International Journal of Operational Research

ISSN online: 1745-7653 - ISSN print: 1745-7645
https://www.inderscience.com/ijor

An optimisation-simulation framework for integrated inventory and cash replenishment problem of automated teller machines in India

Ankush Kamthane, Prashant Singh, Ajinkya N. Tanksale

DOI: 10.1504/IJOR.2021.10040003

Article History:

Received:	03 December 2020
Accepted:	28 March 2021
Published online:	12 January 2024

An optimisation-simulation framework for integrated inventory and cash replenishment problem of automated teller machines in India

Ankush Kamthane, Prashant Singh and
Ajinkya N. Tanksale*

Mechanical Engineering Department,
IIT (BHU),
Varanasi, 221005, India
Email: arkamthane.mec18@itbhu.ac.in
Email: prashantsingh.mec18@itbhu.ac.in
Email: ajinkya.mec@iitbhu.ac.in
*Corresponding author

Abstract: This work is motivated by the problem of managing inventory and the optimal replenishment schedule for a network of automated teller machines (ATMs) in India. The objective is to minimise the occurrences of shortages at the ATMs and in turn to achieve a higher service levels while minimising the cost of holding inventory and replenishment of ATMs. The problem is casted as a rich variant of inventory routing problem with several practical restrictions such as maximum inventory at the ATMs. A two-phase iterative decomposition heuristic is proposed to efficiently solve the practical size problem instance. The computational experiments based on synthetic data are conducted to assess the efficiency and effectiveness of the proposed solution approach and a case of Varanasi city in India is presented for the analysis. The results show the effectiveness of our proposed approach over the conventional replenishment policies.

Keywords: automated teller machine; ATM; inventory routing problem; IRP; mixed-integer programming; heuristic; simulation.

Reference to this paper should be made as follows: Kamthane, A., Singh, P. and Tanksale, A.N. (2024) ‘An optimisation-simulation framework for integrated inventory and cash replenishment problem of automated teller machines in India’, *Int. J. Operational Research*, Vol. 49, No. 1, pp.19–46.

Biographical notes: Ankush Kamthane holds a Master’s in Industrial Management from Indian Institute of Technology (BHU), Varanasi, India and Bachelor’s in Mechanical Engineering. Currently, he is working as an OR Scientist in ORMAE. He has work experience in the field of operations research (optimisation). He has successfully completed projects in the field of discrete optimisation, supply chain management, sales optimisation, scheduling, location optimisation and resource optimisation.

Prashant Singh holds a BTech in Mechanical Engineering from Shri Ramswarrop Memorial College of Engineering and Management, Lucknow. He obtained his MTech in Industrial Management from Indian Institute of Technology (BHU) Varanasi, India. Currently, he is working as a backend developer in Tata Consultancy Services. His area of interest is mathematical modelling and operations research.

Ajinkya N. Tanksale is working as an Assistant Professor in Mechanical Engineering Department, Indian Institute of Technology (BHU) Varanasi, India. He received his PhD from Indian Institute of Technology Kharagpur and was a Postdoctoral Research Fellow at Sabanci University, Turkey. His area of interest is in applied operations research, facility location, supply chain management and heuristics. He has published his research in several reputed journals such as *Journal of Operational Research Society*, *Computers and Industrial Engineering*, *INFOR*, *RAIRO* and *British Food Journal*, and has presented his research in several international conferences. He is an Associate Editor of *OPSEARCH* – journal of OR Society of India, editorial board member of *International Journal of Industrial Engineering: Theory, Applications and Practice*, and serving as a reviewer to many journals in OR/MS domain.

1 Introduction

Due to convenience and utility, the automated teller machines (ATM) are one of the most commonly used services nowadays. There has been an increased penetration of ATMs in both urban and rural areas of developing counties like India in the recent past. Currently, there are 210,195 ATMs operating on-site and off-site in India as per the records of the Reserve Bank of India (https://www.rbi.org.in/Scripts/ATMView.aspx). The total amount of transactions using debit and credit cards through ATM is of the order of ₹3,200,999.6 million in the financial year 2019, which is a staggering amount. It is observed that even after demonetisation in India in the year 2016 and the promotion of cashless transactions through the unified payment interfaces and digitised platforms, the usage of ATMs has been continuously increasing in the country.

With the large network of ATMs, ensuring the availability of cash at each ATM at all times to meet the demand of the customers is a challenging task. This calls for managing an inventory of cash at the ATMs and the order and frequency of cash replenishment at the ATMs. In the Indian scenario, usually the policy of managing inventory at the ATMs is decided by the bank authorities and the task of replenishment of ATMs is outsourced to a service provider. A continuous review policy is practiced for managing the inventories at ATMs. That is, the order for refilling the ATMs are issued when the cash level drop down to the reorder point. However, due to the outsourcing of replenishment activity, in many cases, there is no control over the time when the ATM is visited. As a result, the ATMs run out of cash and the customer demand during the interval of order initiation and actual receipt of the cash is lost. This scenario is prevalent in the case of most of the public sector banks in India. This motivated us to propose an integrated model for both inventory management and replenishment policy for ATMs in the Indian scenario. The problem is further complicated due to several practical restrictions such as maximum inventory at the ATMs, different denominations of currency notes, strict time windows for the replenishment of the ATMs, maximum cash that can be carried by a vehicle during replenishment, security protocols during the operations and the high level of uncertainty associated with the customers demand of cash.

Motivated from this scenario, we propose a mixed-integer programming model to solve the problem of inventory management and cash replenishment of the ATMs. The consideration of inventory and vehicle routing aspects led us to visualise the problem as a variant of the well-known inventory routing problem (IRP). The output of the model is a period wise routing plan along with replenishment quantities at each ATM. Further, since the real-time demand for cash withdrawal is highly uncertain and very difficult to predict, the approach of simulation is followed to mimic the withdrawal patterns and demand quantity. The output of simulation runs is useful in estimating customer demand during a period more accurately. Then, the routing plans obtained from the deterministic model can be revisited based on inputs from the simulation model. This leads to a combined simulation-optimisation framework for the problem.

The IRPs are as well-known for their computational complexities. This fact is also applicable to the proposed model. As explained later, it is observed that the model becomes computationally intractable with an increase in the size of the problem instances. Therefore, to solve the problem efficiently and effectively, we propose a decomposition heuristic. In this, the inventory management problem and the vehicle routing problem (VRP) are attacked independently, sequentially and repetitively. The computational experiments have demonstrated the efficiency of the proposed heuristics. This heuristic, when coupled with the simulation phase, produces overall desirable results for the considered problem. In this way, with a mathematical model for a practical problem about the class of IRPs, a two-phase decomposition heuristics, and an integrated simulation-optimisation framework for the problem, we contribute to the body of the literature.

The remainder of the paper is organised as follows. A crisp review of the related literature is presented in Section 2. Section 3 describes the problem under consideration. A mix-integer programming model is presented in this section. The proposed two-phase decomposition heuristic is presented in Section 4. The methodology for simulation and the integrated simulation-optimisation framework is given in Section 5. Further, the results of computational experiments are presented in Section 6. Finally, Section 7 concludes the paper.

2 Related work

The problem of managing the cash at the ATMs has been considered with different aspects in the literature. For example, Zhang and Kulkarni (2018) devised an optimal strategy for the replenishment of the ATMs to minimise the long-term costs by taking into account the economies of scale while replenishing multiple ATMs simultaneously. Ekinci et al. (2019) views the problem for the individual ATMs as a cash replenishment-planning problem and propose a robust optimisation with linear programming to consider the uncertainty in cash requirement. Xu et al. (2019) considers a network of ATMs and propose a VRP while Kurdel and Sebestyénová (2013) modelled the problem as an IRP.

Although the problem classes such as IRP (Andersson et al., 2010; Coelho et al., 2014; Sofianopoulou and Mitsopoulos, 2018) and VRP (Toth and Vigo, 2002) have been extensively studied in the literature, their applications to the problem of managing inventory and replenishment of cash for the ATMs are rather limited. There are a few studies that modelled the problem as some variant of VRPs such as a single VRP (Min and Chong, 2012), multi-depot VRP (Boonsam et al., 2011), VRP with forced backhauls (Anbuudayasankar et al., 2012), periodic VRP (Michallet et al., 2014) and k-dissimilar VRP (Talarico et al., 2015). The problem is firstly as a rich variant of a multi-period inventory-routing problem for the ATMs in the Netherlands by Van Anholt et al. (2016). A network of ATMs is considered in their work which is to be managed through a single depot. Both cash deposits and withdrawal in the ATMs are allowed that generates both positive and negative demands at the ATMs. A clustering procedure is applied to decompose the problem into more manageable sub-problems. Similar work in the context of cash management and replenishment of ATMs in Santiago, Chile, is undertaken by Larrain et al. (2017). They have considered a special replenishment strategy that is the replacement of the entire cassette of the currency notes. The stock-outs are allowed but penalised to minimise them. A mixed-integer programming model is formulated and a Variable MIP neighbourhood descent algorithm is proposed to solve the problem efficiently. In a recent contribution, Batı and Gözüpek (2019) have considered the joint optimisation of cash management and routing for the recycling ATMs in Istanbul. They have proposed several heuristic algorithms for obtaining good quality solutions. Along with the conventional IRP, their model also takes a decision of replacing the conventional ATMs with the new recycle ATMs. However, in these works, the denomination mix has not been considered explicitly due to the nature of the ATMs. There are two recent works that take into account the issue of the denomination mix. Xu et al. (2019) have modelled the problem of cash logistics as a variant of the capacitated VRP with explicit consideration of a combination of different cash denominations. The minimum, maximum and expected demand of each denomination is taken into account and a penalty for deviating from the expected demand of each denomination is considered while modelling. In the most recent work by Van der Heide et al. (2020), the aspect of the time-varying denomination mix is integrated with inventory management decisions. The decision is the number of bills of certain denominations to be included in a box, inventory of the box at each ATM and whether an ATM will be visited or not in a period. However, the routing aspect has not been explicitly covered in this work.

