Computers and Chemical Engineering 80 (2015) 199–210

Contents lists available at ScienceDirect

Computers and Chemical Engineering

journal homepage: www.elsevier.com/locate/compchemeng

An incremental approach using local-search heuristic for inventory routing problem in industrial gases

Tejinder Singh[a,*], Jeffrey E. Arbogast[a], Nicoleta Neagu[b]

[a] Air Liquide, Delaware Research and Technology Center, Newark, DE 19702, United States
[b] Air Liquide, Paris-Saclay Research Center, 78350 Jouy-en-Josas, France

ARTICLE INFO

Article history:
Received 30 July 2014
Received in revised form 5 May 2015
Accepted 28 May 2015
Available online 7 June 2015

Keywords:
Vendor managed inventory
Liquefied gases
Industrial gases distribution
Inventory routing problem
Bulk gas distribution
Local-search heuristic

ABSTRACT

In this paper we solve the inventory routing problem (IRP) occurring in industrial gas distribution where liquefied industrial gases are distributed to customers that have cryogenic tanks to store the gases on-site. We consider a multi-period inventory routing problem with multiple products assuming deterministic demand rates and the proposed model is formulated as a linear mixed-integer program. We propose an incremental approach based on decomposing the set of customers in the original problem into sub-problems. The smallest sub-problem consists of the customer that needs to be delivered most urgently along with a set of its neighbors. We solve each sub-problem with the number of customers growing successively by providing the solution of the previously solved sub-problem as an input. Each sub-problem is then solved with a randomized local-search heuristic method. We also propose an objective function that drives the local-search heuristics toward a long-term optimal solution. The main purpose of this paper is to develop a solution methodology appropriate for large-scale real-life problem instances particularly in industrial gas distribution.

1. Introduction

The inventory routing problem (IRP) is a challenging problem that arises in various real-life distribution systems. It involves managing inventory and vehicle routing simultaneously where the vendor is responsible for the replenishment of a set of geographically dispersed customers (Campbell et al., 1998). These customers have demands for different products spread over time, and are entitled to keep local inventory. Deliveries are usually made using a fleet of capacitated trucks. The usual vehicle routing problem (VRP) is a much less complex problem than the IRP problem (Bertazzi and Speranza, 2012). In the VRP, routing decisions are made to fulfill, by the end of the day, fixed orders placed by the customers. In the IRP, the routing decisions are dictated by the anticipated inventory behavior of the customers, which is itself driven by their daily demand patterns. Given the customers' inventory data and information on the customers' demands, the logistics analyst must consequently make following important decisions over a given planning horizon:

- When to visit each customer during the planning horizon
- How much to deliver to each customer on each visit
- How to combine customer visits into vehicle routes

In an industrial setting, the IRP can be applied to various distribution systems. Traditionally, researchers and practitioners have focused on applications to the maritime, automotive and supermarket industries, for example (Campbell and Savelsbergh, 2004). In this paper, we focus on the IRP problem in industrial bulk gas distribution for a finite-horizon scenario (e.g., 2 weeks). We assume that the customer's demand is known (deterministic) and is also dynamic. In industrial gas distribution, liquefied gases [e.g., oxygen (O_2), nitrogen (N_2), argon (Ar), carbon dioxide (CO_2), and hydrogen (H_2)] are transported from production plants to cryogenic storage tanks placed at customer sites using cryogenic trailers. This is primarily done through a VMI system (You et al., 2011). In a VMI system, it is the vendor's responsibility to prevent customer stock-out of the product and avoid penalty consequences. However, there are generally a relatively small percentage of customers that call the vendor to place their orders (non-VMI/call-in customers). There are some features of IRP for bulk gas distribution which are unique compared to the classical IRP model. The IRP for bulk gas distribution includes inventory level constraints for both suppliers and customers. Unlike typical IRP systems, a cryogenic trailer cannot deliver different products in a single delivery. There can be other

* Corresponding author. Tel.: +1 713 896 2164.
E-mail address: tejinder.singh@airliquide.com (T. Singh).

http://dx.doi.org/10.1016/j.compchemeng.2015.05.023

specific routing constraints (e.g., some customers should always be delivered first in a trip).

In this paper, we refer to the combination of a driver, a tractor and a cryogenic trailer as a vehicle. The liquefied gas products are stored at the vendor's production plants and also at the customer sites in cryogenic tanks of various sizes. An industrial gas bulk distribution schedule is generally executed for a smaller time horizon (e.g., 1–3 days), but the planning is done over a longer horizon (e.g., 7–14 days). This study assumes that the scheduling problem is solved daily at a given time with the schedule actually implemented only for the first day of the horizon. Each schedule consists of a set of shifts. A shift is defined as a sequence of activities performed by a single driver within a single working period. In a VMI context with some call-in orders, there are three primary objectives:

- Satisfy all call-in orders
- Maintain all VMI customers above safety levels
- Minimize logistics ratio which is defined as the overall cost per unit product delivered

The uncertainty in scheduling generally arises from many factors including unanticipated customer orders, un-forecasted changes in VMI customer demand, lack of product availability from default sources, driver unavailability, and tractor/trailer unavailability due to breakdowns or unplanned maintenance. In this paper, uncertainty in the parameters is not considered.

This paper assumes that the vendor has telemetry access to monitor storage tank levels at VMI customer sites. Based on the tank levels, historical consumption rates are calculated which are used to forecast the demand during the planning horizon. In practice, logistics analysts plan the deliveries to customers for the scheduling horizon. Scheduling horizon is the period for which the planned deliveries are executed. The objective of this paper is to solve the industrial gas IRP problem within a reasonable time-frame (less than half-an hour) and generate a useful delivery schedule.

2. Literature review

Federgruen and Simchi-Levi (1995) provides the motivations for the IRP and develop a framework that distinguishes two variants of the IRP: the single period model, with stochastic demand, and the infinite horizon model, with deterministic demand rate. Two articles, namely Federgruen and Zipkin (1984) and Anily and Federgruen (1990) illustrate these classifications. Though this classification gives an initial overview of the different aspects of the IRP, it overlooks several approaches that do not fit this description, such as single period models with deterministic demand, multi-period models, and infinite horizon models with stochastic demand. A second attempt to classify the IRP can be found in Baita et al. (1998). In this review paper, IRP is defined as a class of problems having the following aspects in common: routing, inventory, and dynamic behavior (repeated decisions have to be made). Within this class of problems, a classification framework is proposed that takes into account all of the characteristics of the different approaches encountered in the literature: topology of the problem, number of items considered, type of demand considered, type of decision to be taken, constraints considered, objectives sought, costs considered and solution approach proposed.

In this paper, we consider a multi-period finite horizon model with deterministic demand specific to industrial gases distribution. We highlight few important papers that deal with a similar problem. Aghezzaf et al. (2006), Archetti et al. (2007), Campbell and Savelsbergh (2004), Chien et al. (1989), Yu et al. (2008), and Benoist et al. (2011) all study a multi-period IRP where the decisions are carried out over a finite horizon. We should note that most of the studies dealing with the infinite horizon problem use a distribution policy that is similar to the fixed partition policy (first introduced in Anily and Federgruen, 1993), direct deliveries, order-up-to level policy and zero-inventory ordering (Bertazzi et al., 2002; Chan et al., 1998). Refer to Coelho et al. (2014) for the history of IRP and a recent detailed review of different exact and heuristic approaches which can be used to solve wide variety of IRP models.

To solve the IRP, most published research has focused on heuristic solution approaches due to the problem's NP-hard complexity. Frequently, the integrated problem is decomposed into sub-problems (Campbell and Savelsbergh, 2004) which are solved by approximate or exact methods (i.e. Branch and Cut, Column Generation). In some cases, heuristic methods are applied to the sub-problems in order to identify upper and lower bounds. In some IRP papers, integrated and iterative approaches are provided and the effectiveness of integrating routing and inventory decisions in the models is evaluated. Others have proposed heuristic methods to be compared with approaches used in industry (Campbell et al., 2002; Dror and Ball, 1987).

