

Dynamic inventory routing problem: Policies considering network disruptions

Francisco Morales^a, Carlos Franco^{b*} and German Mendez-Giraldo^c

^aUniversidad Distrital Francisco José de Caldas, Bogotá, Colombia

^bUniversidad del Rosario, Bogotá, Colombia

^cUniversidad Distrital Francisco José de Caldas, Bogotá, Colombia

CHRONICLE

Article history:

Received June 18 2017
Received in Revised Format
August 25 2017
Accepted November 7 2017
Available online
November 7 2017

Keywords:

Inventory routing problem
Network disruption
Dynamic programming

ABSTRACT

In this paper, we introduce an inventory routing problem with network disruptions. In this problem, not only decisions on inventory levels and vehicle routing are made simultaneously, but also, we consider disruptions over the networks in which a number of arcs are vulnerable to these disruptions, leading to an increase in travel times. We develop a dynamic programming approach to deal with this situation, and we also evaluate some policies adapting well-known instances from the literature.

1. Introduction

The Inventory Routing Problem (IRP) is one extension of the classic vehicle routing problem and has been studied because of its industrial applicability. Vendor Managed Inventory system (VMI) is one of the most used strategies for coordinating inventory policies and transportation management while minimizing the logistics costs. This strategy can be modeled as an IRP and was introduced by Federgruen and Zipkin (1984).

IRP consists in satisfying the demand of customers that are geographically spread over a planning horizon while satisfying inventory levels constraints and vehicle capacity constraints (Azadeh et al., 2017). As an extension of the classic vehicle routing problem, IRP is a difficult optimization problem because of the coordination of the various related decisions. As an extension of the VMI strategy, the IRP can be also used to model distribution process for the same company when they allocates its own vehicles and the distribution is done to its own stores or retailers.

On the other hand, transportation networks deal with problems of traffic congestion, accidents or weather problems, also the demand can change, the time windows change or vehicles can suffer breakdowns, all these situations can produce an increment in the travel times or the necessity to change the planned routes need a change of the planned routes. Nowadays, some companies are developing navigation technologies

* Corresponding author

E-mail: carlosa.franco@urosario.edu.co (C. Franco)

to minimize uncertainties with data and that can consider real-time information. One problem for this kind of company is the high price of information and GPS services. There are also some theories and developments about the handling of disruptions (Yu & Qi, 2004) in order to satisfy solutions in real time when a disruption occurs in the distribution network.

As the classic Vehicle Routing Problem and the IRP have proved to be NP-hard in the static distribution management version, it is therefore more difficult to generate a solution in real time in order to adjust the distribution process in the event of a link disruption, and this new solution must be obtained as fast as possible. Some approximations for managing disruptions in supply chains have been proposed using different approximation methods (Xiangpei & Lijun, 2012; Li et al., 2007). Nevertheless, the IRP with link disruptions is a problem that has not been studied in depth in the literature.

The main purpose of this article is to introduce the dynamic IRP with network disruptions. Our main contributions are: to develop a framework based on dynamic programming to evaluate the dynamic Inventory Routing Problem considering disruptions over the network. Also, we evaluate different scenarios over a set of modified instances using probability distributions.

The remainder of this study is organized as follows: a literature overview about IRP and network disruption is developed in section 2, the mathematical formulation is presented in section 3, in section 4 we present the modeling framework and the algorithm used. Computational results and analysis are developed in section 5. Finally, conclusions are presented in section 6.

1. Literature review

Most of the studies in the literature consider the distribution of a single product from a single depot to a set of customers over a planning horizon using a fleet of homogeneous vehicles and developing a strategy for solving the problem (Archetti et al., 2007; Coelho et al., 2012; Coelho & Laporte, 2013; Franco & Figueroa-García, 2016). In the same way, disruptions over distribution networks have also been studied in the literature and some solution methods have been proposed.

The first approximations for disruption management were applied for scheduling problems relating to aircraft recovery (Teodorovic & Guberinic, 1984). Sever et al. (2013) proposed an approximation based on stochastic dynamic programming for the dynamic shortest path problem using online policies. The first study of vehicle routing problems with disruptions was the one proposed as a dynamic vehicle routing problem by Psaraftis (1988). One of the papers that works with multiple disruptions in vehicle routing problems was developed by Wang et al. (2011). They proposed a model to handle different types of disruptions in VRP with time windows. To model the disruptions, they transform them into requirements for the time windows of each customer, and by using this modification the optimal starting time for vehicles is determined.