There are some fundamental differences in our work with the existing literature. It can be seen that the proposed variants of IRP or VRP are based on the core inventory and routing aspects along with the complicating side constraints that arise due to distinct features of the problem at hand. Almost all the works are carried out, keeping in mind the logistic service provider companies. The service providers are making the replenishment decision. In the Indian context, such decisions are made by the bank authorities and only the logistics part is outsourced. Our work is intended to bridge the resulting gap and allow coordination in the decisions of banks and the cash logistics service providers. Secondly, majority of the ATMs in India are the conventional type that have different cassettes to hold for each denominations. Therefore, the denomination mix is an important decision while replenishing the ATM as well as while meeting the customer demands. We explicitly do consider these decisions in our model. Also, the uncertainty associated with customer demand and withdrawal patterns is tracked with an integrated optimisation-simulation approach. This is the novelty in the present work.

3 Problem definition and mathematical model

In this section, we formally define the inventory-routing problem for ATMs (IRP-A). The IRP-A is defined over a graph $G = (V, A)$, where V is the set of all vertices and $A = \{(i, j): i, j \in V, i \neq j\}$ is the set of arcs. Further, set V consists of a depot O (which is the central currency chest/nodal bank with a sufficient stock of currency) and a set $N = \{1, 2, \dots, n\}$, which is the set of nodes (that is ATMs). Each ATM is compartmentalised into designated storage spaces named 'cassettes' to hold different types of currency notes.

Each cassette has a fixed capacity and it is dedicated to a specific denomination of currency notes. Further, currency notes have different face values/denominations. The customer services are usually offered at the ATMs round the clock. However, there is a significant variation in the customer demand faced at an ATM over time and location. For example, the demand for cash at an ATM is much higher during peak periods as compared to off-peak periods. A variation in the demand can also be visible during different days of the week. The demand patterns of an ATM that is installed at a marketplace may significantly vary as compared to the ATM installed inside a university campus. Furthermore, there are time windows associated with the replenishment of the ATMs. Therefore, for the planning purpose, the planning horizon can be divided into discrete time periods. Due to the discretisation of periods, we can consider available operational hours during each period. In addition to the dynamic nature, there is another interesting feature associated with the customer's demand for cash. The same demand of a customer can be met in a variety of ways due to different types of currency notes and their face values. For example, the demand of ₹10,000 could be met with several denominations such as five notes of ₹2,000, 20 notes of ₹500, 100 notes of ₹100, or any possible combination of available currency notes. Additionally, if the cash available in an ATM is insufficient to meet a customer's demand, it is assumed to be lost. To provide better customer service, an objective is to meet demand during each period at all the ATMs without occurrences of shortages.

To ensure the availability of cash at the ATMs to meet the demands, one need to carry sufficient stock at the ATMs, or the ATMs should be replenished as and when required. However, as stated earlier, an ATM can carry only a limited number of currency notes due to the fixed capacity of the cassettes. Additionally, there are restrictions imposed by regulatory bodies on the aggregate amount of cash that can be stocked into an ATM at any instance. The logistics and replenishment of cash are typically outsourced to a third-party service provider who provides specialised and secured vehicles and human resources for the handling of cash from the depot to the ATMs. A set of capacitated vehicles is assumed to be available at the depot for the distribution of cash. In contrast to the typical volumetric or weight capacity, the capacity of a vehicle in this context refers to the maximum amount of cash that can be transported at a time.

A fixed cost is incurred if an ATM is visited in a period, which is typically charged by the service provider. Each ATM ensures unit inventory holding cost per period for currency notes that are being carried. These two costs exhibit a traditional trade-off similar to the inventory systems. Additionally, it should be noted that due to several practical aspects such as variations in the demand, limited storage capacity at ATMs and restriction on vehicle capacity, it may be impossible to meet the entire demand at each ATM during each period. Therefore, shortages are allowed in the model. However, each occurrence of shortage is penalised for minimising them. Considering this, the objective of the problem is to determine an optimal plan for replenishment and managing cash inventory at each ATM to minimise the total relevant cost. Apparently, due to the outsourcing of the cash logistics part, it seems that the routing problem to visit the ATMs is beyond the scope of the decision makers in the bank. However, as it is already clarified that the fixed cost of visiting an ATM is being charged by the service provider. Further, due to the limited time and practical constraints, it is important to know that how many ATMs can possibly be visited in a day. For this, consideration of routing aspects in the work is absolutely important. Incorporating the routing aspects in the model also allows decision maker to fully consider the possible solutions exploring optimal trade-off between visiting and inventory carrying costs.

The problem is formulated as a mixed-integer programming problem using the following notation.

3.1 Sets and indices

O	depot
$N = \{1, 2, \ldots, \lvert N \rvert\}$	set of all ATMs (nodes) indexed by i, j
$V = N \cup \{O\}$	set of all nodes indexed by i, j
A	set of all arcs i, j
$T = \{1, 2, \ldots, \lvert T \rvert\}$	set of periods indexed by t
$P = \{1, 2, \ldots, \lvert P \rvert\}$	set of currency notes indexed by p
$K = \{1, 2, \ldots, \lvert K \rvert\}$	set of vehicles indexed by k.

3.2 Parameters

C_i	cost of visiting node $i \in N$
h_{ip}	inventory holding cost of note type p at node i
π_i	per unit shortage cost at node $i \in N$
F_p	face value of note type p
D_{it}	demand at node $i \in N$ during period t
τ_{ij}	travel time on the arc $i, j \in A$
δ_i	service time at node i
θ_t	the available time during period t
U_p	maximum capacity of the cassette for holding notes of type p
U	maximum allowed cash in the ATMs
Q_k	capacity of vehicle (maximum cash carried).

3.3 Decision variable

I_{itp}	inventory of currency notes of type p at ATM i at the end of period t
f_{itp}	number of currency notes of type p withdrawal to meet demand at ATM i during time period t
q_{iktp}	number of currency notes of type p refilled at ATM i by vehicle k during the start of time period t

$$x_{ikt} = \begin{cases} 1, & \text{if vehicle } k \text{ visits node } i \text{ during period } t \\ 0, & \text{Otherwise} \end{cases}$$

$$y_{ijkt} = \begin{cases} 1, & \text{if vehicle } k \text{ travels path } i \text{ to } j \text{ during period } t \\ 0, & \text{Otherwise} \end{cases}$$

z_{ikt} number of times node i is visited by vehicle k during period t

U_{ikt} variable for sub-tour elimination.

3.4 Mathematical model

Objective function

$$Min \sum_{i \in N} \sum_{t \in T} \sum_{p \in P} h_p F_p I_{itp} + \sum_{i \in N} \sum_{k \in K} \sum_{t \in T} c_i x_{ikt} + \sum_{i \in N} \sum_{t \in T} \pi_i S_{it} \tag{1}$$

subject to

$$I_{itp} = I_{i(t-1)p} + \sum_{k \in K} q_{iktp} - f_{itp} \qquad \forall i \in N, t \in T, p \in P \tag{2}$$

$$S_{it} = D_{it} - \sum_{p \in P} f_{itp} F_p \qquad \forall i \in N, t \in T \tag{3}$$

$$I_{itp} \le U_p \qquad \forall i \in N, t \in T, p \in P \tag{4}$$

$$q_{iktp} \le U_p z_{ikt} \qquad \forall i \in N, k \in K, t \in T, p \in P \tag{5}$$

$$\sum_{k \in K} q_{iktp} + I_{i(t-1)} \le U_p \qquad \forall i \in N, t \in T, p \in P \tag{6}$$

$$\sum_{p} I_{itp} F_p \le U \qquad \forall i \in N, t \in T \tag{7}$$

$$\sum_{i \in N} \sum_{p \in P} q_{iktp} F_p \le Q_k z_{0kt} \qquad \forall k \in K, t \in T \tag{8}$$

$$\sum_{j \in V} y_{ijkt} = z_{ikt} \qquad \forall i \in V, k \in K, t \in T \tag{9}$$

$$\sum_{j \in V} y_{ijkt} = z_{ikt} \qquad \forall i \in V, k \in K, t \in T \tag{10}$$

$$U_{ikt} - U_{jkt} + Q_{kt} y_{ijkt} \le Q_{kt} - \sum_{p \in P} q_{jktp} F_p \qquad \forall i \in N, j \in V, k \in K, t \in T \tag{11}$$

$$Q_{kt} \ge U_{ikt} \ge \sum_{p \in P} q_{iktp} F_p \qquad \forall i \in V, t \in T, k \in K \tag{12}$$

$$y_{ijkt} \leq x_{ikt} \qquad \forall i \in V, j \in N, k \in K, t \in T \qquad (13)$$

$$\sum_{i \in V} \sum_{j \in V} y_{ijkt} \tau_{ij} + \delta_i \sum_{i \in V} z_{ikt} \leq \theta_{tk} \qquad \forall k \in K, t \in T \qquad (14)$$

$$\sum_{k} x_{ikt} \leq 1 \qquad \forall i \in N, t \in T \qquad (15)$$

$$I_{itp}, q_{iktp}, f_{itp}, S_{it}, U_{ikt} \geq 0 \qquad \forall i \in N, k \in K, t \in T, p \in P \qquad (16)$$

$$y_{ijkt} \in \{0, 1\} \qquad \forall (i, j) \in N, k \in K, t \in T \qquad (17)$$

$$z_{ikt} \in \{0, 1\} \qquad \forall i \in N, k \in K, t \in T \qquad (18)$$

$$z_{ikt} \in \mathbb{Z} \qquad \forall i \in \{0\}, k \in K, t \in T \qquad (19)$$

$$x_{ikt} \in \{0, 1\} \qquad \forall i \in N, k \in K, t \in T \qquad (20)$$

The objective function (1) minimises the total cost, including inventory holding cost, cost of visiting ATMs and the shortage cost at the ATMs. Constraint (2) is the cash flow conservation equation to balance the inventory carried from the previous period plus cash inflow (replenishment) to the cash outflow (withdrawal) at an ATM in each period. Constraint (3) defines shortage as the unmet demand which is the difference between the estimated demand and the actual withdrawal quantity at an ATM. Constraints (4)–(6) are due to the limited storage capacity of a cassette in an ATM. They ensure that the number of currency notes of each type stored in an ATM does not exceed the capacity of the ATM at any point of time during a period. Constraint (7) is to ensure that the total cash at an ATM does not exceed the overall cash limit. Constraint (8) guarantees that the total cash carried by a vehicle during a period does not exceed the permissible limit of the cash during its tour in the period. Constraints (9) and (10) are to make sure that arcs going into and emanating from a node are travelled only if a visit is planned to the node. Constraints (11) and (12) are sub-tour elimination constraints. Constraint (13) related the variables for visiting a node and path traversed through the node. Constraint (14) limits the total travel time and service time at ATMs by a vehicle with respect to the maximum available time to each vehicle in each period. Constraint (15) ensures that each ATM is visited by at most one vehicle in each period. Finally, constraints (16)–(20) define the nature of the variables.