We next provide a list of more recent papers using heuristic approaches to solve IRP. Liu and Lee (2011) solve the IRP with time windows using tabu search to improve a given initial solution on the basis of the average supply chain cost consisting of transportation cost, time window violation penalty cost and inventory cost. Archetti et al. (2010) combine tabu search with mixed integer programming models to solve an IRP with a multi-period horizon where a supplier uses a single vehicle to serve a set of customers having limited capacity to hold inventory. Abdelmaguid et al. (2009) develop constructive and improvement heuristics to obtain an approximate solution for the IRP with backlogging. This considers a case having a depot with an infinite supply of a single product satisfying deterministic demand at each customer. Huang and Lin (2010) introduces a modified ant colony optimization for the IRP having multi-item replenishment with uncertain demand. A three-phase heuristic to solve multiple-product IRP is developed by Cordeau et al. (2015). In the first phase, replenishment plans are determined by a Lagrangian-based method, sequencing of the planned deliveries is performed in the second phase, and the third phase incorporates planning and routing decisions into a mixed-integer linear programming model. Moin et al. (2011) address a finite-horizon, multi-period, multi-suppliers, and multi-products IRP problem with a fleet of homogenous vehicles and propose a hybrid genetic algorithm based on the allocation first route second strategy to solve medium and small-sized IRP problems.

More recent research papers that deal with optimal distribution of industrial gases supply-chains are by You et al. (2011), Benoist et al. (2011), Ellis et al. (2014) and Marchetti et al. (2014). You et al. (2011) focus on strategic decisions with long-term planning horizon using decomposition and continuous approximation approaches. Similarly, Ellis et al. (2014) optimize the strategic-level decision of allocating bulk gas tanks to customer sites. Marchetti et al. (2014) assess the benefits of optimal coordination of production and distribution in industrial gas supply-chains. Benoist et al. (2011) propose a method to solve a real-life IRP for operational scheduling using a randomized local-search heuristic for short-term planning horizon. This method uses a novel surrogate objective function based on long-term lower bounds and report savings exceeding 20% on average compared to solutions built by expert logistics analysts. This method applies a greedy algorithm to generate an initial solution which is then improved upon by the local-search heuristic. In this paper, we apply the same local-search methodology to generate optimal solutions without using greedy algorithm for initial solutions. The local-search heuristic approach is interesting to solve real-life IRP scheduling decisions due to its

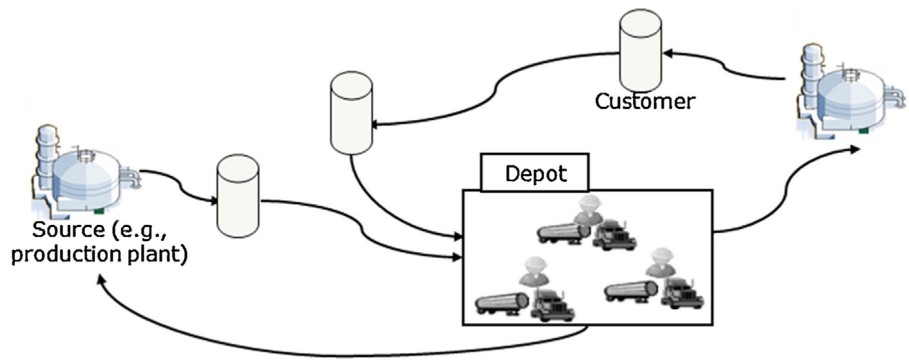

Fig. 1. Industrial gas distribution.

close resemblance to logistics analysts' manual scheduling optimization approach. The main contributions of this paper are as follows:

- The randomized local-search heuristic considered by Benoist et al. (2011) is improved by using a new long-term objective function.
- A novel incremental approach is proposed to optimize the IRP problem by solving many sub-problems iteratively. The solution of previously solved sub-problem is used as an initial solution for the next sub-problem.

The paper is organized as follows: Section 3 provides the mathematical modeling along with the description of parameters and variables used. Section 4 illustrates the local-search heuristic using incremental modeling approach. In Section 5, test results are discussed and Section 6 concludes the paper and outlines some potential for future research.

3. IRP mathematical model

This section provides the description of the instance data and then outlines the mathematical model, i.e. objective functions and constraints for the industrial gas distribution.

3.1. Instance description

The model considers a multi-product, multi-period finite horizon model with heterogeneous tractors and cryogenic trailers required to deliver the liquefied gases. Time is represented as a continuous variable with horizon T. In other words, any time instant is given by a point in the interval $[0, T]$. Since the telemetry provides hourly tank levels, it is hard to track and forecast the inventory levels at a more frequent (e.g., minute) level. Thus, consumption and production rates are given discretely for time-steps of size V, such that $V \times H = T$ with H the number of time-steps over the horizon. In the problem instances, the granularity adopted for V is 1 h. H is the total number of hours in the horizon T.

In practice, a cryogenic trailer can load many products but a purging operation has to be performed before a different product is loaded. In our model, a trailer is dedicated to carry a specific liquefied product only. Since we consider a multi-product IRP model, a problem instance should have dedicated trailers for each product. The distribution of all products needs to be optimized together because of the shared resources (drivers and tractors) between products. If not the case, each product could be individually addressed. In practice, it is possible for a driver to perform deliveries of different products in one shift by hitching/unhitching trailers. In our model, we assume that only one product can be delivered in a single shift of the driver. The size of a problem instance is essentially defined by the number of depots (where tractors and trailers are based), customers and sources (e.g., production plants). The number of drivers, tractors, and trailers as well as the number of call-in orders, and the time-steps for which consumptions/productions are provided over the horizon also impact the problem size.

As shown in Fig. 1, a given IRP instance consists of a number of depots, customers, and production plants. Depots are the facilities at which drivers, tractors, and trailers are based. Each driver, tractor, and trailer is assigned to only one depot. Depots are available during the whole horizon whereas tractors and trailers have limited availability represented through time windows. A time window is the time interval during which a resource is available continuously. A driver is allowed to drive certain tractors based on the certifications achieved. A tractor can be hitched to certain trailers based on the compatibility. We call a vehicle as the combination of a driver, a tractor, and a trailer. Each vehicle has an activity based fixed costs and variable costs. The activity based costs are related to loading operations, delivery operations, and layovers.

A layover of a minimum duration is taken when a driver has either driven for legally allowed maximum driving time or worked for legally allowed maximum working time. Drivers, in practice, take layovers even before either limit is reached depending upon when they reach appropriate resting places. We define maximum working time, maximum driving time, and maximum amplitude for each driver in a problem instance. Maximum working time is the maximum time duration during which a driver can work after taking a layover. The definition of work includes driving and non-driving activities. Maximum driving time is the maximum time duration during which a driver can drive after taking a layover. The maximum amplitude puts the limit on maximum duration for a driver to be continuously *on duty*. The definition of *on duty* includes working and layover activities. The variable cost of a vehicle is calculated based on time taken and distance traveled during a shift.

A vehicle located at a depot is used to transport products from sources (e.g., production plants) to customers. In our model, a production plant producing multiple products is modeled as separate production plants producing each product individually. Similarly, if a customer consumes more than one product, it is modeled as a separate customer for each product. All production plants and customers have cryogenic tanks installed to store liquefied products. Each tank can store only one type of product. A tank in a problem instance is represented by its maximum capacity, safety level and initial tank quantity at the beginning of the time horizon. Production sources and customers can only be accessed for loading and delivery during allowed time windows respectively. Depots are always accessible and therefore, do not have any specific time windows. Since our model is deterministic, we have an hourly forecast for the production and the consumption at each plant and customer respectively for the whole time horizon. A positive forecast value is consumption at a customer and a negative value is production at a plant. Not all drivers, tractors and trailers can visit a location

Table 1
Global parameters.

Notation	Definition
B	Set of depots
I	Set of customers
J	Set of sources (e.g., production plants)
P	Set of products
D	Set of drivers
R	Set of tractors
S	Set of trailers
T	Time horizon
V	Size of each time-step in the horizon T
H	Number of time-steps in the horizon T
C	Array of IRP sub-problems
G	Number of customers to be considered in a cluster
minLayoverDuration	Minimum duration of any required layover
threshold	A constant between 0 and 1
tripSetUpTime	Set up time at the start and end of each shift/layover

due to access restrictions. The set up time for customers is the time taken to fill the tank with the required product quantity and for plants it is the time to load the trailer with the required product. These set-up times are independent of quantity delivered or loaded which is usually a good assumption for practical purposes. We differentiate the call-in customers (which are order-based) from the forecast customers under this VMI system. Therefore, there is no need to have a consumption forecast for call-in customers. Forecast customers can also place specific orders during the planning horizon.