Some approximations have been developed to approximate the problem of disruptions in distribution networks, using the specific case of vehicle routing problems. A specific approximation was developed by Mu et al. (2011). In this case, they studied the problem of vehicle breakdowns during deliveries and the necessity to reoptimize the solution obtained. To solve this problem in a reasonable amount of time, they developed two tabu search algorithms that allowed them to find near-optimal solutions in a reasonable amount of time. The algorithms proposed and the problem studied do not contain random parameters or uncertainty.

Real-time optimization or the online vehicle routing problem was also studied by Ng et al. (2017), and this problem is characterized by traffic congestion. The main idea is to reduce the risk of late delivery. The authors developed a multiple colonies artificial bee colony algorithm. They tested the proposed algorithm using real data and instances from the literature.

Some extensions of the VRP under disruptions have been studied in the literature. The location routing problem with disruption has been studied by Zhang et al. (2015), who consider a multi-product problem

with stochastic demand in which the supply of the product is randomly disrupted in distribution centers. To tackle this problem, the authors presented a bi-objective mixed-integer nonlinear programming model with one objective of minimizing the overall costs and the second objective was to minimize the total failure costs related to disrupted distribution centers. That leads to the reliability of the supply chain network, and they also developed a multi-objective simulated annealing algorithm combined with the Taguchi method to tune the parameters. A recent development of this problem was by Rayat et al. (2017). In this problem, also the depots were randomly disrupted, and they developed a scenario-based mixed-integer programming model to optimize the reinsertion when depots suffer disruptions, also combining neighborhood search and simulated annealing algorithms. Several other articles have tried to approximate this problem (Mohammadi et al., 2016; Rahimi et al., 2016; Zhalechian et al., 2016; Ahmadi-Javid & Seddighi, 2013; Asl-Najafi et al., 2015). Nevertheless, this review enables us to determine that this is the first approximation of the inventory routing problem under disruptions.

2. Mathematical model

For our approximation we have adapted the formulation proposed by Archetti et al. (2007) and we have introduced the time windows constraints according to the model developed by Kaligari and Guerrero (2015).

3.1 Notation

3.1.1 Sets

T	Set of period time
V	Set of nodes
V_p	Set of customers $=\{2, \dots, n\}$

3.1.2 Parameters

r_t	Amount of product available at the depot for each time period t
dt_i^t	Demand for each customer i , for each time period t
L_i	Lower bound of inventory for each customer i
C_i	Upper bound of inventory for each customer i
c_{ij}	Travel cost between customer i and customer j
Q_k	Capacity of each vehicle
a_i	Lower bound for the time windows for each customer i
b_i	Upper bound for the time windows for each customer i
s_i	Service time for each customer i
M_{ij}	Upper case M used for the time windows for each pair of customers i, j
h_i	Inventory holding cost for each customer i

3.1.3 Variables

I_i^t	Inventory levels for each node i for each time period t
X_{ij}^{kt}	=1 if the arc i, j is traversed by vehicle k in a time period t
q_i^{kt}	Amount of product delivery to customer i by vehicle k in a time period t
Y_i^{kt}	=1 if a customer i is visited by vehicle k in a time period t
e_i^{kt}	Time of beginning of a service for a customer i in a time period t for a specific vehicle k

3.2 Model formulation

The mathematical formulation for the problem is presented as follows:

The objective function given in Eq. (1) is used to minimize the total cost of the model. The first part of the objective function comprises the inventory holding cost at the depot and the customers. The second part of the objective function contains the routing cost.

