RFID-enabled flexible warehousing

Wei Zhou, Selwyn Piramuthu, Feng Chu, Chengbin Chu

RFID-Enabled Flexible Warehousing

Wei Zhou[1], Selwyn Piramuthu[2], Feng Chu[3,4],Chengbin Chu[5,6]
[1]Information & Operations Management, ESCP Europe, Paris, France
[2]Information Systems and Operations Management, University of Florida, Gainesville, Florida 32611, USA
[3] Laboratory IBISC, University of Evry-Val d'Essonne, Evry 91020, France
[4] Management Engineering Center, Xihua University, 999 Jin Zhou Rd, Chengdu 610039, Sichuan Province, China
[5]School of Economics and Management, Tongji University, 1239 Siping Road, Shanghai 200092, China
[6]Laboratoire Genie Industriel, CentraleSuplec, Universit Paris-Saclay, Grande Voie des Vignes, 92290 Chatenay-Malabry, France
[1]wzhou@escpeurope.eu, [2]selwyn@ufl.edu,[3]feng.chu@ibisc.univ-evry.fr,[5]chengbin.chu@ecp.fr

Abstract

We propose a smart warehouse environment where not only inventory items but also the shelves are tracked by an RFID-based system. Both operational activities and warehouse configurations are continually monitored to facilitate real-time response. We study the dynamics of a flexible warehouse scenario where items of any type can be dropped off anywhere within the premises. Unlike existing models, we relax both the location constraint and local (e.g., item-type level) capacity constraints with a periodically renewable fixed global capacity. Dynamic decisions on location and local capacity are made based on the stochastic Markovian demand states. We optimize processing and routing constraints and compare the performance of this flexible storage setup with classical models through multiple levels of real-time decision support. Our results provide corroborating evidence to support the following observations: (1) "free pick-n-drop" combined with fluid warehousing mechanism greatly reduces trip costs and lead time for single trip demand, (2) there exists a lower bound on the performance in such a setup with fixed local capacities, and (3) the lower bound can be further improved when inventory capacity and location are dynamically adjusted according to actual demand patterns.

Keywords: Flexible Warehousing, Smart Logistics, Knowledge-Based System

1 Introduction

With the presence of multiple intertwining business processes, complexity-induced errors are common in warehouse management. While such errors are reduced through intelligent automation, the complexity of processes that deal with a large number of tools, work-in-process, and raw materials contributes to this reality. For example, in large automobile manufacturing plants, a significant amount of resource wastage occurs due to misplaced items and containers (Hanebeck and Lunani 2008). Automobile manufacturers spend millions of dollars each year just to retrieve and to replace missing containers in a plant. Operational uncertainties from both demand and supply sides are associated with deleterious impact on logistics services that value high efficiency and low cost. In well-managed manufacturing plants, adopted common practice such as the al-

location of specific locations for each item type where the items are sorted as well as adherence to 5S principles lower the probability of errors.

Randomness in a complex operational system is manageable through today's IoT-based tracking technologies. Enabled by RFID, the provision of a more efficient service can be achieved through a logistic decision support system that is different from those in existing literature. In this study, we are motivated to investigate such a system by allowing traceability on both inventory items and warehouse storage equipment. Existing literature on logistics traceability are mostly focused on tracing inventory items (e.g., CAkiCi et al. 2011, Chou et al. 2007, Tu et al. 2009). We find it equally important to track and trace storage equipment based on inventory flow information. A "smart" self adjusting warehouse is able to manage more operational randomness, a characteristic that can benefit a wide spectrum of business applications from workshop management in SMEs to large-scale production inventory management in big firms.