4 Solution approach

It can be seen that the IRP-A is a variant of the IRP that has been a well-known NP-hard problem. Our computational experiments also confirm the fact that the computational time required to solve IRP-A increases exponentially with an increase in the problem size and it becomes computationally intractable. Therefore, it calls for a specialised solution approach that is computationally efficient and effective. In this work, we propose a decomposition-based iterative heuristic procedure to solve the IRP-A, as explained below.

It can be seen that IRP-A consists of two prominent decision phases – inventory management and vehicle routing. The first decision is the management of the inventory of cash at the ATMs over the entire planning horizon and the second decision is the optimal routing for the replenishment of ATMs in each period. Both the decisions are interrelated and considered in an integrated fashion in the IRP-A. However, there is a possibility to take these decisions sequentially. For example, initially, the decision-maker may decide which ATMs are to be visited and how much cash is to be delivered to the ATMs during each period to minimise the total cost of visiting ATMs and inventory holding costs. Then, having known the ATMs to be visited in each period, the plan for visiting them using different vehicles can be worked out. That is, inventory management decisions may preside over the routing decision. This allows the disintegration of the model into two parts and many small sub-problems that can be solved efficiently. Our proposed two-phase decomposition heuristic follows this very idea. Phase 1 of the heuristic deals with inventory management decisions for the entire planning horizon and phase 2 is to plan the routing of vehicles for the replenishment of ATMs in each period. The details of the heuristic are given in the following subsections.

4.1 Phase 1: inventory management

The inventory management phase deals with the decisions of replenishment quantity and frequency of visits for each ATM. These decisions are intertwined and exhibit the trade-off between the inventory holding cost and the fixed cost of visiting the ATMs. Therefore, the objective is to determine a policy that minimises the total relevant costs. This problem resembles the classical dynamic lot-sizing problem presented by Wagner and Whitin (1958) with some additional constraints that arise due to the problem setting. Further, when the vehicle routing phase is disjointed from the original model of IRP-A, the inventory management problem of each ATM becomes unique and can be solved independently. That is, the model naturally gets decomposed into one sub-problem for each ATM. Also, the replenishment decisions are based on the capacity of ATM and there is no point in considering multiple vehicles in this phase. Therefore, the relevant decision variables such as q_{iktp} and x_{ikt} are modified by omitting the indices of vehicles as q_{itp} and x_{it}, respectively. Then the model for phase 1 takes the following form.

Sub-problem 1: for each ATM $i \in N$

$$Min \sum_{t\in T} \sum_{p\in P} h_p * F_p * I_{itp} + \sum_{t\in T} C_i * x_{it} + \sum_{t\in T} \pi_i * S_{it} \tag{21}$$

subject to

$$I_{itp} = I_{i(t-1)p} + q_{itp} - f_{itp} \qquad \forall t \in T,\ p \in P \tag{22}$$

$$S_{it} = D_{it} - \sum_{p\in P} f_{itp} * F_p \qquad \forall t \in T \tag{23}$$

$$I_{itp} \le U_p \qquad \forall t \in T,\ p \in P \tag{24}$$

$$q_{itp} \le U_p * x_{it} \qquad \forall t \in T,\ p \in P \tag{25}$$

$$q_{itp} + I_{i(t-1)p} \leq U_p \qquad \forall t \in T, p \in P \tag{26}$$

$$\sum_p I_{itp} * F_p \leq U \qquad \forall i \in N \tag{27}$$

$$x_{it} \leq M_i \theta_t \qquad \forall t \in T \tag{28}$$

Equations (21)–(27) are analogous to equations (1)–(7) in the model IRP-A. Constraint (28) ensures that the ATM can be replenished in a period only if the available time during that period is non-zero. Here, M_i is a large positive number to ensure the sufficiency of the order quantity at ATM *i*. As the problem needs to be solved for one ATM at a time, this approach is highly efficient. The sub-problem 1 can be solved with a black box solver such as GUROBI within brisk computation time. Furthermore, a dynamic programming based approach can be followed for a very large size problem if the need be.

4.2 Phase 2: vehicle routing

Based on the output of phase 1, the next step is the optimal routing of vehicles. For each time period *t*, a subset of ATMs to be visited $N' \subseteq N$ and the number of each currency notes q_{itp} to be replenished in ATM $i \in N'$ is known from Phase 1. This quantity is hereafter regarded as the demand of each ATM for the brevity and designated as $\bar{D}_{ip}$ for a specific period *t*. To meet the demands of the ATM $i \in N'$, it needs to be visited by one of the vehicles from a set of vehicles *K* and must receive the exact quantity equivalent to $\bar{D}_{ip}$. Partial fulfilment of the demand of the ATMs is not allowed. If a planned ATM is not visited in a period, it leads to shortages and subsequent high penalty costs. Therefore, with the limited number of capacitated vehicles and available time for vehicle operations in each period, the objective is to produce a routing plan such that the maximum demand of the ATMs is met. This problem is equivalent to the capacitated vehicle routing problem (CVRP), with some complicating side constraints. However, unlike the traditional CVRP, the objective is not to minimise the travel cost. Also, it can be noted that this phase does not directly contribute to the objective function of IRP-A. Instead, the objective of this phase is to produce a feasible solution for each period based on the input from Phase 1.

Considering $\tilde{q}_{ikp}$ as an additional variable for the actual number of currency notes of type p replenished to ATM *i* by vehicle *k*, the mathematical model for phase 2 for each period (omitting the index for the period) is as follows.

Sub-problem 2: for each period $t \in T$

$$\text{Maximise} \sum_{k \in K} \sum_{i \in N'} \sum_{p \in P} \bar{D}_{ip} * F_p * x_{ik} \tag{29}$$

subject to

$$\tilde{q}_{ikp} \leq \bar{D}_{ip} x_{ik} \qquad \forall i \in N', k \tag{30}$$

$$\sum_{p} \tilde{q}_{ikp} = \sum_{p} \bar{D}_{ip} x_{ik} \qquad \forall i \in N', k \qquad (31)$$

$$\sum_{k \in K} x_{ik} \leq 1 \qquad \forall i \in N' \qquad (32)$$

$$\sum_{i \in N'} \sum_{p \in P} \tilde{q}_{ikp} * F_p \leq Q_{kt} * z_{0k} \qquad \forall k \in K \qquad (33)$$

$$\sum_{j \in V} y_{ijt} = z_{ik} \qquad \forall i \in \{o\} + N', k \in K \qquad (34)$$

$$\sum_{j \in V} y_{jik} = z_{ik} \qquad \forall i \in \{o\} + N', k \in K \qquad (35)$$

$$U_{ik} - U_{jk} + Q_k * y_{ijk} \leq Q_k - \sum_{p \in P} \tilde{q}_{ikp} * F_p \qquad \forall i \in N', j \in V, k \in K \qquad (36)$$

$$Q_k \geq U_{ik} \geq \sum_{p \in P} \tilde{q}_{ikp} * F_p \qquad \forall i \in \{o\} + N', k \in K \qquad (37)$$

$$y_{ijk} \leq x_{ik} \qquad \forall i \in \{o\} + N', j \in N, k \in K \qquad (38)$$

$$\sum\nolimits_{i \in V} \sum\nolimits_{j \in V} y_{ijk} * \tau_{ij} + \delta_i * \sum\nolimits_{i \in V} z_{ik} + \theta_k \qquad \forall k \in K \qquad (39)$$

Constraints (30) ensure that the actual number of notes delivered to an ATM is less than or equal to its actual demand. This constraint imposes that the number of notes delivered to an ATM are positive only if the ATM is actually visited during the period. Constraint (31) lays the similar requirement and adds up a restriction on partial fulfilment. Constraint (32) ensures that that the entire demand of an ATM must be fulfilled by the same vehicle. The remaining constraints in the problem 2 are the blatant adoption of their equivalent constraints from the IRP-A.

The sub-problem 2 can be seen as a CVRP with complicating side constraints which is NP-Hard and computationally intractable. Even the professional math programming solvers such as Lingo, LINDO, Cplex and GUROBI takes too long to produce an optimal or even a feasible solution to a medium/large size instance. Therefore, we propose a greedy heuristic reinforced with a local search to solve the VRP for each period. The greedy approach is a construction heuristic that develops a feasible solution by constructing a path for each vehicle. Given a set of ATMs to be visited in a period and the demand of these ATMS, the next node (ATM) to be visited by a vehicle is selected based on the ratio of demand and travel time. The visits of each vehicle are constructed such that the capacity of the vehicle and travel time restrictions are respected. The paths constructed using the greedy approach are further improved using 1-opt local search. The pseudo-code for the proposed construction and improvement heuristic is shown in Figure 1.