$$\min \sum_{i \in V} \sum_{t \in T} h_i I_i^t + \sum_{i \in V} \sum_{j \in V} \sum_{k \in K} \sum_{t \in T} c_{ij} X_{ij}^{kt} \quad (1)$$

$$I_i^t = I_i^{t-1} + \sum_{k \in K} q_i^{kt} - dt_i^t \quad \forall i \in Vp, t \in T \quad (2)$$

$$I_i^t \geq L_i \quad \forall i \in Vp, t \in T \quad (3)$$

$$I_i^t \leq C_i \quad \forall i \in Vp, t \in T \quad (4)$$

$$\sum_{k \in K} q_i^{kt} \leq C_i - I_i^{t-1} \quad \forall i \in Vp, t \in T \quad (5)$$

$$\sum_{k \in K} q_i^{kt} \leq C_i \sum_{j \in V} \sum_{k \in K} X_{ij}^{kt} \quad \forall i \in Vp, t \in T \quad (6)$$

$$\sum_{k \in K} q_i^{kt} \leq Qk \quad \forall t \in T, k \in K \quad (7)$$

$$q_i^{kt} \leq Y_i^{kt} C_i \quad \forall i \in Vp, t \in T, k \in K \quad (8)$$

$$\sum_{j \in V} X_{ij}^{kt} = \sum_{j \in V} X_{ji}^{kt} \quad \forall i \in Vp, t \in T, k \in K \quad (9)$$

$$\sum_{j \in V} X_{ij}^{kt} = Y_i^{kt} \quad \forall i \in Vp, t \in T, k \in K \quad (10)$$

$$\sum_{j \in Vp} X_{1j}^{kt} \leq 1 \quad \forall k \in K, t \in T \quad (11)$$

$$\sum_{k \in K} Y_i^{kt} \leq 1 \quad \forall i \in Vp, t \in T \quad (12)$$

$$e_i^{kt} + s_i + c_{ij} - e_j^{kt} \leq (1 - X_{ij}^{kt}) M_{ij} \quad \forall i \in V, j \in Vp, k \in K, t \in T, i \neq j \quad (13)$$

$$e_i^{kt} \geq a_i \quad \forall i \in Vp, k \in K, t \in T \quad (14)$$

$$e_i^{kt} \leq b_i \quad \forall i \in Vp, k \in K, t \in T \quad (15)$$

$$I_i^t \geq 0 \quad \forall i \in Vp, t \in T \quad (16)$$

$$X_{ij}^{kt}, Y_i^{kt} \in \{0,1\} \quad \forall i, j \in Vp, i \neq j, t \in T, k \in K \quad (17)$$

$$q_i^{kt} \geq 0 \quad \forall i \in Vp, t \in T, k \in K \quad (18)$$

$$e_i^{kt} \geq 0 \quad \forall i \in Vp, t \in T, k \in K \quad (19)$$

Eq. (2) ensures the balance inventory constraints. Constraints (3) and (4) ensure that for each customer the inventory levels cannot exceed the minimum and maximum level respectively. Constraints (5) and (6) model the quantities of product that can be delivered to each customer. Eq. (7) guarantees that the amount of product delivered in a vehicle does not exceed its capacity. With constraint (8) the consistency between the amount of product delivered and the activation of a route is given. Eqs. (9-10) represent the flow conservation and activation constraints respectively. Constraints (11) and (12) ensure that a vehicle only can begin one route from the depot. Eqs (13-15) model the time windows constraints. Finally, Eqs. (16-19) define the type of variable.

3. Modeling framework

For solving the problem considered before considering the dynamism and disruptions over the network, a dynamic programming method with finite stages is used. We have determined a probability of going

into a disruption for each link in the network and it varies for each period of time. For each node and each period of time the decision is made on which node to travel to next. The main objective is to satisfy the customer’s requirements by minimizing the overall costs considering the natural constraints and the disruptions over the network. For this situation, we have included the following variables:

Cf_{ij}	Failure cost between each pair of nodes i,j
PS_{ij}	Probability without disruption between each pair of nodes i,j
PC_{ij}	Probability with disruption between each pair of nodes i,j
GS_{ij}	Expected profit without disruption between each pair of nodes i,j
GC_{ij}	Expected profit with disruption between each pair of nodes i,j
GES_i^t	Expected profit without disruption in node i in the time period t
GEC_i^t	Expected profit with disruption in node i in the time period t
GEM_i^t	Maximum expected profit in node i in the time period t

Each arc (i, j) has an associated probability of success, which means that there is a probability that a vehicle traveling through this arc will reach its destination without any inconvenience, and this probability is affected by different factors such as traffic or damage. Dynamic programming with finite stages is based on the final steps of the process, in this case the time periods, as well as the policies used to make the best decision. Two policies are developed, the first considering the scenarios with the disruption, and the other one ignoring these scenarios, as mentioned in the justification for the model.