An order is defined with three attributes: order quantity, earliest delivery time and latest delivery time. An order is considered satisfied when a delivery is made within the specified time window with the specified order quantity. A forecast customer does not always have orders but a call-in customer is delivered only to satisfy orders. Therefore, we do not manage the product inventory at a call-in customer. In our model, a service-level interruption for a forecast customer is called a stock out. Stock-out costs are incurred when tank level at a customer goes below safety level to ensure that the customer is always delivered at or above safety level. A stock-out cost per unit step is provided in the instance description. The total stock-out cost for a customer is calculated based on the number of time-steps under service-level interruption. An order is considered missed if not satisfied. A missed order cost per order is also specified. A time and distance matrix is provided in the input to generate the routes. The notations used to describe a problem instance are given in Tables 1–3.

3.2. Solution description

The solution of the heuristic algorithm to solve IRP is a set of shifts over the horizon H. A shift essentially is defined by the vehicle used. A vehicle is a triplet of driver, tractor, and trailer used in the shift. A shift always starts and ends at a depot. Therefore, the initial state of a shift is defined by the depot from where it starts and the initial product quantity in the trailer used in the shift.

Table 2
Resource parameters.

Notation	Definition
loadingCost(k)	Trailer loading cost for the resource $k \in D \cup R \cup S$
deliveryCost(k)	Product delivery cost for the resource $k \in D \cup R \cup S$
distanceCost(k)	Unit distance cost for the resource $k \in D \cup R \cup S$
timeWindow(k)	Set of availability time windows for the resource $k \in D \cup R \cup S$
timeCost(k)	Unit time cost for the resource $k \in D \cup R \cup S$
tractors(d) $\subseteq R$	Set of tractors driver d is allowed to drive
maxDrivingTime(d)	Maximum driving time for driver d before a layover
maxWorkingTime(d)	Maximum working time for driver d before a layover
maxAmplitude(d)	Maximum length of a shift for driver d
Trailers(r) $\subseteq S$	Set of trailers allowed to be hitched to tractor r
trailerCapacity(s)	Total capacity of the trailer s

Table 3
Facility and customer parameters.

Notation	Definition
tankCapacity(q)	Total capacity of the tank installed at facility $q \in I \cup J$
safetyLevel(q)	Safety level of the tank installed at facility $q \in I \cup J$
initialTankLevel(q)	Initial tank level at the start of the horizon at facility $q \in I \cup J$
allowedWindows(q)	Set of time windows during which the facility $q \in I \cup J$ can be accessed
allowedDrivers(q)	Set of drivers that can access the facility $q \in I \cup J$
allowedTractors(q)	Set of tractors that can access the facility $q \in I \cup J$
allowedTrailers(q)	Set of trailers that can access the facility $q \in I \cup J$
setupTime(q)	Set-up time required for loading/delivery at facility $q \in I \cup J$
forecast(q, h)	Production/consumption forecast for facility $q \in I \cup J$ at time-step $h \in H$
distance(p1, p2)	Actual distance between two points $p1$, $p2 \in B \cup I \cup J$
time(p1, p2)	Travel time between two points $p1$, $p2 \in B \cup I \cup J$
callIn(i)	A binary parameter such that *callIn(i)* = 1 when customer $i \in I$ is a call-in customer otherwise a forecast customer
orders(i)	Set of orders for customer i
orderQuantity(i, o)	Order quantity of order $o \in$ *orders(i)* for customer i
earliestDeliveryTime(i, o)	Earliest delivery time of order $o \in$ *orders(i)* for customer i
latestDeliveryTime(i, o)	Latest delivery time of order $o \in$ *orders(i)* for customer i
stockOutCost(i)	Cost per unit time-step of when level is below safety level for customer i
missedOrderCost(i)	Cost of not satisfying an order for customer i
PLB(i)	Lower bound of logistics ratio for customer i

A shift is essentially a list of chronologically ordered operations. An operation is characterized by the facility where it is performed, quantity either loaded at a plant or delivered to a customer, start-time, and end-time. A positive quantity in an operation means that it has been performed at a customer facility whereas a negative quantity means a loading operation at a plant. A shift can also be defined as a list of trips. A trip is a list of operations chronologically ordered such that it starts from either a depot or a plant and ends at either a depot or a plant. In practice, a shift usually has the number of trips between 1 and 3. Fig. 2 shows an example of different operations and trips in a shift.

The start-time and the end-time of a shift are defined by the start-time of the first operation and the end-time of the last operation respectively in the list of operations. The first and last operations in the list are always performed at the same depot. The first operation of a shift is always a pre-trip operation and the last a post-trip operation. The pre-trip and post-trip activities at a depot are usually inspection activities to determine any potential equipment failure that has to be taken care of before continuing a shift. The quantity delivered for both pre-trip and post-trip operations is set to 0 as trailer quantity does not change for these operations. The time duration for both pre-trip and post-trip operations is the same for all shifts and is set to a constant value. A list of layovers is also maintained for each operation. The list has all the mandatory layovers taken after the end of the previous operation and before the start of the current operation. The inventory levels at a given time-step at a plant and a customer are calculated using the loading and delivery operations during that time-step respectively. Tables 4–6 define the notation for the solution of the heuristic model.

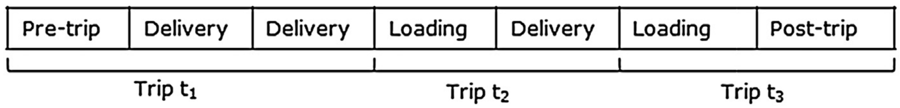

Fig. 2. Trips/operations in a shift.

Table 4
Shift specific notation.

Notation	Definition
Y	Set of shifts over the Horizon *H*
driver(y)	Driver assigned to the shift $y \in Y$
tractor(y)	Tractor assigned to the shift $y \in Y$
trailer(y)	Trailer assigned to the shift $y \in Y$
depot(y)	Depot assigned to the shift $y \in Y$
nbOfLoadings(y)	Number of loading operations during shift $y \in Y$
nbOfDeliveries(y)	Number of delivery operations during shift $y \in Y$
nbOfLayovers(y)	Number of layover operations during shift $y \in Y$
totalDistance(y)	Total distance traveled during shift $y \in Y$
totalTime(y)	Total time taken by the shift $y \in Y$
initialTrailerLevel(y)	Product left in the trailer at the beginning of the shift $y \in Y$
shiftCost(y)	Cost of the whole shift
OSC(y)	Optimal shift cost used in Benoist et al. (2011) for a shift $y \in Y$
trips(y)	List of trips in shift $y \in Y$
endTrailerLevel(y)	Product left in the trailer when shift $y \in Y$ is completed

Table 5
Trip specific notation.

Notation	Definition
tripDeliveries(t)	List of deliveries in a trip *t*
triggerDeliveries(t)	List of trigger deliveries in a trip *t*
tripCost(t)	*Total cost* associated with a trip *t*
depotTrip(t)	Depot used in the shift *y* where $t \in trips(y)$
tripMarginalCost(t)	Sum of marginal costs of all balance drop operations in trip *t*
noLoading(t)	A binary variable. If there is no loading operation in a trip and initial trailer level is full for the trip, then *noLoading(t)* = 1 otherwise 0

3.3. Business constraints

The model has to ensure that the components of a vehicle (i.e., the driver, tractor, and trailer) used in a given shift are compatible with one another. The vehicle used should be available during the whole duration of a shift. Also, a vehicle can only be assigned to a single shift at any point of time during the horizon. All facilities that are visited during the operations of a given shift should be accessible to the vehicle used. And, the operations should be performed at the facilities when they are open. There are also some additional constraints for drivers.

Table 6
Operation and facility specific notation.

