \right] y_i$$
$$s.a.$$
$$\sum_i a_i y_i \ge D, \forall j,$$
$$y_i = 0,1, \forall i.$$

Which result is Q = 0. This allows to establish that the origins *i*, for i = 1,...,m, that satisfies the demand also solves the S transportation problem, and minimizes the dual Lagrangean subproblem. With this result it has been possible to develop an algorithm simpler that those previously obtained [20].

SEPARABLE CROSS DECOMPOSITION ALGORITHM

Using the primal SP_y and dual SD_λ subproblems and iterating between their solutions with the purpose of fixing the values of the primal y_i or dual λ_i variables it is possible exchange the following stages:

i) fix y_i at its current value and solve the Benders subproblem SP_y that is a transport problem, in order to generate a new value for the upper bound of *(P)* and,
ii) fix λ_i at its current value and solve the Lagrangean subproblem SD_λ to generate a new value for the lower bound of *(P)*.

The stages of the algorithm are:

1.- **Start.**

$v_D(-\infty),\ \ v_P(+\infty);\ \ Y_i^0 = 1,\ \ for\ \ i = 1,...,m;$

$\lambda_i^0 = -\dfrac{f_i}{a_i},\ \ \text{for}\ \ i = 1,...,m.$

2.- **Solve.** Dual subproblem SD_{λ^k} (transport problem), to obtain $y_i^1 = 1,\ \ \sum_j x_{ij}$ y $v(SD_{\lambda^k})$.

2.1.-**Calculate.** $\lambda_i^k = -\dfrac{f_i}{\sum_j x_{ij}},\ \ for\ \ i = 1,...,m$.

3.- **Test.** If $\lambda_i^{k-1} = \lambda_i^k$ for $y_i^k = 1$, then end. And identify which $y_i^k = 1$ for $i = 1,\ldots,m$.

4.- **Solve.** Primal subproblem $SP_{y_i^k}$ (transport problem), to obtain $v(SP_{y^k})$.

5.- **Test.** If $v(SP_{y^k}) = v(SD_{\lambda^k})$, then end. Otherwise return to stage 2 but now with λ_i^k.

An analysis of the convergence of the Cross Decomposition algorithm developed by Holmberg [22] indicates that the cross decomposition algorithm has finite convergence for problems where the Dantzig-Wolfe decomposition algorithm or the Benders decomposition algorithms have finite convergence. Thus, it is necessary to see if the algorithm used to solve the transport problems has finite convergence.

In subsequent iterations of the Separable Cross Decomposition the most economic routes gradually get cheaper and, in consequence, we will be able to allocate greater flow until they are saturated, either by using the higher capacity of the vehicle or meeting the demand, i.e.:

$$\sum_j x_{tj}^{k-1} \le \sum_j x_{tj}^k \le \sum_j x_{tj}^{k+1} \le \ldots \quad \text{with} \quad \lambda_t^{k-1} \le \lambda_t^k \le \lambda_t^{k+1} \le \ldots \le -\frac{f_t}{a_t}$$

Thus, the vehicles that satisfy the demand obtained from the solution of the Lagrangean dual subproblem SD_λ can be used as active vehicles; i.e. the $y_i = 1$, with $i = 1,\ldots,m$, that were fixed in the primal subproblem SP_y.

So the separable cross decomposition algorithm ends in a finite number of iterations and obtains the solution to the FL problem of Phase II. Thus it is known: which vehicle looks after each customers and what amount of product is delivered to them.

From this result you go on to the next phase.

Phase III responds to the question: *4. what delivery route will each vehicle take?*

This is answered with the solution used at Salesman Problem (TSP) for each vehicle?

According to Castañeda [23], the idea of hybridation arises as a result of the 2-Opt theory and mentions that this method is better when begins with a good route. So the Adaptation of Prim's Algorithm is used to find this initial route that will be used by the 2-Opt method, thus, we obtain the Adaptation-Prim-2Opt-Hybrid method.

The proposed algorithm is the Adaptation-Prim-2-Opt-Hybrid heuristic method, whose components are the 2-Opt Method and an Adaptation of Prim's Algorithm Method. The use of this heuristic obeys the initial supposition of a great number of customers for each vehicle to visit. Castañeda [23] has an interesting study where she compares heuristic and exact methods for the TSP and proposes the aforementioned hybrid as highly advantageous in respect of other heuristics for very large TSP cases.

The 2-Opt algorithm and Prim Adjustment are generally presented below.