Each policy has a transition matrix, as well as a matrix of profits, and these matrices can be defined respectively as follows:

$$PS_{ij} = 0 \quad \forall i \in V, j \in V, i = j \quad (22)$$

$$PS_{ij} = \frac{(\sum_{j \in V}^{MAX} c_{ij}) - c_{ij}}{\sum_{j \in V}^{MAX} c_{ij} - c_{ij}} \quad \forall i \in V, j \in V, i \neq j \quad (23)$$

$$GS_{ij} = PS_{ij} \sum_{j \in V} c_{ij} \quad \forall i \in V, j \in V \quad (24)$$

$$cf_{ij} = \frac{c_{ij}}{Pe_{ij}} \quad \forall i \in V, j \in V \quad (25)$$

$$PC_{ij} = 0 \quad \forall i \in V, j \in V, i = j \quad (26)$$

$$PC_{ij} = \frac{(\sum_{j \in V}^{MAX} cf_{ij}) - cf_{ij}}{\sum_{j \in V}^{MAX} cf_{ij} - cf_{ij}} \quad \forall i \in V, j \in V, i \neq j \quad (27)$$

$$GC_{ij} = PC_{ij} \sum_{j \in V} c_{ij} \quad \forall i \in V, j \in V \quad (28)$$

Eqs. (20-22) correspond to the scenario without disruptions. These equations present the way in which the transition matrix between the nodes is calculated by using the distance between them. Also, the profit matrix is presented.

On the other hand, Eqs. (23-26) take into account scenarios with disruptions. In these new equations a new term is introduced ($cf_{i,j}$), which is used to calculate the profit matrix with disruptions. This new term takes into account the distance in proportion to the success probability for each arc (i,j) , hence these probabilities are used to take the decisions.

The dynamic programming model with finite stages begins from the most distant period and decreases in each iteration until reaching the first period. That is, if the planning horizon is 3, the first iteration will be for the third period, then for the second, and finally for the first, taking decisions in descending stages

first for the first period, then for the second and finally the third, reaching the defined planning horizon. In this way and using the equations previously defined, the following equations are defined:

$$\forall t = P$$

$$GES_i^P = \sum_{j \in V} (PS_{ij} GS_{ij}) \quad \forall i \in V \quad (27)$$

$$GEC_i^P = \sum_{j \in V} (PC_{ij} GC_{ij}) \quad \forall i \in V \quad (28)$$

$$if \begin{cases} GES_i^P > GEC_i^P \rightarrow GEM_i^P = GES_i^P \\ GEC_i^P \geq GES_i^P \rightarrow GEM_i^P = GEC_i^P \end{cases} \quad \forall i \in V \quad (29)$$

$$\forall t = \{P - 1, \dots, 1\}$$

$$GES_i^t = \sum_{j \in V} (PS_{ij} GEM_i^{t+1}) \quad \forall i \in V \quad (30)$$

$$GEC_i^t = \sum_{j \in V} (PC_{ij} GEM_i^{t+1}) \quad \forall i \in V \quad (31)$$

$$if \begin{cases} GES_i^t > GEC_i^t \rightarrow GEM_i^t = GES_i^t \\ GEC_i^t \geq GES_i^t \rightarrow GEM_i^t = GEC_i^t \end{cases} \quad \forall i \in V \quad (32)$$

Eq. (27) presents the profit without disruption when the time period is equal to P, while Equation (28) gives the profit including disruptions when the time period is equal to P. Equation (29) represents the maximum expected profit for each node in the time period P, which is the maximum between GES_i^P and GEC_i^P . Finally, Eqs. (30-32) represent the same as Eqs. (27-29) but the time period varies between $t = P-1 \dots t = 1$.

3.1. Dynamic components

Another factor used in this work is the dynamism of the nodes. We have added the dynamism of the nodes by using the deterministic demand. Most of the articles in the literature deal with the problem of a finite and known number of nodes (or customers) with a known demand for each time period. In our approach, we have included nodes that can be dynamic by using a probability function that determines for each time period if a node is active or inactive. This means that the demand for each time period is also dynamic and is determined by the number of nodes in each time period and the demand of each one of them that is deterministic.