Notation	Definition
point(m, y)	Facility at which the operation $m \in operations(y)$ in shift $y \in Y$ is done
quantity(m)	Quantity loaded/delivered during operation *m*
startTime(m)	Time at which operation *m* is started
endTime(m)	Time at which operation *m* is completed
LR(m)	Logistics ratio of operation *m*
layovers(m)	List of layovers which are taken before the start of the operation *m* and after the end of operation previous to *m*
tankQuantity(q, h)	Product in the tank installed at facility $q \in I \cup J$ at time-step $h \in H$
operations(q, h)	List of all operations at facility $q \in I \cup J$ at time-step $h \in H$
siteOperations(q)	List of all operations at facility $q \in I \cup J$
nbMissedOrders(i)	Number of missed orders at customer *i*
nbStockOuts(i)	Number of stock outs at customer *i*
maximumQuantity(i)	Maximum quantity that can be delivered to a customer *i* in a single operation

A driver can only start his shift after taking a minimum amount of rest which is the minimum layover duration. During a shift, a driver cannot violate the maximum driving time and maximum working time constraints. Also, a driver's shift duration cannot exceed maximum amplitude time. When a driver takes a layover, he has to perform pre-trip and post-trip operations at the start and the end of the layover respectively. Fig. 3 provides some examples of possible shifts. Shift 1 starts and ends at a depot with pre-trip and post-trip operations. Trailer is loaded at a plant and deliveries are made to four customers. Shift 2 starts with no loading operation as the trailer has already been loaded in its previous shift. Shift 3 has a layover operation which is taken by the driver to meet hours of service regulations. At the end of the shift 3, the driver performs a loading operation to ensure that the trailer is full for the next shift of the trailer and possibly to use the driver's remaining working time effectively.

In some cases, only a loading operation is possible during an entire shift (e.g. due to a plant outage, product is sourced from an alternate plant) which is quite far from the depot as shown in shift 4. Inventory management constraints ensure that the inventory at a facility is managed effectively.

Eq. (1) is a mass-balance equation where the tank level at any time-step is equal to the tank level at the previous time-step minus the forecast plus the sum of all quantities in the operations made between previous and current time-step. The tank level at a facility *q* at each time-step *h* of the horizon cannot be negative, cannot exceed the installed capacity and is calculated as:

$$\tan kQuantity(q,h) = \min\left(\tan kCapacity(q), \max\left(0, \tan kQuantity(q,h-1) - forecast(q,h) + \sum_{m \in operations(q,h)} quantity(m)\right)\right) \quad (1)$$

3.4. Long-term objective function

The objective function is related to both service-level requirements and the efficiency of the operations. The service-level requirements for call-in and forecast customers are different and therefore, are separately treated. The first part of the objective function is to ensure that all orders are satisfied for all call-in and forecast customers during the horizon *H*. If an order is missed for a customer, a missed order cost is applied. As mentioned in Section 3.1, an order for a customer is considered missed if either the delivery quantity is not satisfied or the delivery is not made between earliest delivery time and latest delivery time. The total missed order cost considers all customers as forecast customers can

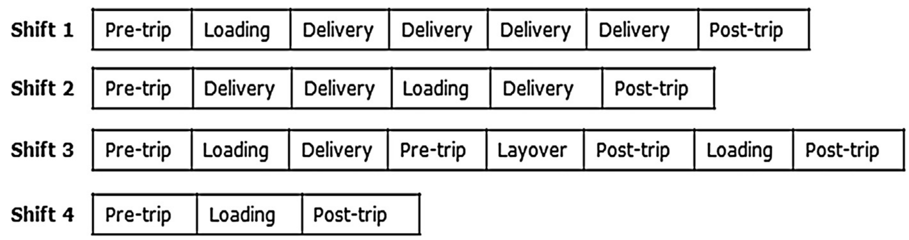

Fig. 3. Shift examples.

have orders as well is given as:

$$Total\_Missed\_Order\_Cost = \sum_{i \in I} missedOrderCost(i) \times nbMissedOrders(i) \quad (2)$$

The second part of the objective function is to ensure service-level interruptions (i.e. stock outs) do not happen. Since we are using a deterministic forecast, safety level accounts for any uncertainty in actual consumption for forecast customers. In practice, the logistics analysts can allow tank levels for forecast customers to go below safety levels. In our model, a stock-out cost is applied for every time-step when the tank level at a forecast customer is below safety level. For a call-in customer, the supplier is not responsible for avoiding stock out. So, the inventory level is not monitored and no stock-out cost is applied in the model. Total stock-out cost for all forecast customers during the horizon T is shown below:

$$Total\_Stock\_Out\_Cost = \sum_{i \in I} stockOutCost(i) \times nbStockOuts(i) \quad (3)$$

The third part of the objective function is related to the efficiency of the bulk gas distribution. Efficiency is measured through logistics ratio (LR). LR is the ratio of the total distribution costs and the total product delivered. The total distribution cost is the sum of total cost of each shift in the heuristic solution. The total shift cost is determined based on the total distance traveled, total shift duration as well as number of all loadings and deliveries in the shift. Total shift cost for the horizon T is shown below:

$$Total\_Shift\_Cost = \sum_{y \in Y} \left(\begin{array}{l} loadingCost(tractor(y)) \times nbOfLoadings(y) + \\ loadingCost(driver(y)) \times nbOfLoadings(y) + \\ deliveryCost(driver(y)) \times nbOfDeliveries(y) + \\ layoverCost(driver(y)) \times nbOfLayovers(y) + \\ Dis\tan ceCost(driver(y)) \times totalDis\tan ce(y) + \\ timeCost(driver(y)) \times totalTime(y) \end{array} \right) \quad (4)$$

Total product delivered to customers is based on all products and is given below:

$$Total\_Product\_Delivered = \sum_{i \in I} \sum_{m \in siteOperations(i)} quantity(m) \quad (5)$$

Logistics ratio (LR) is then calculated as:

$$Logistics\ Ratio = \frac{Total\_Shift\_Cost}{Total\_Product\_Delivered} \quad (6)$$

In the local-search heuristics, LR is not directly optimized as optimizing the logistics ratio for a short-term horizon may not lead to long-term optimal schedules. Therefore, a long-term logistics ratio (LTLR) is defined and optimized in the heuristics which is based on the notion of practical lower bound (PLB). PLB is the realistic cost of delivering unit product to a customer. It is not clear how to find the theoretical lower bound of cost of delivering unit product to a customer but PLB is easy to calculate and implement. PLB for a customer $i \in I$ denoted by PLB_i is determined assuming a direct delivery to the customer and the maximum delivery quantity possible. The maximum delivery quantity that can be delivered to a customer i is calculated as:

$$\max imumQuantity(i) = \min \left((\tan kCapacity(i) - safetyLevel(i)), \max_{s} \{trailerCapacity(s), s \in S\} \right) \quad (7)$$

The cost of a direct delivery to a given customer is the same irrespective of the volume delivered. PLB is therefore calculated as the ratio of the direct delivery cost and the maximum possible deliverable quantity. LTLR calculation is based on the premise that every delivery in the optimal schedule should have LR either the same or better than PLB. The challenge is to calculate LR for each delivery made rather than LR for the whole shift in the objective function. We devise a methodology to allocate the shift cost to each delivery operation in the shift such that the sum of the allocated shift costs to all delivery operations is equal to the total shift cost. In each trip in an optimal shift, it is expected that there should be at least one delivery to a customer which has its tank level at or close to safety level. We call such customers as trigger customers. It is possible that there may be two or more such trigger customers in a trip. Deliveries to customers other than trigger customers are called balance drop deliveries. Balance drop deliveries are made at customers that do not need the product but are delivered to improve the efficiency of the trip/shift. Therefore, a trip always consists of at least one trigger and some balance drop deliveries. Loading operations may happen either at the start or the end of the trip. We allocate the shift costs to trigger and balance drop deliveries differently. Trips are made to satisfy customers that trigger deliveries and therefore, are allocated most of the cost of the shift. Balance drop deliveries are only assigned the marginal increase in the shift cost. The next step is to classify each delivery into either a trigger or a balance drop delivery. We first define a threshold factor that takes a value between 0 and 1. If a delivery is made to satisfy an order, then it is automatically considered as a triggered delivery. For the non-order deliveries, Eq. (8) is evaluated.

$$\tan kQuantity\left(i, \frac{startTime(m)}{V}\right) \leq safetyLevel(i) + threshold \cdot \max imumQuantity(i), \; m \in trips(y), \quad y \in Y \quad (8)$$