Difficulties of managing inventory in large complex operational environments originate from randomness and various intertwining business processes (e.g., Thiétart and Forgues 1995) because it is hard to monitor, collect, and process necessary information and to make real time decisions based on seemingly random information (e.g., Simon 1959). Less information uncertainty, simpler operational routine and work space assignment result in reduced human errors and higher quality output based on operational production and services (e.g., Akerlof 1970). Appropriate rules are needed to ensure a relatively simple and clean working environment in order to facilitate the core operational tasks, although it creates additional costs (e.g., Hasle et al. 2012). Well-structured warehouses, carefully designed inventory routines and locations, and strict need-to-follow operational schedules create various costs that are significant to the organizations. In inventory management, routine inventory check, warehouse shelf optimization, item assortment and item localization demand intense labor investment and could also lead to lag in lead times (e.g., Rosenblatt et al. 1993). To minimize uncertain and inaccurate information in inventory, transportation, and production processes, the traditional means have been to adopt a set of optimization rules that simplify the processes. It is costly to maintain pre-scheduled orders and rules, including transaction costs, additional trip costs, and prolonged operational lead time.

We investigate the possibility of relaxing these rules with minimal interference on the core jobs. In applications where pick-up and drop-off of multiple item-types occur, for example, it is traditionally common for similar items to be grouped together in one location to facilitate ease of retrieval. Such a setup generally faces two issues: (a) drop-off incurs both search cost to identify the location to drop off as well as the drop-off cost itself and (b) expansion to

facilitate overflow drop-offs may not be easy to accomplish. To address issues created by these rigid-rules, we introduce flexibility in inventory system by incorporating moveable warehouse shelves as trackable items. While existing methods do not seem to solve today's increasing complexity of warehouse and production-site management problems, IoT/RFID enabled tracking technologies can help reduce such problems with increased information transparency in real-time. In our proposed system, antenna-equipped moveable warehouse shelves are technologically feasible whereby either the whole or parts of the shelves can be reallocated to another location at low operational cost. Equipped with a reader, these intelligent shelves integrate to form a Machine to Machine (M2M) network. Based on RFID-enabled tracking/tracing capability, it is easy to acquire and possess complete information on both item-level inventory and each item's exact location in the warehouse. When an RFID-tagged item is freely dropped off or picked up, the system is aware of its location change and new status (e.g., in working order or temporarily out of service). Similar information with regard to moveable warehouse equipment can also be readily monitored. We study a novel concept of flexible warehousing by relaxing some of the most commonly adopted rules and practices, which were traditionally built upon the economy-of-scale. In summary, the contribution of our approach is twofold. Firstly, we propose a free-pick-and-drop mechanism that allows workers to freely (or even randomly) move inventory items at the highest convenience without any need to remember or search for exact drop-off locations, thanks to the RFID-based tracking system. Secondly, we propose a fluid warehousing concept that enables adjustment of warehousing configurations (location, capacity, routine, etc.) in real-time according to the dynamics of demand. The fluid warehousing part is also enabled by tracking/tracing technology on shelves and containers that allows the information system to establish and remain in real-time contact with (RFID-)tagged items.

We model and validate the proposed flexible warehousing concept based on a classical facility location problem setting where we attempt to understand its dynamics and performance by gradually relaxing the parameters on location, local capacity, and item pick-up/return policies. Then, we consider associating/disassociating item_identification and storage_identification information in real-time as necessary and appropriate. This association is temporary compared to that in traditional storage/item association where it is kept permanently bound. In the classical model, the placement of an item in the 'wrong' shelf or container would result in information discrepancy and, in many cases, inventory shrinkage. Traditionally, in order to reduce error and consequently inventory shrinkage, the arrangement and movement of physical goods and components in a production environment is characterized by a set of previously designed routines and rules so that any issue that arises in terms of locating goods/materials/tools is controlled.

We argue that such a setting, while simultaneously bringing certainty to the management of the system, results in additional cost and efficiency loss. For example, when a given component is stored at a pre-specified location in the warehouse, it signifies that each operational transaction related to this component should be traced or be designated back to this location. There is an associated cost that arises as a direct consequence of detouring, sorting and returning items. Unlike in traditional literature, we don't assume the type of inventory items. Instead, we consider a large variety of tools, machines, work-in-process, containers, and storage shelves that may be transferred on site. The way to effectively manage the inventory of a large number of different types of items is also different from managing inventory of a single type.