To model this situation a vector Dt_i^t is used. This vector contains the demand in a node in a time period. If a node is active in a specific time period, then $Dt_i^t = dt_i$; otherwise $Dt_i^t = 0$. In other words, if the node is active in a specific time period the demand of the node will be equal to its deterministic demand, and if the node is not active the demand will be zero.

Considering this new feature, some constraints and conditions are added to the mathematical model presented in the previous section. We must guarantee that if a customer is inactive in a time period the delivery quantity to this customer must be zero. This can be guaranteed by using Eq. (33).

$$q_i^{kt} = 0 \quad \forall i \in V, k \in K, t \in T \mid Dt_i^t = 0 \quad (33)$$

Besides, Eq. (13) must be replaced by the following constraints:

$$if \begin{cases} Dt_i^t > 0 \rightarrow e_i^{kt} + s_i + c_{ij} - e_j^{kt} \leq (1 - X_{ij}^{kt}) M_{ij} \\ Dt_i^t = 0 \rightarrow e_i^{kt} + c_{ij} - e_j^{kt} \leq (1 - X_{ij}^{kt}) M_{ij} \end{cases} \quad \forall i \in V, j \in Vp, k \in K, t \in T, i \neq j \tag{32}$$

Finally, with this new feature a modification to the dynamic programming method with finite stages is required, because the transition matrices and the profit matrix will vary for each time period and will depend on the active nodes.

Eqs. (20-32) will repeat for each time period taking into account that the active nodes vary for each time period, hence the size of matrices and vectors will vary and the size depends directly on the number of active nodes. Eqs. (27-32) are repeated in each time period but they have a modification because they must iterate from the time period P to the period t so that the decision can be made if the profit matrix and the nodes have remained. In other words, if the time period is P=3, then Equations (20) to (32) will repeat for each $t = \{1, 2, 3\}$. Nevertheless, Equations (27) to (32) must iterate in the following way: for $t=1$ three iterations must be done beginning from the third period to the first period, taking the related decisions for each node according to the maximum expected profit GEM_{it} . When $t=2$, two iterations must be done, for $t=2$ and $t=3$, where the maximum expected profit GEM_{it} is also calculated. Finally, when $t=3$ only one iteration must be done and the policy selection must be done.

4. Computational results and analysis

4.1. Problem instances

We have generated instances in order to test our approach by using the classic instances of the Inventory Routing Problem proposed by Coelho and Laporte (see <http://www.leandro-coelho.com/instances/inventory-routing/>) and the time windows are generated using the method proposed by Kalgari et al. (2015). In addition to the information generated before, we have generated for each customer two random numbers necessary to our approach. All these data are required as inputs for our approximation. Due to our approach changing dynamically, the decisions also change dynamically for each iteration, as mentioned earlier.

The success probability, as abovementioned, depends on a set of factors such as traffic, which may influence the probability of a delivery. Since the proposed method uses random numbers, two special situations can be presented. The first one is that the success probability of delivery is equal to 0, which means there is no direct connection between a pair of nodes; otherwise the probability is equal to 1, which means the connection does not present any failure or disruption. To clarify, the success probability between all the nodes cannot be 0 because it would represent an infeasible problem; and in the other case, if all the success probabilities are equal to 1, the problem is reduced to the classic deterministic inventory routing problem.

4.2. Results

We will describe step by step with a single instance and then we will present the average results obtained. We will first present the results on an instance with a maximum of 6 customers for one to 3 periods. To illustrate our procedure, we first present the distances without disruptions in Table 1 and distances with disruptions in Table 2.

Table 1
Distances without disruptions

i	j					
	1	2	3	4	5	6
1	0	101	443	22	228	381
2	101	0	342	123	293	316
3	443	342	0	465	445	294
4	22	123	465	0	218	391
5	228	293	445	218	0	609
6	381	316	294	391	609	0

Table 2

Distances with disruptions

<i>i</i>	<i>i</i>					
	1	2	3	4	5	6
1	0	121,982	447,777	22,6069	311,322	487,942
2	105,916	0	404,096	125,802	338,254	317,033
3	555,055	406,451	0	473,538	582,938	411,794
4	27,1891	140,588	474,474	0	223,912	398,842
5	302,76	333,228	573,174	231,796	0	657,971
6	489,989	338,326	298,664	400,332	754,207	0

Table 3 presents the policies for each scenario in each time period. The decisions are related with taking or not taking into account the disruption scenario for each period for each node.