Eq. (8) provides the tank level at customer i at the time-step $\frac{startTime(m)}{V}$ where V is the step-size as defined in Section 3.1. If Eq. (8) is true, then the delivery is considered as triggered, otherwise it is a balance drop delivery. We then determine the marginal cost for balance drop deliveries. To do that, we calculate the trip costs for the actual trip and the modified trip. The modified trip is the actual trip without the balance drop operation for which marginal cost has to be determined. The difference in the trip costs between the actual and the modified trip is the marginal cost assigned to the balance drop delivery. Once the marginal cost is known, LR for a balance drop delivery is simply calculated by dividing it with the quantity delivered. The total marginal cost is calculated for all balance drop operations in a trip. The remaining trip cost is assigned to the trigger deliveries based on the amount of product delivered to each trigger delivery. The remaining trip cost is the cost associated with all trigger deliveries in a trip and is given below:

$$\begin{aligned} tripTriggerCost(t) = {} & tripCost(t) - tripMarginalCost(t) \\ & + noLoadings(t) \times (loadingCost(tractor(y)) \\ & + loadingCost(driver(y))), \quad t \in trips(y), y \in Y \end{aligned} \tag{9}$$

Eq. (9) shows that if the trailer level at the start of trip is full, then the cost of a loading operation is applied to the trip even though no loading operation is performed in the trip. This essentially means that the loading operation has been performed in the previous shift of the trailer in a trip having only a loading operation. Logistics ratio for a trigger delivery $m \in triggerDeliveries(t)$, such that $t \in trips(y)$, $y \in Y$ is calculated as:

$$LR(m) = \frac{tripTriggerCost(t) \times distance(depotTrip(t), \mathrm{point}(m)) \times quantity(m)}{\sum_{j \in triggerDeliveries(t)} distance(depotTrip(t), \mathrm{point}(j)) \times quantity(j)} \tag{10}$$

Eq. (10) shows that the cost to satisfy a triggered delivery to a customer is dependent on the amount of product delivered as well as its distance from the depot. Thus, LR for each delivery either triggered or balance-dropped in a given trip can be easily calculated. We already know how to calculate PLB for each customer. The LTLR for all shifts is then calculated as shown below:

$$LTLR = \sum_{y \in Y} \sum_{t \in trips(y)} \sum_{m \in tripDeliveries(t)} (LR(m) - PLB(\mathrm{point}(m))) \tag{11}$$

The advantage of Eq. (11) is that it allocates cost to a delivery based on its urgency to avoid stock outs and drives the heuristic to an optimal solution by checking the optimality of each operation directly. A move that improves or maintains *LTLR* in the third phase of optimization is accepted otherwise rejected. Benoist et al. (2011) first calculate lower bound (LB) of LR for each customer. LB for a customer is calculated assuming a direct delivery model with the maximum trailer capacity available. LB may not be realizable in practice for some customers whereas PLB is always achievable in our model. The objective function proposed by Benoist et al. (2011) then determines optimal shift cost (OSC) and surrogate logistics ratio (SLR) as shown in Eq. (13).

$$OSC(y) = \sum_{t \in trips(y)} \sum_{m \in tripDeliveries(t)} (quantity(m) \times LB(\mathrm{point}(m))), \quad y \in Y \tag{12}$$

$$SLR = \frac{\sum_{y \in Y} (shiftCost(y) - OSC(y))}{Total\_Product\_Delivered} \tag{13}$$

SLR is calculated as the sum of difference between real shift cost and OSC for all shifts divided by total product delivered in all the shifts. In other words, SLR is the global extra cost per unit of product delivered compared to LB. Based on our observation, SLR is quite sensitive to LB and encourages sub-optimal balance drops. This is due to the reason that SLR does not consider inter-delivery distances among customers in a shift for calculating OSC. LTLR focuses on the optimality of each operation and the overall shift whereas SLR only tries to ensure shift optimality.

4. Local-search heuristic with incremental approach

In practice, logistics analysts start with a draft of the delivery schedule for the scheduling horizon (e.g., next 1 day) and then make local adjustments to improve it further. Such local adjustments are always finite in number. The local-search heuristic proposed in Benoist et al. (2011) closely resembles this human practice to search locally for optimal solutions using a simple first-improvement stochastic descent approach. The search strategy looks for a new feasible solution in the neighborhood of an existing solution. This paper proposes an improvement to this local-search heuristic by introducing an incremental approach to solve IRP. The heuristic by Benoist et al. (2011) uses an initial solution which is generated using a greedy algorithm. The methodology used in the greedy algorithm is shown in Procedure 1.

Table 7
Test case A.

Customer index	Capacity	Safety level	Initial tank quantity
1	25,000	5000	6000
2	60,000	10,000	15,000

The objective of the local-search heuristic is to first satisfy the customer orders, then avoid stock outs, and finally optimize the cost to deliver unit product (logistics ratio). The stopping criterion for the heuristic is the completion of a given number of iterations to improve upon the initial solution. It is assumed that the convergence to an optimal solution is reached before the stopping criterion is met. The initial solution is improved upon by a set of moves which are randomly selected. Each move is equally likely to be selected for the next iteration. All the moves have two main elements: a direction and a step-size. The direction of a move determines the type of change to be made in an existing solution. The step-size specifies the number of operations to be changed.

There are two main categories of the moves: operation-level and shift-level moves. The operation-level moves work by modifying operations within a single shift. Shift-level moves create new shifts and operate on many shifts together. An operation-level move can try to add a new operation in an existing shift, modify or delete an existing operation, etc. A shift-level move can create a new shift with new operations, exchange operations between two different shifts using different strategies, etc. Some examples of both types of moves are shown graphically in Fig. 4.

The implementation of local-search heuristic allows the execution of millions of such complex moves in a few minutes. In the heuristic, we do not allow deterioration of the solution to get out of the local-minima but rather use complex moves to avoid it. Also, we do accept moves that do not improve the LR. The main challenge in the local-search heuristic is to develop an objective function that can take short-term decisions ensuring long-term optimality. Optimizing the LR over a short-term horizon does not lead necessarily to long-term optimal decisions.

A good short-term decision avoids delivering a far-away customer that does not need product to avoid stock-out during the planning horizon. But, it may be sub-optimal not to deliver the far-away customer now as another customer nearby has to be delivered in the planning horizon. Optimizing LTLR in Eq. (11) avoids such sub-optimal decisions. As mentioned earlier, the heuristic is divided into three optimization phases. The first two phases are related to service-level requirements. The first phase is missed order optimization where dedicated moves are applied to minimize the missed order cost. The second phase is stock-out optimization where stock-out costs are minimized with dedicated moves ensuring the previous phase objective function does not deteriorate. The third phase is improving the efficiency of the deliveries using long-term objective function defined in Eq. (11).

Procedure 2 provides the local-search heuristic used in all the three phases of the heuristics. The objective function evaluates whether the new solution obtained through the local search is an improvement over the previous solution. The test results in

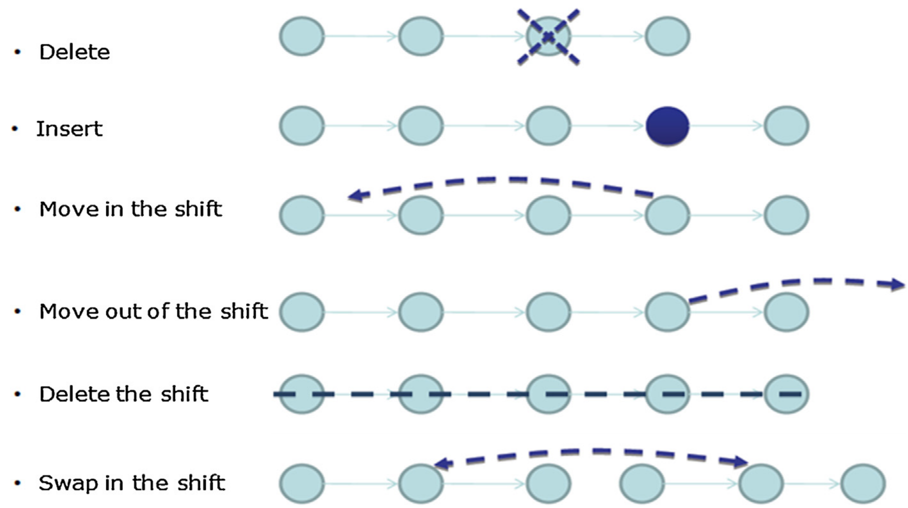

Fig. 4. Move examples.