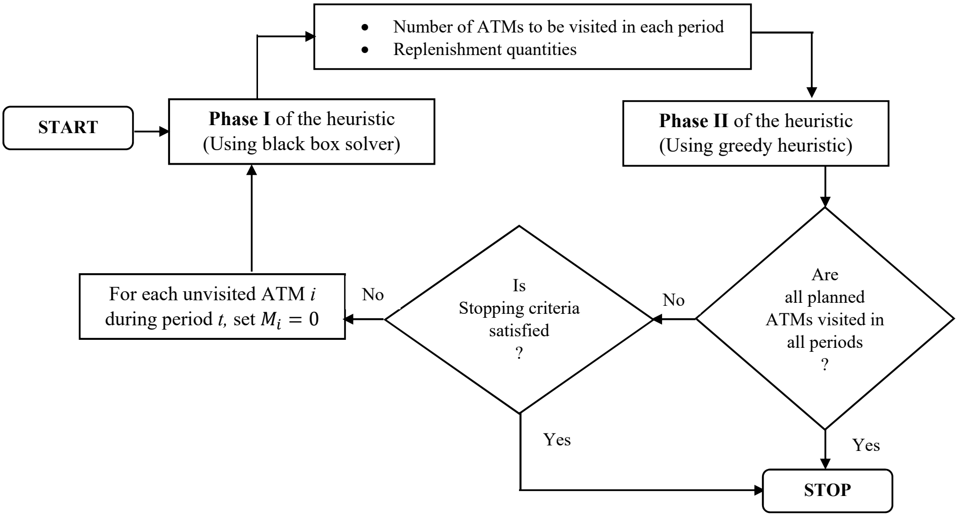
Figure 1 Pscudocode of the greedy heuristic for phase 2

Algorithm: Heuristic for Phase 2

```
0:  **Given**: Demand of ATM $i$ for denomination $p$ $\bar{D}_{ip}$ during time period $t$ from Phase 1
1:  *for* each period $t$ *repeat* Steps 2 to 7
2:      Define Set $\bar{N} = \{\varnothing\}$ for ATMs to be visited during period $t$.
3:      *for* each ATM $i$
4:          if $\sum_p \bar{D}_{ip} > 0, i \in \bar{N}$.
5:      Sort the elements of $\bar{N}$ in descending order of $\sum_p \bar{D}_{ip}$
6:      Define $\bar{K} = \{1, 2, ..., k\}$ as set of vehicles available for cash delivery.
7:      For each vehicle $k \in \bar{K}$
8:          set time travelled by as $T_k = 0$ and cash carried as $C_k = 0$
9:          Set current node of each vehicle $j$ as depot $\{O\}$.
10:     **While** $\bar{N} = \{\varnothing\}$
11:     **do**
12:         Select ATM $i \in \bar{N}$ with the largest value of $\sum_p \bar{D}_{ip}$
13:         if $\bar{K} = \{\varnothing\}$ : :
14:             **break**
15:         *else*
16:             *for* vehicle $k \in \bar{K}$
17:                 *if* $T_k + t_{ji} \le \theta_k$ and $C_k + \sum_p \bar{D}_{ip} f_p \le Q_k$
18:                     Set $x_{ik} = 1$
19:                     Update $T_k \leftarrow T_k + t_{ji}$ AND $C_k \leftarrow C_k + \sum_p \bar{D}_{ip} f_p$
20:                     *if* $T_k == \theta_k$ OR $C_k == Q_k$
21:                         Remove $k$ from $\bar{K}$
22:                     **break**
23:                     *else*
24:                         continue with the next vehicle k from the set $\bar{K}$
25:         **if**
26:             $\sum_k x_{ik} = 1$ remove $i$ from $\bar{N}$ and continue
27:     **else**
28:         **break**
29: **Apply local search to refine solutions**
```

Commonly, the output of phase 2 is a routing plan in which all the ATMs to be visited in each period as determined in phase 1 are visited by the vehicles and there are no incidents of shortages. However, due to the limited number of vehicles, the capacity of each vehicle and available time during each period, there may be a possibility that some of the planned ATMs could not be visited in a period in-spite of the best possible routing. In such cases, there is a need to revisit the inventory management plan worked out in phase 2. If an ATM *i* remains unvisited in phase 2 during a time period *t*, then we assign $M_i = 0$ for the ATM *i* during period *t* while re-solving phase 1. As a result, the ATM i is made unavailable for a visit during period *t* and an alternate period for a visit of ATM i is worked out. Then, phase 2 is solved once again to generate a feasible routing plan. If none of the planned ATM remains unvisited in any period, then the procedure is stopped. Otherwise, the feedback is given to phase 1 and it is solved again. In this way, phase 1 and phase 2 are implemented in an iterative fashion taking the output and feedback from one another. It is worth noting that in some of the cases, the shortages become inevitable due to the limited resources. In such cases, the solutions could not be improved even after several iterations of phases 1 and 2 and the algorithm can be terminated based on stopping criteria such as non-improvement in the solution, a fixed number of iterations or fixed solution time. The overall idea of the solution approach is depicted in Figure 2.

Figure 2 Overview of proposed heuristic algorithm


Number of ATMs to be visited in each period
Replenishment quantities
START
Phase I of the heuristic (Using black box solver)
Phase II of the heuristic (Using greedy heuristic)
For each unvisited ATM i during period t, set $M_i = 0$
No
Is Stopping criteria satisfied ?
No
Are all planned ATMs visited in all periods ?
Yes
Yes
STOP

5 An integrated optimisation-simulation framework

Although the deterministic IRP-A proposed in Section 2 captures several realistic characteristics of the problem, the model has certain limitations and simplistic assumptions that are essential for the tractability. For example, the customer demand at all the ATMs in each time period is considered to be certain and deterministic. Further, for balancing the flow of cash, customer demand is considered to be a one-shot activity in a given period. To illustrate this point, consider an ATM facing certain demand in a period of length of 6 hours. Suppose the initial inventory at this ATM is just enough to meet the demand equivalent to 1 hour. Now, if the ATM is replenished in the first hour, there will not be any shortages. On the contrary, if the ATM is replenished in the fifth hour, the demand between the first and fifth hour would be lost. However, this will not be taken into account in the present model. In the model, if the sum of the initial inventory and the replenishment quantity in a period exceeds demand in that period for an ATM, then there will not be any shortage. Thus, due to the aggregation of demand, the model fails to take into account the effect of individual and continuous customer demands at the ATMs. In addition to this, the number of currency notes withdrawal from an ATM is also considered at an aggregate level and the individual withdrawal patterns are ignored while developing in the model. There could be some scenarios where the demand of a customer cannot be fully satisfied due to the unavailability of certain types of notes. Under the presence of such uncertainty, the results of the deterministic model may not lead to the desired performance of the system.

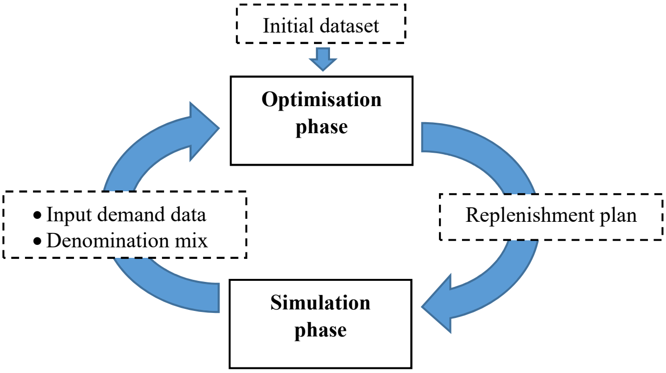
There are different ways to handle uncertainty while modelling the system, such as stochastic modelling or robust optimisation. However, given the complexity and size of the problem under consideration, it would be impossible to track all uncertain aspects using a robust/stochastic modelling approach. In such a scenario, simulation provides a convincing platform to deal with the uncertainties prevailing in the considered problem efficiently. Therefore, a discrete event simulation model is developed for considering uncertainty about the customer's demand and withdrawal patterns at each ATM, as explained below.

The simulation of customer visits and cash withdrawal for each ATM for the entire planning horizon can be thought of as a single server queuing system. The arrival rate of the customers, service time required, specific demand of the customers, the initial status of the currency notes and replenishment policy are the key input to this queuing model for simulation. The customer arrivals are assumed to occur singly and from an infinite population. The inter-arrival time and service time are assumed to be exponentially distributed. Further, the mean rate of arrival is varied depending upon the location of the ATM. The customer demands, in this case, are peculiar. There is a maximum limit for withdrawal from an ATM. Further, most of the ATMs provide an option for withdrawing a predefined amount of cash (e.g., 500, 1,000, 2,000, 5,000 or 10,000) popularly known as fast cash option) from the ATM. Customers can opt for this option or she may specify the withdrawal quantity of her choice. Therefore, upon the arrival of the customer, the desired pattern of the demand and the withdrawal quantity are decided randomly. At this juncture, it is also important to understand the mechanism of meeting the customer's demand. Let us assume three denominations of currency notes as ₹100, 500 and 2,000 that are prevalent in the Indian context. Any customer demand can be expressed as a linear combination of these currency notes. However, for a given demand, there are a large number of possible alternatives. For example, a demand of ₹3,000 can be met with 30 notes of ₹100 or six notes of ₹500 or one note of ₹2,000 and two notes of ₹500 or one note of ₹2,000 and ten notes of ₹100 or different combinations of ₹100 and 500 notes. This necessitates a uniform policy for withdrawal to meet the demands during the simulation run. We adopt a policy to give preference to the currency notes with larger face values to meet the demand. This means that the maximum portion of the customer's demand is met with the largest currency notes subject to availability and feasibility. In case the demand is unmet, the next largest currency note is selected and the maximum possible share of the unmet demand is met out of that. This process continues until the entire demand is met. Therefore, in the above-cited example, the demand of ₹3,000 is met with one note of ₹2,000 and two notes of ₹500, under the assumption that these notes are available in the ATM. It should be noted that if the entire demand cannot be met with the available inventory of currency notes, then it is considered to be unmet. That is a partial fulfilment of demand is not allowed.