The remainder of this paper is organized as follows. We begin with a brief review of related literature and practices in the next section. We then present the model formulation and structural results in the three different systems 1. transaction support, 2. decision support, and 3. knowledge-based system in Section 3. In Section 4, we discuss numerical analysis results and evaluate the performance of the proposed mechanism. We conclude the paper in Section 5 with discussion on the results and possible future directions of this research.

2 Related Literature

Some parts of the proposed flexible warehousing concept have already been applied in practice. For example, in Paris, Vélib' customers can rent bicycles by picking and returning bicycles from their nearest Vélib' depots. The setting considered shares some similarity with bicycle sharing systems from the perspective of pick-up and drop-off at any (different) location in that network. Some of these bicycle sharing systems comprise several features that include the facility to pick-up as well as drop-off bicycles at any of the several stations that are distributed throughout the city (e.g., DiMaio 2009). Smooth operation of such bicycle sharing systems requires each of the stations to have enough bicycles that are fit and ready to be used as well as enough free space for returning bicycles. Violation of such constraints (i.e., not enough bicycles or bicycle return space) necessitates bicycle renters to travel to stations at nearby locations and incur unnecessary associated inconvenience. To minimize the possibility of occurrence of such situations, bicycle stations do their best to ensure that none of the stations are empty or full to allow for timely departure and arrival of rental bicycles. For example, Vélib' in Paris began with around 1500 stations, 20000 bikes, and about 70000 travels each day. The vélib' system balances the stations with the use of 20 trailer-trucks that transport bicycles. These trucks ply among stations to maintain the ratio of the total number of available bicycle parking places and the number of

available bicycles at each station close to a pre-determined value (e.g., Benchimol et al. 2011, Nair et al. 2013). In 2011, the Paris city hall introduced another service called *autolib'* that enables customers to pickup and return rental cars at autolib' charging points or kiosks that are distributed all over Paris (Martin and Darpy 2015, Tironi 2015).

Another example is the physical telephone directory in which once the initial setup cost is incurred to sort all names in alphabetical order and then the directory is printed, it is easy to search for any given name (vs. in a scenario where names are printed in random order). A similar parallel exists in sequential database storage. In the directory and sequential database examples, 'work' (sort based on some index and then identify the exact physical disk or tape location where a given record is stored) is incurred during 'drop-offs.' However, the main distinction in the warehouse, Vélib', and autolib' vs. printed telephone directory and sequential database is in the number of times 'pick-up' of an item instance occurs. In the former, it's just once for each physical item instance from a given location whereas for the latter it could be any number of times which in fact is the primary source of efficiency since the resource spent at the 'drop-off' stage pays off (or, is divided among instances) at the 'pick-up' stage.

One additional recent example is that of Amazon's 'chaotic' warehouses where items are stored in individually bar-coded shelf locations without regard to their types (e.g., a toy item stored next to a stationery item). Similar to that at Amazon, Lego's fully automated warehouses have operated without human interferences for about 20 years now. In Amazon's, Lego's and some fast fashion automated warehousing setting, the demand location and quantity are statistically stable, with these demands generated at the main inbound and outbound gates. Such automated systems are relatively easier to manage because of their statistical certainty and limited number of choices. In our study, we investigate a more general warehousing setting where demands are internally generated from daily job scheduling and are subject to human interferences and other uncertainties. Typical example can be found at an automobile manufacturing/repairing workshop or a congested health care environment where inventory buffer areas are highly sparse and limited and the demand is highly uncertain. In our case with random jobs, human interference along with other important indicators, a system such as Lego's fully robotic warehouse does not work and we need a data-driven mechanism that brings flexibility to warehouse and inventory management.

The considered mechanism is rooted in the classical facilities location problem. Given that there are several excellent review papers that provide a good overview of existing research literature on facilities location (e.g., Balakrishnan and Cheng 1998, Farahani et al. 2009, Snyder 2006), we do not attempt to repeat the same here. While some of these existing research work

consider the facilities location problem from a static perspective, others address the problem from a dynamic perspective. The primary difference between static and dynamic perspectives is in the identification of appropriate locations as a one-shot process in the former, while the latter involves a temporal component wherein facilities are incrementally added over time. For example, Current et al. (1998) consider dynamic models that are implicit as well as explicit, where the former are essentially static where all facilities are opened simultaneously and remain open over the planning horizon and in the latter the facilities are incrementally opened over time. Sana (2011) consider a model that investigate the impact of business strategies such as optimal order size of raw materials, production rate and unit production cost, and idle times in different sectors on collaborating marketing system, with an analytical method to optimize the production rate and raw material order size for maximum expected average profit.