Section 6 show that using the improved objective function gives better results over the objective function used in Benoist et al. (2011) defined in Eq. (13). A good indicator of the near-optimality of a local-search heuristic solution is the number of the moves accepted during the optimization. The acceptance rate (AR) of the moves in the original heuristic is very low (e.g. on average 5% of moves are accepted for test cases using 5 millions of iterations). To improve the acceptance rate of the moves and to exploit the inherent nature of the problem, we decompose the main problem into many sub-problems.

In bulk gas distribution, a customer is optimally delivered only when either a full load drop is possible or when its current level is at safety level. A full load drop delivery is made when the full trailer quantity is delivered to a customer. The other cases of optimal deliveries are when either a delivery is a balance drop or is forced by some business constraint like equipment or product unavailability. A balance drop is made to deliver the remaining trailer quantity to ensure that an empty trailer goes back to the depot. Assuming that there are no system constraints to prevent an optimal delivery, a customer can be delivered either an optimal volume based on its tank capacity and trailer volume or a balance drop for optimality reasons. The balance drop generally happens at a customer which is in the vicinity of the trigger customer for it to be optimal. In other words, an optimal delivery to a given customer is generally independent of the other customers not in the neighborhood of that customer.

This provides us the motivation to solve the industrial gas distribution problem by decomposing it into sub-problems which are quasi-independent. We first divide the IRP into scheduling horizon (SH) and non-scheduling horizon (NSH) customers. SH customers are those that need product delivered during the scheduling horizon otherwise they have a stock out. NSH customers are those which need not to be delivered during the scheduling horizon. The scheduling horizon in real-life instances can be one or several days (e.g. 1, 3 or 7 days).

In practice, the main objective is to generate an optimal schedule for scheduling horizon which is put into execution by the logistics analysts. The optimal schedule for the scheduling horizon has to take into account the impact of NSH customers on the SH customers. This impact on the solution is most likely through either balance drop deliveries to improve the LR or to avoid any stock outs during non-scheduling horizon. All SH customers are arranged by the urgency to deliver product to avoid stock outs. The urgency is determined by the initial tank level at the start of the horizon, the consumption rate and the travel time from the depot/plant. We also determine the G nearest customers by distance for SH customers. In any sub-problem, we assume that all vehicles are the same as in the complete IRP except the number of customers. All the sub-problems are solved with a time horizon equal to the scheduling horizon. In the last iteration, the main IRP is solved with the whole time horizon (typically 2 weeks) with all the customers.

The first sub-problem is solved with a single customer in the array C i.e. *C[0]* and the G nearest customers in its neighborhood. The solution of the first sub-problem is used as an initial solution to the second sub-problem which consists of the customers in the first sub-problem customers along with customer *C[1]* and its G neighbors. Therefore, the ith sub-problem consists of customer *C[i]* and its G neighbors along with the customers in the $(i-1)$th

Table 8
Test case A shifts using original heuristic.

Shift #	Point	Quantity	Arrival (min)	Departure (min)	Trailer quantity
	Plant	−50,000	211	278	50,000
	Customer 1	20,025	294	340	29,975
Shift 1	Plant	−20,025	356	423	50,000
	Customer 2	46,593	487	532	3407
	Depot	0	596	626	3407
	Plant	−50,000	8791	8858	50,000
Shift 2	Customer 1	19,955	8874	8920	30,045
	Depot	0	8936	8966	30,045
	Plant	−50,000	13,951	14,018	50,000
Shift 3	Customer 2	40,002	14,082	14,127	9998
	Depot	0	14,191	14,221	9998

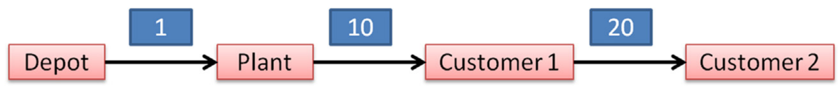

Fig. 5. Test case A distances.

Table 9
Test case A shifts with incremental approach.

Shift Id	Point	Quantity	Arrival (min)	Departure (min)	Trailer quantity
Shift 1	Plant	−50,000	1591	1658	50,000
	Customer 2	50,000	1722	1767	0
	Depot	0	1831	1861	0
Shift 2	Plant	−50,000	8843	8910	50,000
	Customer 1	20,160	8926	8972	29,840
	Depot	0	8988	9018	29,840

sub-problem. The ith sub-problem is solved using the $(i-1)$th sub-problem solution as an initial solution. In this incremental approach, no greedy algorithm is used to generate initial solutions. When new customers are added to the previous sub-problem, we ensure that a customer already in the previous sub-problem is not added again. In this incremental approach, the size of the sub-problem keeps increasing with every iteration. This incremental approach is different from rolling horizon framework used in many papers (e.g. Bard et al., 1998; Jaillet et al., 2002) and decomposition approaches used in Campbell et al. (2004) and You et al. (2011). The rolling horizon approach typically solves IRP of the same size in each step. A decomposition model usually solves different decision variables in different sub-problems such as the decisions made in one sub-problem are given as input to other sub-problems to reduce the complexity of the main problem.

The decisions made in a given sub-problem are not fixed and therefore, can be changed by the heuristic in the next iteration. Each sub-problem is solved using the local-search heuristic having the three phases of optimization. In essence, the incremental approach is reducing the problem size using two dimensions i.e. the number of customers and the time horizon. These two dimensions are the main source of complexity in IRP. The local-search heuristic is able to achieve near-optimal solution with a few thousand iterations for a small problem size as chances of getting stuck in local-optima are less. Therefore, the initial solution provided to a sub-problem is of good quality and helps the heuristic to find global optimal solution quickly. In the last iteration of the incremental approach, the whole IRP is solved together to promote global optimization of the problem. The total number of moves used in the incremental approach is the same when solved using the original heuristic. This is to ensure the comparisons between the two approaches are accurate. Procedure 3 details the incremental approach procedure used along with the local-search heuristic.

5. Illustrative example

In this section, we provide the test results of the incremental approach and the original local-search heuristic implementation by Benoist et al. (2011). The complexity of a particular test case is characterized mainly by the number of customers, depots, drivers, tractors and trailers. The complexity of the IRP problem is also impacted by the number of orders during the whole horizon and the number of customers that stock out at the start of the horizon. To show the effectiveness of the long-term objective function, we consider a manually generated test case with one depot, one source, one product and two customers for a 2-week planning horizon. There is unlimited product supply and no constraint on the number of drivers, tractors, and trailers. The trailer capacity used is 50,000. The details of the test case are shown in Table 7. Fig. 5 provides the distances between depot, source and customers, all of them are on a straight line.

Table 10
Global optimal vs. heuristic.

Test case	# Customers	LR – global optimal	LR – incremental	LR difference
A	2	0.2185	0.2291	−4.85%

Test case A is solved with original heuristic and the incremental approach with LR values of 0.2581 and 0.2347 respectively, an improvement of 9.97% (Tables 8 and 9). To better understand the results, Figs. 6 and 7 provide the output solutions visually. The red line in the graphs represents the safety levels below which a penalty cost is applied and the sudden jump in the tank levels corresponds to the tank deliveries. The time horizon is represented in minutes on X-axis and Y-axis shows the inventory level. The two solutions differ in delivering customer 2 whereas customer 1 is delivered at safety level in both the solutions. The first shift in Fig. 6 solution delivers customer 1 and then customer 2 is delivered little early whereas in Fig. 7, customer 2 is delivered in a separate shift. Also, customer 2 is served twice in the original heuristic thereby increasing its logistics ratio compared to being delivered once in the incremental approach. This example shows that the *LTLR* objective function helps in reducing early deliveries to customers thereby improving the LR.

We compare the incremental approach using local-search heuristic by solving the test case A exactly using CPLEX v12.4. Table 10 shows the comparison of the global optimal solution for the small test-instance discussed above with the incremental approach. The optimality gap for the incremental approach is 4.85% for test case A.