An additional concern for the simulation model is the replenishment of the ATMs. The replenishment schedule (i.e., the frequency of visit and replenishment quantity) for all the ATMs that can be obtained by solving model IRP-A using the solution approach proposed in Section 3 can be incorporated into the simulation model. For the purpose, a set of homogenous vehicles is assumed to be available at the depot and used for the currency notes distribution as per the given schedule. The simulation environment gives an opportunity to consider uncertainties associated with the travel time and service time at the ATMs. Also, the replenishment quantity obtained may not be feasible in some cases due to the uncertainty in demand. In particular, if the estimated demand is less than the actual demand in a period, the remaining capacity in the ATM might not be sufficient enough to accommodate the indented replenishment quantity. In such cases, the replenishment quantity can be adjusted to the maximum allowed quantity by respecting the capacity constraint in the ATM. In this way, the simulation model is developed and run in the given problem environment.

It should be noted that the simulation model generates useful statistics regarding the customer arrival, demand rate, utilisation and service level at the ATMs by capturing the uncertainties in the real world. It also uses the replenishment schedule generated from the optimisation model that works with the estimated values of parameters such as customer demand. Now, if both approaches are coupled together, it would lead to an effective decision-making tool. For the purpose, we first run the optimisation model and generate the best possible replenishment schedule for the ATMs. Then the simulation model is run for a certain number of iterations. The performance of the simulation model and the values of the key parameter that is average aggregate demand during each period is given as an input to the optimisation model. Having known better estimates of the demand, the optimisation model tries to produce a better replenishment schedule. Based on that, the simulation model is re-run. Thus, the optimisation and simulation are run in an integrated fashion for a predefined number of iterations or till the achievement of a satisfactory level of performance measures. This leads to an integrated optimisation-simulation framework, as shown in Figure 3.

Figure 3 Proposed integrated optimisation-simulation approach (see online version for colours)



6 Computational experiment

The purpose of our computational experiments is multi-fold, as given below.

1 To validate the proposed mathematical model

2 To evaluate the computational efficiency of the proposed decomposition heuristic

3 To understand the effect of uncertainty in the selected parameters with the help of the simulation model

4 To judge the efficacy of the proposed simulation-optimisation framework.

The computational experiments are carried in two phases. In the first phase, random problem instances are generated to analyse the performance proposed decomposition heuristic approach. In the second phase, a case of a network of ATMs of a nationalised bank in Varanasi city in India is considered and the results of the proposed optimisation and simulation framework are discussed. The details of these experiments are given in the following sub-sections.

6.1 Computational experiments on random test instances

6.1.1 Dataset

For computational experiments, we generate ten different problem instances. Each problem instance is characterised by the number of nodes that include a single depot representing the central currency chest/nodal bank and different number of ATMs. The smallest instance has 10 ATMs and the largest has 100 ATMs. For the remaining instances, the number of ATMs is increased in the steps of 10. For each problem instance, locations of the depot and ATMs are generated randomly on a grid of 100 by 100. The distance between the nodes is assumed to be Euclidian and the travel time between them is calculated by dividing the distance with the average speed of the vehicle. The planning horizon is assumed to be of one, two and three days and the length of each period in the planning horizon is assumed to be three hours. With the three-different lengths of planning horizon under consideration, the total number of problem instances becomes 10 × 3 = 30. Then, for each ATM, the aggregate level demand is generated for each time period uniformly between 500 and 10,000. Further, three vehicles are assumed to available for cash replenishment in case of the smallest problem instance having ten ATMs. For each of the next instances, three more vehicles are added to the fleet.

6.1.2 Experimental settings

Each problem instance is first solved with the exact method using GUROBI 8.0 math programming solver with a python interface to get the optimal solution wherever possible. The decomposition-heuristic is also coded in Python and uses GUROBI 8.0 solver for the first phase of inventory management. A time limit of 3600 seconds is imposed on the instances for GUROBI as well as for the heuristic. If an optimal solution is not obtained, the upper bound and its gap from the best known lower bound is reported. These computational experiments are performed on a Laptop having specification Intel Core i7 9th generation processor running at 2.6 GHz with 8 GB RAM, Microsoft Window 10 and 64-bit operating system.

Table 1 Summary of results of the computational experiments for 8 periods, 16 periods and 24 periods before planning

Instance number	*Length of the planning horizon*	*Number of ATMs*	*Results of GUROBI*			*Results of decomposition heuristic*		*Gap (%)*
			Best known objective	*Diff. (%)*	*CPU time (sec)*	*Best known objective*	*CPU time (sec)*	
1	1 day (8 time periods)	10	67,254.2	0.00	0.45	67,254.2	0.44	0.00
2		20	133,771.6	0.00	6.44	133,772	0.86	0.00
3		30	195,092.2	0.00	116.87	195,092.2	1.43	0.00
4		40	263,004.6	0.00	644.79	263,004.6	1.97	0.00
5		50	331,931.8	0.00	1,299.32	331,931.8	2.98	0.00
6		60	397,121.4	0.42	3,602.15	397,122.6	3.82	0.00
7		70	461,543.8	0.78	3,605.11	461,537.2	4.16	0.00
8		80	537,828	5.99	3,605.22	529,351.2	6.19	–1.58
9		90	596,254.8	4.20	3,609.88	591,391.8	5.98	–0.82
10		100	24,017,744	--	3,641.75	658,220.4	8.16	–97.26
11	2 days (16 time periods)	10	128,808.0	0.0	12.08	128,808.0	1.27	0.00
12		20	261,650.4	0.0	1,444.94	261,650.4	3.39	0.00
13		30	391,476.8	14.5	3,600.70	391,476.8	5.33	0.00
14		40	523,429.4	42.9	3,601.64	523,084.8	8.96	–0.07
15		50	654,865.2	43.4	3,627.07	653,705.6	13.22	–0.18
16		60	823,644.8	45.8	3,604.11	784,609.2	24.44	–4.74
17		70	929,342.8	43.2	3,607.43	920,142.4	37.87	–0.99
18		80	38,297,268.0	288.7	3,653.75	1,047,270.0	48.09	–97.27
19		90	---	---	---	1,179,362.0	76.16	NA
20		100	---	---	---	1,318,339.0	93.61	NA
21	3 days (24 time periods)	10	192,868.6	0	250.04	192,870.0	2.56	0.00
22		20	389,164.6	0.02	3,600.58	389,164.6	5.66	0.00
23		30	584,639.2	---	3,600.47	583,378.0	11.37	–0.22
24		40	785,403.0	---	3,601.48	784,544.0	23.73	–0.11
25		50	1,057,281.0	---	3,615.94	973,355.4	26.50	–7.94
26		60	---	---	---	1,174,787.0	37.01	NA
27		70	---	---	---	1,370,333.0	67.58	NA
28		80	---	---	---	1,559,516.0	67.65	NA
29		90	---	---	---	1,749,341.0	116.83	NA
30		100						NA

Notes: -- indicates that a lower bound could not be obtained for the problem instance. NA indicates that the percentage gap cannot be calculated for the instance.

6.1.3 Results of computational experiments

Table 1 enlists the results of computational experiments for both GUROBI and the proposed decomposition heuristic. As stated earlier, the experiments are conducted for all the varieties of the length of the planning horizon and the number of ATMs that are given in columns 2 and 3, respectively. The results of GUROBI are summarised in columns 4 to 6.

For each problem instance, the best known objective function value (upper bound in this case) found within the imposed computational limit is reported in Column 4. The percentage difference between the best known upper bound and an established lower bound for each instance is expressed in column 5 and the solution time is recorded in column 6. Similarly, the best known objective function value and the solution time in seconds for the proposed decomposition heuristic is shown in columns 7 and 8, respectively. The last column of Table 1 indicates the solution gap which is a percentage difference between the best-known solutions between GUROBI (say f_G) and the proposed heuristic (say f_H). The gap is calculated using the following formula.

$$Gap(\%) = \frac{f_G - f_H}{f_G} \times 100$$

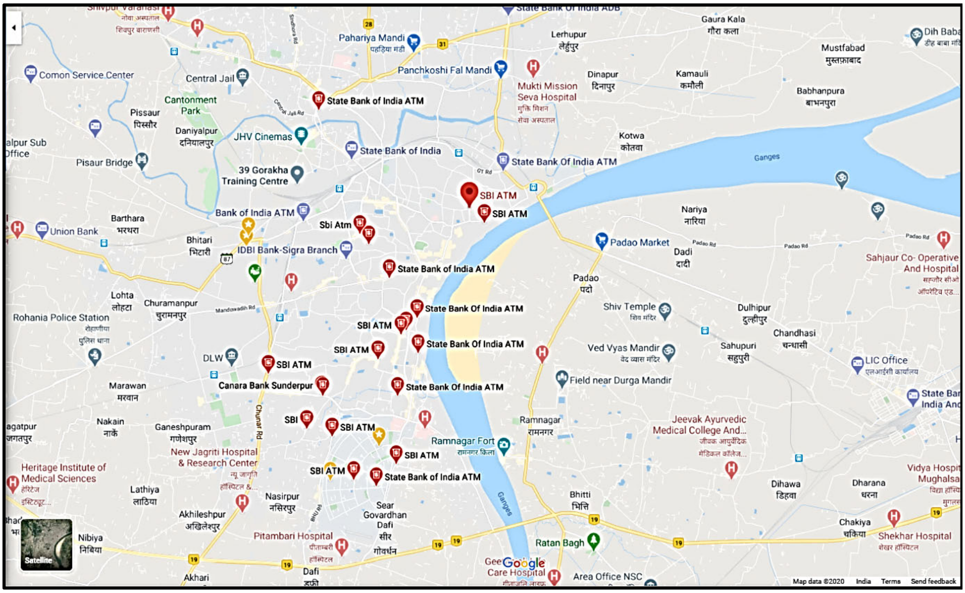
It is evident from Table 1 that GUROBI can solve a very few problem instances to optimality within the imposed time limit of 3600 seconds. Out of 30, it has solved only 11 instances to optimality or near-optimality. The performance of GUROBI deteriorates with an increase in the number of ATMs as well as number of periods in the planning horizon. For several large size problems, GUROBI was not able to even produce upper or lower bounds. On the contrary, the proposed decomposition heuristic exhibit both computational efficiency and quality of the solutions produced. It has produced the same solution as that of GUROBI for the small sized instances. For the medium and large size instances, the solution gap is negative which indicates that the proposed heuristic has produced a better solution as compared to GUROBI. Also, the proposed heuristic is characterised with the very small computational time as compared to GUROBI. The CPU times in seconds are single or double digit for most of the problem instances with a maximum recorded time of 116 seconds. Although, the CPU time for the proposed heuristic increases with increase in the problem size, this increase is found to be polynomial and not exponential. Thus, based on the results of the computational experiments we conclude that the proposed decomposition heuristic outperforms GUROBI.