Table 3

Optimal Policy

<i>T</i>	<i>i</i>	Decision
1	1	The scenario with disruption must be used
1	2	The scenario with disruption should not be used
1	3	The scenario with disruption should not be used
1	4	The scenario with disruption should not be used
1	5	The scenario with disruption should not be used
1	6	The scenario with disruption must be used
2	1	The scenario with disruption must be used
2	2	The scenario with disruption should not be used
2	3	The scenario with disruption should not be used
2	4	The scenario with disruption should not be used
2	5	The scenario with disruption should not be used
2	6	The scenario with disruption must be used
3	1	The scenario with disruption must be used
3	2	The scenario with disruption should not be used
3	3	The scenario with disruption should not be used
3	4	The scenario with disruption should not be used
3	5	The scenario with disruption should not be used
3	6	The scenario with disruption must be used

Also, the number of active nodes per period and their demands are presented in Table 4.

Table 4

Number of active nodes and their demand

<i>t</i>	<i>i</i>	<i>Dt^t</i>
1	3	35
1	4	58
1	5	58
1	6	11
2	2	65
2	3	35
2	4	58
2	5	58
2	6	11
3	3	35
3	5	58
3	6	11

With the results presented in Table 3 we can determine in which period the success probability must be an active constraint. Once these results are determined, we can next obtain the following results about

the vehicles used for each service, the beginning of the service and inventory levels (Table 5, Table 6 and Table 7).

Table 5
Vehicles used for each node for each period

<i>i</i>	<i>j</i>	<i>K</i>	<i>t</i>
1	2	2	1
2	5	2	1
5	1	2	1
1	2	1	2
2	5	1	2
5	1	1	2
1	3	2	2
3	6	2	2
4	1	2	2
6	4	2	2
1	2	2	3
2	5	2	3
5	1	2	3

Table 6
Vehicles used for each service in each time period

<i>I</i>	<i>t=1</i>		<i>t=2</i>		<i>t=3</i>	
	<i>k=1</i>	<i>k=2</i>	<i>k=1</i>	<i>k=2</i>	<i>k=1</i>	<i>k=2</i>
1	0	0	0	0	0	0
2	1095	1095	1095	1095	1095	1095
3	1437	1437	1447	1407	1437	1437
4	3010	3010	3010	3010	3010	3010
5	1399	1399	1399	1399	1399	1399
6	1711	1711	1711	1711	1711	1711

Table 7
Inventory levels for each customer in each time period

<i>i</i>	<i>t=1</i>	<i>t=2</i>	<i>t=3</i>
1	679	634	769
2	130	130	130
3	35	35	0
4	0	0	0
5	14	14	14
6	0	11	0

We have determined three types of policies to compare and verify the results over the different instances. The first policy consists in executing the algorithm without using the success probability, the second one is without using the dynamism of the nodes and the third one is without using both types of dynamisms.

In Table 8 the average results are presented for instances with three time periods and varying the number of customers between 5 and 35.

Table 8

Average results with 3 period times

<i>5 customers</i>	Original Algorithm	Modification 1	Modification 2	Modification 3
<i>Objective Function</i>	2579.38	2579.38	3328.9	3328.9
<i>% Demand satisfied</i>	44.54%	44.54%	50.37%	50.37%
<i>AbsIn10</i>	Original Algorithm	Modification 1	Modification 2	Modification 3
<i>OBJECTIVE FUNCTION</i>	4781.35	4385.01		5220.94
<i>% Demand satisfied</i>	31.52%	31.52%		37.27%
<i>AbsIn15</i>	Original Algorithm	Modification 1	Modification 2	Modification 3
<i>OBJECTIVE FUNCTION</i>	5810.16	5757.2		6495.73
<i>% Demand satisfied</i>	25.78%	25.23%		33.33%
<i>AbsIn25</i>	Original Algorithm	Modification 1	Modification 2	Modification 3
<i>OBJECTIVE FUNCTION</i>	7561.01	7549.01	9472.47	9121.17
<i>% Demand satisfied</i>	24.61%	24.61%	33.33%	33.33%
<i>AbsIn30</i>	Original Algorithm	Modification 1	Modification 2	Modification 3
<i>OBJECTIVE FUNCTION</i>	11968.4	11968.4	12892.3	12801.8
<i>% Demand satisfied</i>	17.71%	20.84%	34.89%	33.33%
<i>AbsIn35</i>	Original Algorithm	Modification 1	Modification 2	Modification 3
<i>OBJECTIVE FUNCTION</i>	11968.4	11225.3		11757.7
<i>% Demand satisfied</i>	13.30%	14.36%		32.59%