Table 11
Test instances.

Instance	# Customers	# Drivers	# Depots	# Tractors	# Trailers
1	50	5	1	5	5
2	75	13	1	8	8
3	110	15	1	10	12
4	155	20	1	12	14
5	181	25	1	12	14
6	185	25	1	12	16
7	192	30	1	15	18
8	195	30	1	15	18
9	210	30	1	15	18
10	231	30	1	15	18
11	248	30	2	15	19
12	260	30	2	15	19
13	272	35	2	20	20
14	280	35	2	20	20
15	310	30	2	20	20

Table 12
Test results.

Test case	Local search – without incremental approach			Modified local search – with incremental approach				LR gain (%)	SLR gain (%)
	Avg. AR (%)	Avg. LR ($/lb)	Avg. SLR ($/lb)	Avg. AR (%)	Avg. LR ($/lb)	Avg. SLR ($/lb)	Avg. LTLR ($/lb)		
1	2.66	0.28	0.083	2.81	0.27	0.078	0.074	3.6	6.0
2	2.92	0.31	0.091	3.11	0.28	0.086	0.083	9.7	5.5
3	3.12	0.67	0.043	3.34	0.63	0.04	0.037	6.0	7.0
4	3.83	0.50	0.022	4.19	0.46	0.02	0.018	8.0	9.1
5	4.64	0.12	0.085	4.98	0.11	0.079	0.074	8.3	7.1
6	4.72	0.29	0.041	5.05	0.26	0.037	0.033	10.3	9.8
7	4.8	0.87	0.023	5.19	0.80	0.021	0.019	8.0	8.7
8	4.81	0.94	0.026	5.25	0.86	0.023	0.02	8.5	11.5
9	5.12	0.48	0.045	5.62	0.43	0.039	0.035	10.4	13.3
10	5.45	0.23	0.088	5.94	0.21	0.078	0.072	8.7	11.4
11	5.61	0.62	0.031	6.06	0.57	0.027	0.024	8.1	12.9
12	5.88	0.76	0.038	6.47	0.70	0.034	0.03	7.9	10.5
13	5.96	0.47	0.057	6.55	0.44	0.049	0.042	6.4	14.0
14	6.21	0.38	0.014	6.89	0.34	0.012	0.011	10.5	14.3
15	6.23	0.35	0.079	6.86	0.32	0.071	0.066	8.6	10.1

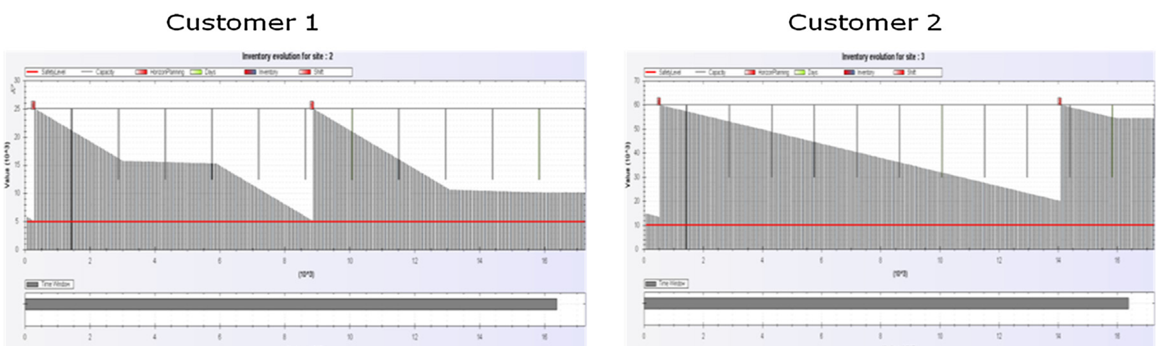

Fig. 6. Test case A inventories. (For interpretation of the references to color in the text, the reader is referred to the web version of this article.)

6. Test results

The algorithm and the changes proposed in this paper are implemented in C# and tested on a 16-core; 8 GB RAM computer server running windows O/S. Each of 15 test cases adapted from real-life data are solved twice, using the original heuristic first implemented in Benoist et al. (2011) and again with the modified heuristic using the incremental approach. The complexity of a particular test case is characterized by the number of customers, depots, drivers, tractors and trailers. Among the 15 test cases summarized in Table 11, instance 1 is the least complex and instance 15 is the most complex test case. We use 20 million iterations (or moves) as a stop criteria for each instance solved with the original heuristic. The choice of using 20 million iterations is based on the observation that the objective function converges before reaching this limit on the number of iterations for the given test data set. The heuristic is also run many times for the same instance (each time with a different random seed), and an average performance is reported. We observe that the results are not sensitive to the random seed used. The average standard deviation in LR when an instance is solved with different seeds is less than 1%.

Table 12 shows the test results for the 15 instances. We use LR and SLR to compare the results obtained by solving each test case with original heuristic and the incremental approach. Even though

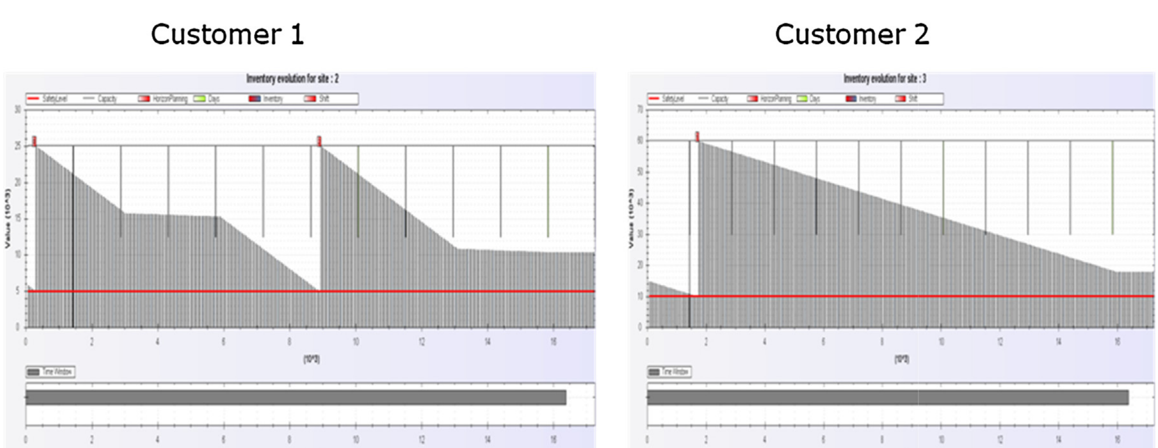

Fig. 7. Test case A inventories with incremental approach. (For interpretation of the references to color in the text, the reader is referred to the web version of this article.)

Procedure 1
Greedy algorithm used.

```
Procedure Greedy (IRP I)
Solution ← Φ;
List all demands and orders in IRP I;
While   Solution is not completed do
  Select the demand d with the earliest deadline;
  Create the least expensive delivery to satisfy d;
  Update the Solution to include this delivery;
  Update the list of demands and Orders;
End;
Return Solution;
End Greedy
```

Procedure 2
Local-search heuristics.

```
Procedure Local_Search_Heuristics (IRP I, Initial_Solution O)
Solution ← O;
While   missed_Order_Cost > 0 and
  current_iteration <= max_missed_orders_phase_iteration_limit
  Choose a move M randomly from the pool of missed_order_phase moves;
  If move M does not increase missed_order_cost
    Solution := (Solution + move M);
  End;
End;
While   stock_out_cost > 0 and
  current_iteration <= max_stock_out_phase_iteration_limit
  Choose a move M randomly from the pool of stock_out_phase moves;
  If move M does not increase stock_out_cost and missed_order_cost
    Solution := (Solution + move M);
  End;
End;
  While   current_iteration <= max_logistics_ratio_phase_iteration_limit
  Choose a move M randomly from the pool of logistics_ratio_phase moves;
  If move M improves LTLR and does not increase missed_order_cost
    Solution := (Solution + move M);
  End;
End;
Return Solution;
End Local_Search_Heuristics
```

we use LTLR as an objective function in the incremental approach, we also calculate SLR after optimal solution is generated to compare against the objective function used in Benoist et al. (2011). The results show that an average improvement of 7.86% in LR i.e. cost per unit product and an average improvement of 10.1% in SLR over original heuristic for 15 instances. This means that the incremental approach is able to drive the heuristic solution closer to the lower bound for LR used in the original heuristic.