6.2 An illustrative example from Varanasi City in India

For the face validation of our model, we opt to present an example inspired by practical settings in the Indian context. For the purpose, we identify a network of ATMs and a nodal branch of a major public sector bank in India in the city of Varanasi, Uttar Pradesh. Specifically, 19 ATMs and a nodal branch are considered, as shown in Figure 4.

The location of these ATMs is obtained using Google maps and the actual distance between them is calculated. The travel time between the ATMs is calculated considering the actual distance between them and the average speed of vehicles as 30 kmph. The length of the planning horizon is considered to be three days. Each day is further divided into eight time periods, each of length three hours. The rationale of such refinement is to account for the variations in the demand more precisely. Therefore, a total 24 periods are considered in the given planning horizon with the assumption that the first period starts at 12.00 AM on the first day. The replenishment of ATMs from 9 PM to 9 AM is restricted.

Figure 4 Location of selected ATMs in Varanasi city (see online version for colours)



The customer demand at the ATMs varies according to its location and the time period based on the estimates provided by the bank officials. The overall demand is varied in the range of ₹10,000 to ₹300,000. In each ATM, there are four cassettes out of which one cassette each is dedicated to currency notes of ₹2,000 and ₹500 and two cassettes are considered for ₹100. Each cassette can hold up to 2000 currency notes. Thus, the maximum capacity of an ATM is assumed to be ₹5,400,000. However, as per the directives from the governing bodies, an ATM is not supposed to stock more than ₹1,200,000 at any instance. It is assumed that the initial inventory of denominations of ₹2,000, 500 and 100 are in the range of 0–50, 50–500 and 500–1,000 notes, respectively.

On the logistics side, five vehicles are considered to be available for distribution of cash and each vehicle cannot carry cash more than ₹5,000,000 in a single visit. The fixed cost of visiting an ATM is set as ₹2,000 and the inventory carrying charges and shortages costs are considered to be 0.2% of the face value of denomination per period and 0.2 per unit of shortage, respectively. The data set is prepared in consultation with the bank officials and existing norms and the suitable changes are made for the illustration purpose.

The IRP-A model is for the considered example is solved using GUROBI as well as the proposed decomposition heuristic. GUROBI could not produce a feasible solution for it within the execution time of 3,600 hours. The example can be solved with the proposed heuristic approach within 18.94 seconds. The objective function value obtained by the heuristic is ₹358,015. As the demand is estimated beforehand, there are no incidence of shortages and hence no penalty cost is incurred. The major share in the cost component is the cost of visiting the ATM. It is ₹228,000, which is 63.8% of the total cost and the remaining is the cost of carrying the inventory. The replenishment of ATMs takes place in periods 4, 6, 12, 14, 20 and 22. Periods 4, 12 and 20 refer to the time slots 9–12 AM while periods 6, 14 and 22 refer to the time slot of 3–6 PM of the respective days. There are six paths formed in each of these periods and interestingly, the paths are identical. One such path formed by vehicle 2 in period 4 is shown in Figure 5.

Figure 5 Path formed by vehicle 2 in period 4 (see online version for colours)

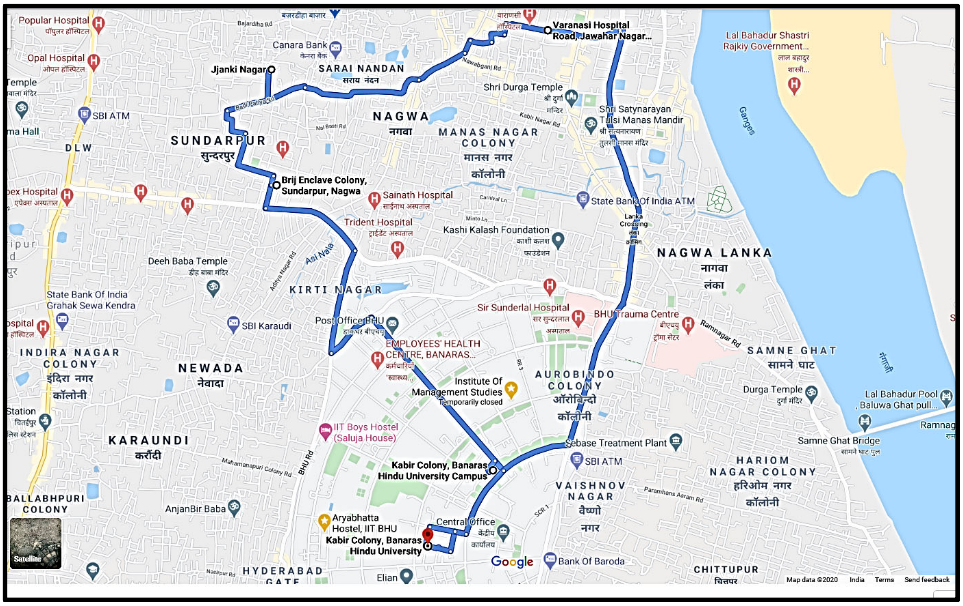
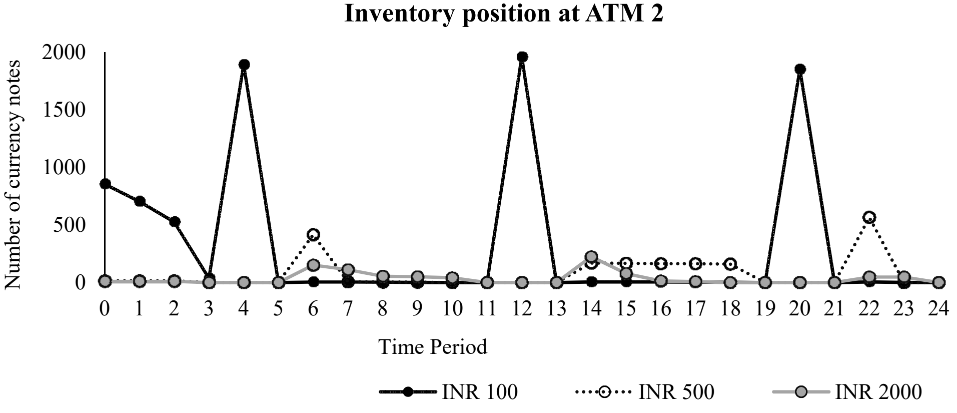


Figure 6 Inventory position of currency notes



It starts from the nodal bank and visits the ATMs 8-7-10-11 in this sequence. The same path is repeated for the remaining periods of visit as well. However, the replenishment quantities are different. For example, consider the ATM number 2 for illustration purposes. This ATM has an initial inventory of 10, 15 and 855 number of notes of denominations ₹2,000, 500 and 100, respectively. The inventory position of currency notes at this ATM over the period is shown in Figure 6. There are some interesting observations regarding the pattern of replenishment and withdrawal of the currency notes. The denomination of ₹100 is replenished in periods 4, 12 and 20, whereas the notes of denominations ₹500 and 2,000 are replenished in periods 6, 14 and 22. That means the replenishment and the use of the denominations are complementary to each other. The inventory of ₹100 bills is more than the other higher valued bills. It is also observed that the inventory pattern for ₹100 bills are sporadic in nature. These bills are consumed to meet the customer's demand in the period of replenishment and its subsequent period and then its inventory comes down to zero.

As the demand during each period is assumed to be known, the replenishment quantities are sufficient to meet the entire demand and there are no shortages observed. Further, due to the aggregation of the demand, the effect of individual customer orders and varied withdrawal patterns could not be posturised from the results of the model. For the purpose, the simulation model is developed, which as explained in the following.

As explained in Section 5, each ATM is envisioned as a single-serve and a simulation of the queuing model is developed for that. The inter-arrival times of the customers are varied according to the time slots. For the rush hours, they are considered in the range of [2–10] minutes, while for the odd hours, they are considered in the range of [20–40] minutes. The inter-arrival times are also varied depending upon the location of the ATMs. The service times, however, are considered to be uniformly varying across the ATMs and assumed to be in the range of [2–5] minutes for each customer. The demand of each customer is generated in the range of ₹500 to 10,000, which is the maximum limit of withdrawal from the ATMs. The simulation model is integrated with the optimisation model (the proposed decomposition heuristic in particular) to get the replenishment schedule of the ATMs. In each iteration, keeping the delivery schedule the same, the simulation is run ten times and its results are compiled. A total of five iterations of the simulation are carried out with different initial inventory values and the consequent delivery schedules. A sample result of simulation for one iteration for ATM number 2 and day 1 is presented in Table 2.

The results are shown only for a few customers for the brevity. The table indicates service start and end time for each customer along with the customer demand. The inventory position of the currency notes as a result of withdrawal and replenishment after each transaction is also presented in the table. The results of the simulation instantly highlight the imitation of IRP-A model in terms of fulfilment of the customer's demand. The first replenishment of currency notes is realised at 615th minute. Till then, customer's demand is met out of the initial inventory. As the initial inventory is insufficient to meet the entire customer's demand until the first shipment, the shortages are inevitable. However, there are instances of shortages (e.g., customer number 130, 150 and so on) where in spite of the sufficient quantity of cash, the customer's demand cannot be satisfied completely due to the unavailability of an appropriate number of currency bills. Service level is the metric used to account for the number of shortages in this context, which is defined as the ratio of the total number of customers whose demand cannot be fulfilled completely to the total number of customers visiting the ATM. The average service level of the ATMs for the five iterations of simulations is given in Table 3.