It can be observed that our approach can find solutions close to those policies that relax some features used by our work, even when the uncertainty is higher in our approach. Also by our approach we can guarantee that the demand satisfied is close to those without using less uncertain policies. In those cases in which we cannot obtain the optimal solution, this is mainly because the time of iteration exceeded the maximum time allowed.

In Table 9 the average results of instances with six time periods is presented.

Table 9

Average results with 6 time periods

<i>5 Customers</i>	Original Algorithm	Modification 1	Modification 2	Modification 3
<i>OBJECTIVE FUNCTION</i>	4781.35	5328		6643.6
<i>% Demand satisfied</i>	28.14%	28.14%		40.19%
<i>10 Customers</i>	Original Algorithm	Modification 1	Modification 2	Modification 3
<i>OBJECTIVE FUNCTION</i>	10420.3	10010.3		11277.6
<i>% Demand satisfied</i>	33.44%	33.44%		42.37%
<i>15 Customers</i>	Original Algorithm	Modification 1	Modification 2	Modification 3
<i>OBJECTIVE FUNCTION</i>	10420.3	10010.3		11277.6
<i>% Demand satisfied</i>	33.44%	33.44%		42.37%
<i>20 Customers</i>	Original Algorithm	Modification 1	Modification 2	Modification 3
<i>OBJECTIVE FUNCTION</i>	17718	17718	18692.3	
<i>% Demand satisfied</i>	33.80%	33.80%	42.40%	
<i>25 Customers</i>	Original Algorithm	Modification 1	Modification 2	Modification 3
<i>OBJECTIVE FUNCTION</i>	17915.4	17487.1	19795.6	18726.7
<i>% Demand satisfied</i>	33.52%	32.42%	42.29%	41.60%

5. Conclusions

In this paper, we have presented an inventory routing problem under disruption networks. To tackle this problem, we have used a stochastic dynamic programming with finite stages approximation. We have tested different policies combining disruption and dynamism of the nodes, finding that the full algorithm works appropriately over the full instances. We have also observed that the performance when combining disruptions with dynamism of nodes could obtain solutions close to those that only used single policies or worked with the full problem. Another advantage of using the information of each scenario is that it leads to a significant decrease of the complexity of the problem.

The literature review allows us to conclude that some studies have tried to develop algorithms for the online version or using disruptions in VRP and ILRP, but our proposal has not been developed in the literature before. Future works will be focused on describing the probabilities of disruptions and will consider modifications to the traditional Inventory Routing Problem, possibly including backorders and other types of characteristics.

Acknowledgments

We thank Fair Isaac Corporation (FICO) for providing us with Xpress-MP licenses under the Academic Partner Program subscribed with Universidad Distrital Francisco Jose de Caldas (Bogotá, Colombia).