There is also an average improvement of 9.1% in LTLR over SLR calculated for the heuristic solutions with the incremental approach. This improvement shows that the lower bound for LR used in our approach is more realistic than the lower bound used in Benoist et al. (2011). The difference in LTLR and SLR is how the lower bound for LR is calculated for each customer. LTLR uses the actual customer tank capacity whereas SLR assumes minimum tank capacity equal to maximum trailer capacity. We also note an average increase of 8.53% in acceptance rate (AR) over original heuristic. An average increase in AR of moves leads to better quality solutions and hence, the decrease in the cost to deliver unit product to customers. The coefficient of correlation between the increase in AR of moves and the increase in LR is 0.5 for the test data set which shows some linear dependence. There is also positive correlation between the increase in AR over original heuristic and the size of the problem. This relationship is expected as with more customers in the problem instance, a move made in the local heuristic has higher probability to get accepted.

Procedure 3
Incremental approach.

```
Procedure Incremental_IRP (IRP I)
Read the IRP problem I;
Determine the number of customers M that need product during Scheduling
  Horizon;
Create Array C of size M such that C[i] is the customer with ith priority to be
  delivered;
Create Array P of size M such that P[i] is a sub-problem to be solved;
Initialize P[0] ← Sub-problem consisting of customer C[0] and its G nearest
  customers by distance
Initialize Solution ← Local_Search_Heuristics(P[0], NULL);
For   i := 1:: M − 1 do
  P[i] := P[i − 1] + (C[i] + G nearest customers to C[i]);
  Solution := Local_Search_Heuristics(P[i], Solution);
End;
Solution := Local_Search_Heuristics(I, Solution);
Return Solution;
End Incremental_IRP
```

7. Conclusions

In this paper we propose improvements to the local-search heuristics proposed in Benoist et al. (2011) to solve an IRP occurring in bulk industrial gas distribution. The local search is an effective way to search for an optimal solution of IRP problem instance due to its ease to implement and ability to explore the solution space quickly. To ensure the local-search heuristic does not get stuck in local-minima, a long-term objective function is implemented to avoid such possibilities. At the same time, an incremental approach is applied where an IRP problem instance is divided into many sub-problems to improve the solution quality. Each sub-problem uses an initial solution obtained from solving the previous sub-problem using local-search heuristic using incremental approach. Therefore, no greedy algorithm is used to obtain an initial solution as in the original heuristic. The complete solution to the IRP problem is iteratively developed based on the new long-term objective function and incremental approach. The results show a significant improvement in cost to deliver unit product called logistics ratio for the test data set. A small manually generated test case is used to show the effectiveness of using long-term objective function to optimize the IRP problem. The long-term objective function for the logistic ratio phase is highly dependent on the lower bound to deliver a customer. In this paper, we use a direct delivery model to find the lower bound. Therefore, it needs to be further investigated to improve the lower bound for the cost to deliver unit product for each customer. We also do not consider uncertainty in the system parameters which can be explored in future research work especially the product supply and the demand uncertainties.

References

Abdelmaguid TF, Dessouky MM, Ordonez F. Heuristic approaches for the inventory-routing problem with backlogging. Comput Ind Eng 2009;56(4):1519–34.

Aghezzaf EH, Raa B, Landeghem HV. Modeling inventory routing problem in supply chain of high consumption products. Eur J Oper Res 2006;169:1048–63.

Anily S, Federgruen A. One warehouse multiple retailer inventory systems with vehicle routing costs. Manage Sci 1990;36(1):92–114.

Anily S, Federgruen A. Two echelon distribution systems with vehicle routing costs and central inventories. Special issue on Stochastic and Dynamic Models in Transportation edited by Dror, M. Oper Res 1993;41(1):37–47.

Archetti C, Bertazzi L, Hertz A, Speranza MG. A hybrid heuristic for an inventory routing problem. INFORMS J Comput 2010;24(1):101–16.

Archetti C, Bertazzi L, Laporte G, Speranza MG. A branch-and-cut algorithm for a vendor managed inventory routing problem. Transport Sci 2007;41:382–91.

Baita F, Ukovich W, Pesenti R, Favaretto D. Dynamic routing and inventory problems: a review. Transport Res 1998;32:585–98.

Bard J, Huang L, Jaillet P, Dror M. A decomposition approach to the inventory routing problem with satellite facilities. Transport Sci 1998;32(2):189–203.

Benoist T, Gardi F, Jeanjean A, Estellon B. Randomized local search for real life inventory routing. Transport Sci 2011;45(3):381–98.

Bertazzi L, Paletta G, Speranza MG. Deterministic order-up-to level policies in an inventory routing problem. Transport Sci 2002;36(1):119–32.

Bertazzi L, Speranza MG. Inventory routing problems: an introduction. EURO J Transport Logist 2012;1(4):307–26.
Campbell AM, Clarke L, Kleywegt A, Savelsbergh MWP. The inventory routing problem. Fleet Manage Logist 1998:95–113.
Campbell AM, Clarke L, Savelsbergh MWP. Inventory routing in practice. In: Toth P, Vigo D, editors. The vehicle routing problem. Philadelphia: Society for Industrial and Applied Mathematics; 2002. p. 309–30.
Campbell AM, Savelsbergh MWP. A decomposition approach for the inventory routing problem. Transport Sci 2004;38(4):488–502.
Chan LMA, Federgruen A, Simchi-Levi D. Probabilistic analysis and practical algorithms for inventory-routing models. Oper Res 1998;46:96–106.
Chien TW, Balakrishnan A, Wong RT. An integrated inventory allocation and vehicle routing problem. Transport Sci 1989;23:67–76.
Coelho L, Cordeau J-F, Laporte G. Thirty years of inventory routing. Transport Sci 2014;48:1–19.
Cordeau J-F, Lagana D, Musmanno R, Vocaturo F. A decomposition-based heuristic for the multi-product inventory-routing problem. Comput Oper Res 2015;55:153–66.
Dror M, Ball M. Inventory routing: reduction from an annual to a short period problem. Naval Res Logist Q 1987;34(6):897–903.
Ellis K, Sherali HD, Saunders L, Shao S, Crawford C. Bulk tank allocation to improve distribution planning for the industrial gas industry. IIE Trans 2014;46(6):557–66.
Federgruen A, Simchi-Levi D. Analytical analysis of vehicle routing and inventory routing problems. In: Ball MO, Magnanti TL, Monma CL, Nemhauser G, editors. Handbooks in operations research and management science, the volume on network routing. Amsterdam: North-Holland; 1995. p. 297–373.
Federgruen A, Zipkin P. Combined vehicle routing and inventory allocation problem. Oper Res 1984;32:1019–37.
Huang S-H, Lin P-C. A modified ant colony optimization algorithm for multi-item inventory routing problems with demand uncertainty. Transport Res E: Logist Transport Rev 2010;46(5):598–611.
Jaillet P, Bard JF, Huang L, Dror M. Delivery cost approximations for inventory routing problems in a rolling horizon framework. Transport Sci 2002;36(3):292–300.
Liu S-C, Lee W-T. A heuristic method for the inventory routing problem with time windows. Expert Syst Appl 2011;38(10):13223–31.
Marchetti PA, Gupta V, Grossmann IE, Cook L, Valton P-M, Singh T, Li T, Andre J. Simultaneous production and distribution of industrial gas supply-chains. Comp Chem Eng 2014;69(3):39–58.
Moin NH, Salhi S, Aziz NAB. An efficient hybrid genetic algorithm for the multi-period inventory routing problem. Int J Prod Econ 2011;133(1):334–43.
You F, Pinto JM, Grossman IE, Arora N, Megan L. Optimal distribution-inventory planning of industrial gases: I – fast computational strategies for large-scale problems. Ind Eng Chem Res 2011;50(5):2910–27.
Yu Y, Chen H, Chu F. A new model and hybrid approach for large scale inventory routing problems. Eur J Oper Res 2008;189:1022–40.