The 2-Opt procedure is an approximate solution to the symmetric TSP problem, with n nodes and an adjacency matrix W. The initial cycle is forms by a set of edges H= $(\mathrm{x}_{1,}\ \mathrm{x}_2\ , \dots , \mathrm{x}_\mathrm{n})$. Let X = $\{\mathrm{x}_\mathrm{i}, \mathrm{x}_\mathrm{j}\}$ be the set of two edges in H, which will be removed and replaced by the edges Y=$\{\mathrm{y}_\mathrm{p}\ , \mathrm{y}_\mathrm{q}\}$, if there is an improvement. This is, $\mathrm{H}' = (\mathrm{H} - \mathrm{X}) \cup \mathrm{Y}$ is a new and improved route. Note that the two edges $\mathrm{x}_\mathrm{i}\ , \mathrm{x}_\mathrm{j}$ in X can't be adjacent and when the set X has been selected, the set Y is determined.

So we'll have $\frac{n(n-3)}{2}$ possible routes H' for a given H. For either of these routes improvement will be denoted with δ, obtained by:

$$\delta = w(H) - w(H') = w(x_i) + w(x_j) - w(y_p) - w(y_q)$$

If $\delta_{max} > 0$, the corresponding solution is used as the initial route and the whole process is repeated, i.e., the improving of the route is continued until δ_{max} becomes a negative number or zero.

Prim Adjustment method can be applied to TSP, because it produces a minimal spanning tree *T*. At each stage the algorithm searches for a lower-cost edge that connects a vertex of *T* with a new vertex outside *T*. That edge and the vertex are added to *T* and the process is repeated. This is called a glutton or greedy algorithm because it always adds the minimum cost edge and removes the oldest.

The algorithm makes minimal set, while keeping connected and acyclic subgraph. Is doesn't need the edges of G to be ordered in advance. The adjustment is to take the vertices of the minimum spanning tree and produce a Hamiltonian cycle.

EFFICIENCY OF THE STRATEGY

In the idealization of the strategy, the need to define algorithms that would solve more manageable problems in an easy and implementable way was very present. As we already mentioned, the essential part of the strategy is the application of Separable Cross Decomposition in Phase II.

Thus, an algorithm is obtained that produces a solution for two transport subproblems. It's interesting to note the enormous advantage of having this type of problem as their solution algorithms have a low order of complexity, $O(n^3)$ [13] and moreover, any type of commercial software can be used to solve them.

The proposed strategy consists in three aforementioned stages. Each phase's process has the following runtime:

Table I – Order of the phases

Phase of the strategy	Order of the algorithm
Phase I	$O(n)$
Phase II	$O(n^2)$
Phase III	$O(n^3)$

In such way that the order of the proposed strategy is $O(n^3)$, which makes it extremely efficient.

In order to assess the strategy's efficiency, 9 problems were constructed: 4 small problems with less than 10 customers, 2 medium-sized with 30 and 35 customers respectively and 3 large problems with 150, 200 and 250 customers. The fixed costs, capacities and demands were taken from the examples given in Aceves [20].

The data generated as random variables were the customers' demands, the matrices for distance between customers and between customers and central warehouse. The variables were obtained by means of the statistical package STATGRAPHICS. The capacities of the vehicles (m^3) took the values 24,16,8 and 4. The fixed costs varied between 50, 40, 35, 30, 25 and 20.

Every problem was solved by obtaining the optimum solution using the Branch and Bound (B-B) algorithm by means of the LINGO package. Afterwards, each problem was solved with the proposed algorithm by means of a computer program developed in Pascal. The system used was Pentium 4, 2.00Ghz-256 MB RAM.

RESULTS

The results table for the problem's solution is the following:

Table II – The results obtained

Problem	Number of customers	Vehicles	Solution with the sugested strategy	Time	Optimal solution B-B	Time	Deviation from the optimal solution %
E005-03	4	3	9819	2 sec	9819	3 sec	0
E006-04	5	4	16724	2 sec	16724	3 sec	0

E008-02	7	2	21256	2 sec	21256	3 sec	0
E009-04	8	4	13114	2 sec	13114	3 sec	0
E031-09	30	9	4332	2 sec	4332	28 days	0
E036-11	35	11	3120	2 sec	3120	35 days	0
E151-08	150	8	14770	2 sec	Unknown	-	-
E201-16	200	16	44586	3 sec	Unknown	-	-
E251-20	250	20	72123	3 sec	Unknown	-	-

Some of the results obtained were: 1. - an extremely short solution time, with three iterations at most used to find the solution and, in every case, the results were obtained in less than 2 seconds; 2. - the quality of the solutions was excellent, the optimum was obtained in all of the problems that were assessed.

Among the advantages of the proposed strategy we have:

- Speed,
- quality of the solution,
- possibility of dealing with large numbers of clients,
- simple implementation and low cost.