Table 2 Sample result of simulation for one iteration for ATM number 2 and day 1

Customer no.	*Timing (minute)*		*Demand*	*Withdrawal denominations*			*Replenishment*			*Inventory balance*		
	Service start	*Service end*		*2,000*	*500*	*100*	*2,000*	*500*	*100*	*2,000*	*500*	*100*
1	1	5	5,100	2	2	1	0	0	0	7	9	585
2	61	64	4,400	2	0	4	0	0	0	5	9	581
3	82	86	2,700	1	1	2	0	0	0	4	8	579
4	234	238	3,900	1	3	4	0	0	0	3	5	575
5	270	273	9,800	0	0	98	0	0	0	3	5	477
10	395	398	800	0	1	3	0	0	0	2	2	428
15	438	441	7,200	0	0	72	0	0	0	2	2	1
16	447	450	6,300	0	0	0	0	0	0	2	2	1
20	546	549	5,000	0	0	0	0	0	0	2	2	1
30	574	576	3,000	0	0	0	0	0	0	2	2	1
43	611	615	3,700	0	0	0	0	0	0	2	2	1
44	615	618	10,000	0	0	0	*0*	*0*	*3657*	2	2	3,658
45	618	620	6,900	0	0	69	0	0	0	2	2	3,589
60	745	747	6,900	0	0	69	0	0	0	2	2	2,893
70	774	777	3,800	0	0	38	0	0	0	2	2	2,319
100	939	942	6,600	0	0	66	0	0	0	2	2	890
118	988	992	1,900	0	0	0	0	0	0	2	2	2
119	992	994	5,300	0	0	0	*405*	*42*	*10*	407	44	12
120	994	997	3,800	1	3	3	0	0	0	406	41	9
130	1,025	1,027	3,800	0	0	0	0	0	0	395	35	1
144	1,066	1,068	2,000	1	0	0	0	0	0	393	33	1
150	1,096	1,098	4,300	0	0	0	0	0	0	393	33	1
170	1,157	1,159	8,100	0	0	0	0	0	0	380	28	1
180	1,187	1,191	8,300	0	0	0	0	0	0	370	26	1
190	1,217	1,220	8,900	0	0	0	0	0	0	365	23	1
200	1,266	1,270	9,500	4	3	0	0	0	0	352	14	1
210	1,309	1,311	5,200	0	0	0	0	0	0	345	8	1
222	1,357	1,361	9,000	222	1,357	1,361	0	0	0	338	6	1

Table 3 Average service levels (%) for the ATMs

ATM number	*Simulation iteration number*					*Average of averages*
	1	*2*	*3*	*4*	*5*	
2	90.35	91.21	93.20	74.32	82.45	86.31
3	71.85	92.51	93.94	89.49	39.43	77.44
4	93.23	73.05	81.99	90.40	90.88	85.91
5	89.59	71.78	74.61	88.56	91.75	83.26
6	96.52	89.69	89.88	94.97	75.47	89.31
7	92.19	83.59	44.34	94.12	90.76	81.00
8	91.92	92.00	48.58	92.22	89.06	82.76
9	90.15	91.35	94.39	96.36	90.54	92.56
10	89.85	67.53	88.31	90.01	88.48	84.84
11	87.85	89.07	90.56	78.64	91.99	87.62
12	49.76	90.83	95.09	94.71	98.96	85.87
13	94.92	90.41	91.25	91.84	80.64	89.81
14	70.70	70.00	86.73	98.38	90.35	83.23
15	88.43	90.74	91.16	87.57	93.95	90.37
16	88.37	92.93	91.73	89.25	86.88	89.83
17	88.78	87.07	87.85	91.82	94.18	89.94
18	92.81	90.88	91.50	93.94	82.96	90.42
19	80.16	89.91	97.26	94.41	99.20	92.19
20	97.83	87.18	86.06	77.65	57.49	81.24

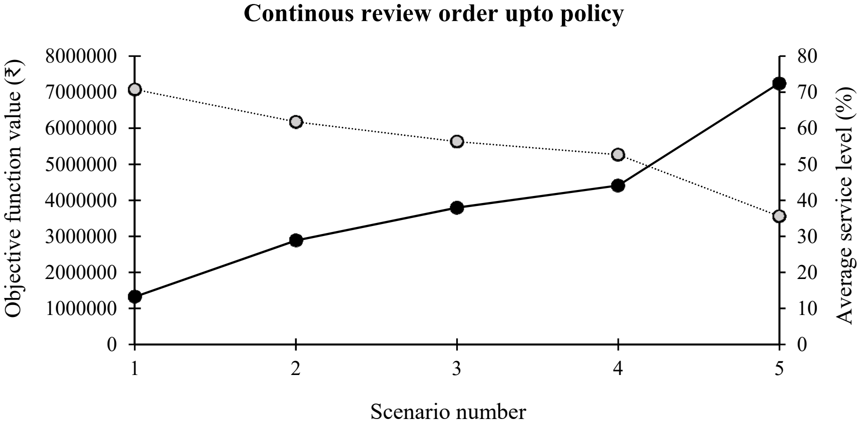
The objective function value obtained for different iterations of simulation is summarised in Table 4. For each iteration, ten simulations runs are worked out. The objective values of the individuals run and the average of all ten runs is indicated in Table 4. The best known objective function value obtained by solving the IRP-A using the proposed heuristic is also given in the last row of Table 4 for the comparison. These values exhibit a sharp contrast. The total cost (objective function value) of maintaining the inventory and replenishment of the ATMs in the simulation model is approximately ten times higher than its deterministic counterpart. Although the inventory carrying cost and the cost of visiting the ATMs is the same in both the approaches, the shortage costs arising due to unmet demands are much higher in the simulation model. Nevertheless, the simulation model helped in understanding the possible real-life scenarios and is helpful in estimating the level of services for the ATMs.

The output of the simulation model can further be used to improve the performance of the system. A ready use would be in determining the denomination mix strategies. As explained earlier, it is observed from the simulation mere a ‘sufficient’ amount of cash in the ATMs does not guaranty the 100% service level. A proper denomination mix can alleviate the problem. For example, consider the results of the first simulation run of ATM number 2. During the sixth period, the number of currency notes of ₹2,000, 500 and 100 replenished is 405, 42 and 11, respectively. This is the net worth of ₹832,100. The next replenishment is due in period 12. During this period, total 145 customers visit the ATM. Out of this, the demand of 118 customers could not be met, which lead to a service level of just 18.92% during this period. Moreover, 330 bills of ₹2,000 remains unused in the ATM. If following the approach of prioritising the higher bills for withdrawal is followed and the denomination mix is changing to 299 bills of ₹2,000, 230 bills of ₹500 and 327 bills of ₹100, then 100% of service level can be achieved during this period. The net worth of the modified denomination mix is ₹745,700, which is lower than the one determined from the IRP-A. In this case, the actual demand is lower than anticipated and the appropriate denomination mix leads to a higher level of service. In general, the different demands generated during different simulation runs can be averaged out to determine the appropriate denomination mix.

Table 4 Objective function values of 10 runs and 5 iterations

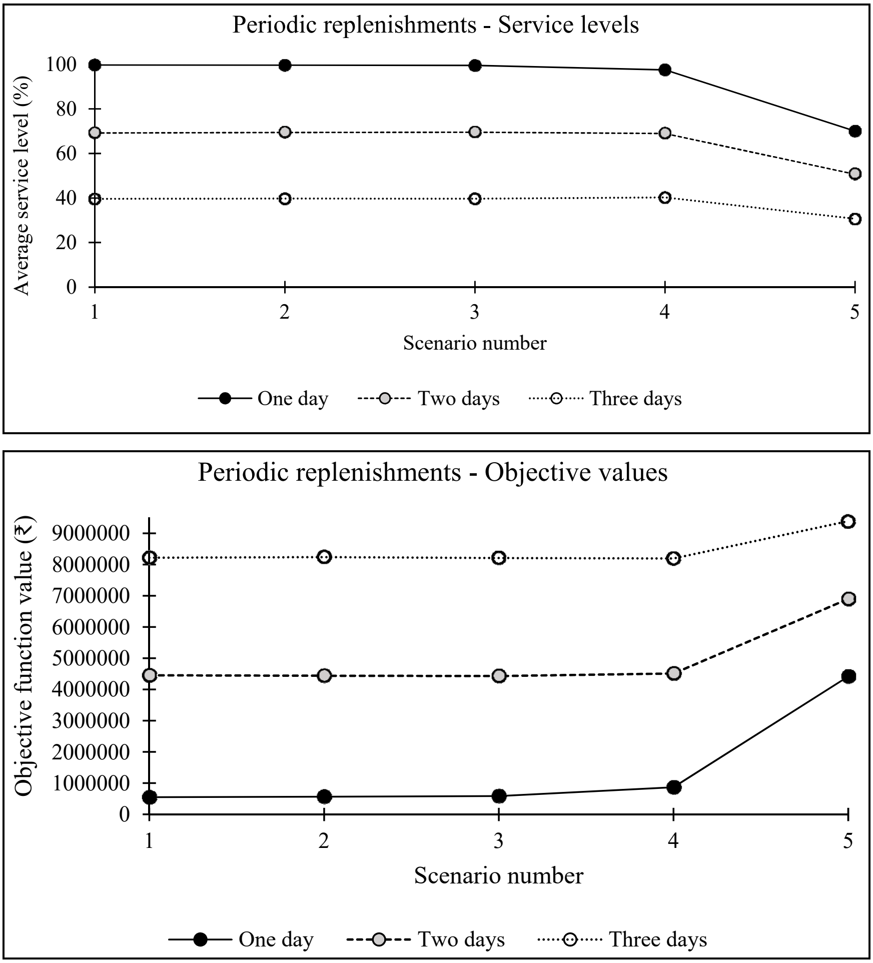
Simulation run	*Simulation iteration number*				
	Iteration 1	*Iteration 2*	*Iteration 3*	*Iteration 4*	*Iteration 5*
1	2,859,777	2,624,189	2,747,666	2,747,666	3,113,548
2	2,940,711	2,986,912	3,087,977	3,087,977	3,264,765
3	2,836,995	2,512,029	2,900,589	2,900,589	3,209,901
4	2,650,387	2,754,403	3,025,289	3,025,289	2,721,548
5	2,641,793	2,572,406	2,936,068	2,936,068	2,807,380
6	2,488,010	2,435,433	2,978,140	2,978,140	3,161,272
7	2,857,246	2,605,710	2,887,842	2,887,842	3,228,510
8	2,456,970	2,467,549	3,214,740	3,214,740	3,070,556
9	2,639,698	2,861,644	2,879,861	2,879,861	3,270,449
10	2,620,788	2,663,479	2,773,610	2,773,610	3,008,587
Average	*26,99,238*	*26,48,375*	*29,43,178*	*29,43,178*	*30,85,652*
Optimal objective value of IRP-A	3,64,229	3,73,785	3,80,607	3,74,655	3,61,603