References

- Ahmadi-Javid, A., & Seddighi, A. H. (2013). A location-routing problem with disruption risk. *Transportation Research Part E: Logistics and Transportation Review*, 53, 63–82. <https://doi.org/10.1016/j.tre.2013.02.002>
- Archetti, C., Bertazzi, L., Laporte, G., & Speranza, M. G. (2007). A Branch-and-Cut Algorithm for a Vendor-Managed Inventory-Routing Problem. *Transportation Science*, 41(3), 382–391.
- Asl-Najafi, J., Zahiri, B., Bozorgi-Amiri, A., & Taheri-Moghaddam, A. (2015). A dynamic closed-loop location-inventory problem under disruption risk. *Computers & Industrial Engineering*, 90, 414–428.
- Azadeh, A., Elahi, S., Hosseinabadi Farahani, M., & Nasirian, B. (2017). A genetic algorithm-Taguchi based approach to inventory routing problem of a single perishable product with transshipment. *Computers & Industrial Engineering*, 104, 124–133.
- Coelho, L. C., Cordeau, J.-F., & Laporte, G. (2012). Consistency in multi-vehicle inventory-routing. *Transportation Research Part C: Emerging Technologies*, 24, 270–287.
- Coelho, L. C., & Laporte, G. (2013). The exact solution of several classes of inventory-routing problems. *Computers and Operations Research*, 40(2), 558–565. <https://doi.org/10.1016/j.cor.2012.08.012>
- Federgruen, A., & Zipkin, P. (1984). A Combined Vehicle Routing and Inventory Allocation Problem. *Operations Research*, 32(5), 1019–1037.
- Franco, C., & Figueroa-García, J. C. (2016). A column generation-based algorithm for solving combined inventory and routing problems. *Ingeniare*, 24(2), 305–313.
- Ng, K.K.H., Lee, C.K.M., Zhang, S.Z., Wu, K., & Ho, W. (2017). A multiple colonies artificial bee colony algorithm for a capacitated vehicle routing problem and re-routing strategies under time-dependent traffic congestion. *Computers & Industrial Engineering*, 109, 151–168.
- Kaligari, E., & Guerrero, W. (2015). Métodos de optimización para el problema de ruteo de vehículos con inventarios y ventanas de tiempo duras. *Revista Ingeniería Industrial*, 14(3), 31–49.
- Li, J.-Q., Borenstein, D., & Mirchandani, P. B. (2007). A decision support system for the single-depot vehicle rescheduling problem. *Computers & Operations Research*, 34(4), 1008–1032.
- Mohammadi, M., Tavakkoli-Moghaddam, R., Siadat, A., & Dantan, J.-Y. (2016). Design of a reliable logistics network with hub disruption under uncertainty. *Applied Mathematical Modelling*, 40(9–10), 5621–5642.
- Mu, Q., Fu, Z., Lysgaard, J., & Eglese, R. (2011). Disruption management of the vehicle routing problem with vehicle breakdown. *Journal of the Operational Research Society*, 62(62), 742–749.
- Psaraftis, H. (1988). Vehicle Routing: Methods and Studies. In *Dynamic Vehicle Routing Problems*. B.V: North Holland.: Elsevier Science Publishers.
- Rahimi, Y., Tavakkoli-Moghaddam, R., Mohammadi, M., & Sadeghi, M. (2016). Multi-objective hub network design under uncertainty considering congestion: An M/M/c/K queue system. *Applied Mathematical Modelling*, 40(5–6), 4179–4198.
- Rayat, F., Musavi, M., & Bozorgi-Amiri, A. (2017). Bi-objective reliable location-inventory-routing problem with partial backordering under disruption risks: A modified AMOSA approach. *Applied Soft Computing*, 59, 622–643.
- Sever, D., Dellaert, N., Van Woensel, T., & De Kok, T. (2013). Dynamic shortest path problems: Hybrid routing policies considering network disruptions. *Computers and Operations Research*, 40(12), 2852–

2863.

- Teodorovic, D., & Guberinic, S. (1984). Optimal dispatching strategy on an airline network after a schedule perturbation. *European Journal of Operational Research*, *15*(2), 178–182.
- Wang, X., Ruan, J., Shang, H., & Ma, C. (2011). A Combinational Disruption Recovery Model for Vehicle Routing Problem with Time Windows. In *Intelligent Decision Technologies* (Vol. 10, pp. 3–12).
- Xiangpei, H., & Lijun, S. (2012). Knowledge-based modeling for disruption management in urban distribution. *Expert Systems with Applications*, *39*, 906–916.
- Zhang, Y., Qi, M.-Y., Lin, W.-H., & Miao, L.-X. (2015). A metaheuristic approach to the reliable location routing problem under disruptions. *Transportation Research Part E*, *83*, 90–110.
- Yu, G., & Qi, X. (2004). *Disruption Management*. World Scientific. <https://doi.org/10.1142/5632>
- Zhalechian, M., Tavakkoli-Moghaddam, R., Zahiri, B., & Mohammadi, M. (2016). Sustainable design of a closed-loop location-routing-inventory supply chain network under mixed uncertainty. *Transportation Research Part E: Logistics and Transportation Review*, *89*, 182–214.



© 2018 by the authors; licensee Growing Science, Canada. This is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC-BY) license (<http://creativecommons.org/licenses/by/4.0/>).