CONCLUSION

An order strategy $O(n^3)$ was obtained of easily implementation and application that gives good quality results and is efficient in computational effort even for large problems.

This strategy has low order of complexity, which is more efficient than the methods used to solve the IRP. Thus, this work demonstrated the superiority of the proposed strategy as it can handle bigger problems with less computational effort.

It is necessary to promote and continue the research into the methodology for optimal solutions to different instances of the problem and reduce their computational complexity.

System's optimization is very recommended as a proposal that can consider the economic, social, political and technological challenges which the decision-makers are faced, this makes business more competitive. This is important for every type of organization and enterprise that wants to be updated and successful in their field of action and contributes to their own economic development and the one of their surroundings.

REFERENCES

[1] Federgruen, A. and Zipkin. P. (1984). A comprised vehicle routing and inventory allocation problem. *Operations Research,* 32:1019-1037.

[2] Baita, F., Ukovich, W., Pesenti, R. and Favaretto, D. (1998). Dynamic Routing-And-Inventory Problems: A Review. *Transportation Research* , 32: 8, 585-598.

[3] Daganzo, C. (1996). *Logistics System Analysis.* New York; Springer.

[4] Hall, R. W. (1992). A note on bounds for direct shipping cost. *Management Science,* 38: 1212.

[5] Dror, M. and Ball, M. (1987). Inventory/routing: reduction from an annual to a short-period problem. *Naval Research Logistics,* 34: 891-905.

[6] Dror, M. and Levy, L. (1986). A vehicle routing improvement algorithm comparison of a greedy and a matching implementation for inventory routing. *Computers and Operations Research,* 13: 33-45.

[7] Dror, M. and Trudeau, P. (1996). Cash flow optimization in delivery scheduling. *European Journal of Operational Research,* 88: 504-515.

[8] Trudeau, P. and Dror, M. (1992). Stochastic inventory routing: route design with stockouts and route failures. *Transportation Science,* 26: 171-184.

[9] Chandra, P. and Fisher, M.L. (1994). Coordination of production and distribution planning. *European Journal of Operational Research,* 72: 503-517.

[10] Hwang, H. (1999). A Food Distribution Model for Famine Relief. *Computers & Industrial Engineering* 37: 335-338.

[11] Campbell, A., Clarke L., and Savelsbergh M. (2002). Inventory Routing in Practice. *The Vehicle Routing Problem.* Eds. P. Toth and D. Vigo. Philadelphia: *Society for Industrial and Applied Mathematics,* 309-330.

[12] Cousineau-Ouimet, K. (2002). A Tabu Search Heuristic for the Inventory Routing Problem. *Artículo no editado. Department of Quantitative Methods, École des Hautes Études Commerciales.* Montreal, Canada.

[13] Toth, P. and Vigo, D. (2002). *The Vehicle Routing Problem. SIAM Monographs on Discrete Mathematics and Applications.* Philadelphia: Society for Industrial and Applied Mathematics.

[14] Chien, W., Balakrishnan, A. and Wong, R. (1989). An integrated inventory allocation and vehicle routing problem. *Transportation Science,* 23,67.

[15] Christiansen, M. (1999). Decomposition of a Combined Inventory and Time Constrained Ship Routing Problem. *Transportation Science,* 33: 3-16.

[16] Van Roy, T. J. (1983). Cross decomposition for mixed integer programming. *Mathematical Programming,* 25: 46-63.

[17] Lasdon, L.S. (1970). *Optimization Theory for Large Systems*. USA: McMillan Series in Operations Research.

[18] Garey, M. and Johnson, D. (1979). *Computers and intractability: A guide to the theory of np-completeness.* San Francisco, Calif., Freeman.

[19] Bramel, J. and Simchil Levi, D. (1995). A location based heuristic for general routing problems. *Operations Research*, 43: 649-660.

[20] Aceves García, R. (1996). Un Algoritmo para Resolver el Problema de Localización de Servicios con Restricciones de Demanda y Adicionales. *Tesis Doctoral (Investigación de Operaciones)*, México: *DEP-FI*, UNAM.

[21] Van Roy, T. J. (1984). A cross decomposition algorithm for capacitated facility location. *Operation Research Society of America*, 34: 145-163.

[22] Holmberg, K. (1990). On the Convergence of Cross Decomposition. *Mathematical Programming,* 47: 269-296.

[23] Castañeda Roldán, C. (2000). Estudio comparativo de diversos solution methods del problem del agente viajero (PAV). *Tesis de Maestría en Ingeniería en Sistemas Computacionales,* Puebla: Universidad de las Américas.