Figure 7 Results of continuous review order up to policy



Next, the outcomes of the proposed optimisation-simulation framework are compared with the existing continuous review order up to policy and periodic replenishments such as one, two or three days replenishment policies for each ATMs. For the continuous review policy, the reorder point is set as ₹50,000. Thus, when the net worth of cash in the inventory falls below the reorder point, the replenishment process is initiated and the ATM is refilled to reach the maximum allowed quantity of cash that is set as ₹1,200,000. The logistics service provider carries out the necessary steps towards the replenishment of the concerned ATM. This policy becomes interesting due to the hard time windows available for replenishment and lead time associated. The lead time includes order preparation, processing, cash collection, transportation and actual replenishment of ATMs. In practice, this lead time is also highly uncertain that may vary from a couple of hours to days. Therefore, for the given reorder point and the same level of initial inventory and demand, we consider five different variations in the lead time as 3, 6, 9, 12 and more than 12 hours. The results of the continuous review order up to policy are as shown in Figure 7.

Figure 8 Performance of periodic replenishment policies



It is evident from Figure 7 that the continuous review order up to policy is effective only if the leads time are considerably short (scenario 1 and 2 that implies the replenishment of the ATMs within six hours of reaching to the reorder level). In this case, the objective function values are the lowest and the average service levels of all the ATMs are above 60%. With the increase in lead times up to 12 hours, the objective function value doubles and the average service level deteriorates up to 50%. In the last scenario, where the lead times are more than 12 hours indicates the worst results with a tri-fold increase in the objective value and drop in the service levels up to 35%.

The order up to policies can also be implemented with periodic replenishment, such as every day replenishment, once in two days and once in three days replenishment policies for suitability in practice. The uncertainty associated with the lead time also prevails in these replenishment practices. The results are depicted in Figure 8. It clearly indicates that everyday replenishment policy outperforms the once in two and three days replenishment policies in terms of lower objective function value and higher service levels. Service levels of this policy are almost 100% in the first four scenarios, where the lead time is less than 12 hours. The higher inventory carrying cost is compensated with the improved level of service and consequently, the reduced shortage costs. It can also be seen that there is no significant difference in the cost or service levels for a policy if the lead time is varied within 12 hours. The only practical difficulty associated with this policy is a strict requirement of visiting all the ATMs on the designated periods. With the limited number of vehicles available and the hard time windows for replenishment imposes hurdles for this policy and the competitive advantage could not be obtained. This is evident from scenario 5 for all the variations of the policy.

When the results of the reorder point order up to policy and the periodic replenishment policies are compared with that of the proposed model, it can be seen that the result of the proposed model is quite robust. As an inventory replenishment plan for the entire planning horizon can be worked out beforehand, it eases the replenishment procedure that involves a third party. The proposed model results in a lower cost and an improved average service level as compared to the scenario 4 and 5 of the order up to policy and the periodic replenishment policies that are close to the reality. The order up to policy and the periodic replenishment policies are only effective with the shorter lead times and with the sufficient number of vehicles to visit all ATMs at the stipulated time.

7 Conclusions and future scope of work

This paper has presented a rich variant of IRP that is motivated from the problem of cash management and replenishment of ATMs in India. A mixed-integer programming model is presented and solved using a two-phase iterative decomposition heuristic. It is observed that the proposed heuristic approach outperforms the state of the art math programming solver GUROBI for medium and large size problem instances. To take into account the uncertainties associated with the customer's arrival and demand, a single server queuing model is also developed for each ATM over the entire planning horizon. The output of the optimisation model (replenishment plans) are given as an input to the simulation model. The simulation model is useful in understanding the impact of replenishment quantity and denomination mix on the service level. It is observed that the denomination mix plays a crucial role in the service level as due to unavailability of certain denominations may lead to unfulfilment of the demands. The data gathered from different simulation runs is useful in estimating the aggregate demand during a period as well as the appropriate denomination mix. Further, the proposed optimisation-simulation approach seems to be practical and it outperform the conventional continuous review-order up to policy and the periodic replenishment policies.

There are several possible extensions to the present work. The present work has been from the point of view of the public sector banks in India. A totally different perspective can be adopted if the problem is considered from the viewpoint of a thirds party service provider.

Instead of considering only conventional ATMs, a mix of conventional and recycle ATMs that allows both withdrawal and deposits can be considered. The problem would be considerably complex due to the different denomination mix and pick-up and deliveries in this hybrid system. The present work suggest the optimal policy for the existing system that is the use of third party service provider for cash logistics. It would be interesting to see the results if the bank owns the vehicles or carries out the cash handling part on its own. Other practical considerations such as non-repetitive routes for the safety considerations and the use of real time geospatial data would enrich the present work. On the modelling side, use of a scenario based robust optimisation approach and consideration of value at risk or conditional value of risk would be another avenues to extend the present work. Further, to boost the cashless transactions, banks are trying to reduce the number of ATMs. On the other hand, use of ATMs in the country is continuously increasing. This scenario demands a study on network design and inventory policies. We plan to take these issues as a future work.

References

Anbuudayasankar, S.P., Ganesh, K., Koh, S.L. and Ducq, Y. (2012) 'Modified savings heuristics and genetic algorithm for bi-objective vehicle routing problem with forced backhauls', *Expert Systems with Applications*, Vol. 39, No. 3, pp.2296–2305.

Andersson, H., Hoff, A., Christiansen, M., Hasle, G. and Løkketangen, A. (2010) 'Industrial aspects and literature survey: Combined inventory management and routing', *Computers & Operations Research*, Vol. 37, No. 9, pp.1515–1536.

Batı, Ş. and Gözüpek, D. (2017) 'Joint optimization of cash management and routing for new-generation automated teller machine networks', *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, Vol. 49, No. 12, pp.2724–2738.

Boonsam, P., Suthikarnnarunai, N. and Chitphaiboon, W. (2011) 'Assignment problem and vehicle routing problem for an improvement of cash distribution', in *Proceedings of the World Congress on Engineering and Computer Science*, October, Vol. 2, pp.1160–1164.

Coelho, L.C., Cordeau, J.F. and Laporte, G. (2014) 'Thirty years of inventory routing', *Transportation Science*, Vol. 48, No. 1, pp.1–19.

Ekinci, Y., Serban, N. and Duman, E. (2019) 'Optimal ATM replenishment policies under demand uncertainty', *Operational Research*, Vol. 21, pp.999–1029.

Kurdel, P. and Sebestyénová, J. (2013) 'Routing optimization for ATM cash replenishment', *International Journal of Computers*, Vol. 7, No. 4, pp.135–144.

Larrain, H., Coelho, L.C. and Cataldo, A. (2017) 'A variable MIP neighborhood descent algorithm for managing inventory and distribution of cash in automated teller machines', *Computers & Operations Research*, Vol. 85, pp.22–31.

Michallet, J., Prins, C., Amodeo, L., Yalaoui, F. and Vitry, G. (2014) 'Multi-start iterated local search for the periodic vehicle routing problem with time windows and time spread constraints on services', *Computers & Operations Research*, Vol. 41, pp.196–207.

Min, D. and Chong, L.X. (2012) 'An improved ant colony algorithm for single vehicle route optimization', *Journal of Computational Information Systems*, Vol. 15, No. 5, pp.3963–3969.

Sofianopoulou, S. and Mitsopoulos, I. (2018) 'A review and classification of heuristic algorithms for the inventory routing problem', *International Journal of Operational Research*, Vol. 41, No. 2, pp.282–298.

Talarico, L., Sörensen, K. and Springael, J. (2015) 'The k-dissimilar vehicle routing problem', *European Journal of Operational Research*, Vol. 244, No. 1, pp.129–140.

Toth, P. and Vigo, D. (Eds.) (2002) *The Vehicle Routing Problem*, Society for Industrial and Applied Mathematics, Philadelphia.

Van Anholt, R.G., Coelho, L.C., Laporte, G. and Vis, I.F. (2016) 'An inventory-routing problem with pickups and deliveries arising in the replenishment of automated teller machines', *Transportation Science*, Vol. 50, No. 3, pp.1077–1091.

Van der Heide, L.M., Coelho, L.C., Vis, I.F. and van Anholt, R.G. (2020) 'Replenishment and denomination mix of automated teller machines with dynamic forecast demands', *Computers & Operations Research*, Vol. 114, p.104828.

Wagner, H.M. and Whitin, T.M. (1958) 'Dynamic version of the economic lot size model', *Management Science*, Vol. 5, No. 1, pp.89–96.

Xu, G., Li, Y., Szeto, W.Y. and Li, J. (2019) 'A cash transportation vehicle routing problem with combinations of different cash denominations', *International Transactions in Operational Research*, Vol. 26, No. 6, pp.2179–2198.

Zhang, Y. and Kulkarni, V. (2018) 'Automated teller machine replenishment policies with submodular costs', *Manufacturing & Service Operations Management*, Vol. 20, No. 3, pp.517–530.