Existing literature considers dynamic facility location from different perspectives. For example, Fotakis (2004) applies a novel merger rule to the k-median facility location problem with a guaranteed constant performance ratio for $O(k)$ medians. For k facilities, the algorithm is defined to be R-competitive when the cost is at most R times the optimum cost, and *competitive-ratio* is the smallest such R. Mettu and Plaxton (2003) consider incremental uncapacitated k-median problem and develop a polynomial-time algorithm with a competitive ratio of 29.86. Similarly, Charikar et al. (2004) develop a deterministic 8-competitive algorithm for incremental clustering. They also develop a 2e-competitive algorithm for the hierarchical k-center problem. Gonzalez (2005) develops a 2-competitive algorithm for an incremental version of the facility location problem. Dasgupta et al. (2005) study the hierarchical k-center problem and develop a competitive hierarchical clustering algorithm. Plaxton (2006) develops a 12.16-competitive approximation algorithm for the incremental uncapacitated facility location problem. With a general framework for cardinality constrained problem, Lin et al. (2010) extend this and develop algorithms for k-vertex, k-set cover, k-median, and k-spanning tree, and show polynomial time computation of a 16-competitive incremental median sequence. Hartline and Sharp (2006) consider scenarios where the input varies across time, and develop a general model that converts conventional algorithms to incremental algorithms with only a constant factor loss in approximation power. McCutchen and Khuller (2008) develop streaming algorithm with a constant factor approximation to the cluster radius for two variants of the k-center clustering problem, where each cluster is required to contain at least a predetermined number of input points. Albareda-Sambola et al. (2009) study the multi-period incremental service facility location problem with a set number of new facilities that are introduced over a finite time horizon in order to dynamically satisfy demand generated by customers. Friggstad and Salavatipour (2011)

consider the mobile factory location problem with the goal to minimize the movement of factory locations. Basu et Nair (2014) investigate the portfolio effect of a multi-period inventory control system. Shang et al (2008) propose an effective DSS system to determine the safety stock level and the number of weeks forward coverage (WFC) for each SKU. Atasoy et al (2012) consider the production/inventory problem of a manufacturer (or a retailer) under non-stationary and stochastic supply availability.

We consider a blend of both the static and the dynamic variations in which the number of locations (of buffers, shelves, or silos) is fixed, whereas the locations themselves are mobile as per the dictates of the operations environment (e.g., warehouse, manufacturing shop-floor). The considered mechanism is different from the concepts of emergency hub and expedite hub. There is no need for prior planning. If the nearest silo is not available for service, operators find the second or third nearest silos. The operators can either rely on direct line-of-sight for identifying the available silos/items or utilize hand-held devices to provide real-time inventory information. In both emergency hub and expedite hub, explicit short-range planning and coordination is almost always required to insure the continuality and service quality for the logistic service network.

In addition to the location of the facilities themselves, we consider an environment where every item (e.g., products, tools, shelves) is item-level RFID-tagged, and the location of each of these items are in turn known in real-time with a reasonable degree of accuracy. The dynamics associated with item-level RFID tags have been extensively studied by researchers during the last decade (e.g., Bose et al. 2008, Bose et al. 2011, Meiller et al. 2011, Tu et al. 2009). Iravani et al. (2014) study the optimal control of process and inventory flexibility where process flexibility is modeled as a multi-functional production facility that produces up to two types of products. Inventory flexibility is modeled as made-to-stock firm-driven one-way product substitution. Zhou et al. (2009) study a retailing decision support system with dynamic pricing and retail shelf inventory management, and the results show significant profit increase if the retail stores can dynamically alter the price and the inventory level based on the real-time profile of on-site customers. Another related scenario is that of ambulance location and relocation models with or without the use of GIS (Geographical Information Systems). Brotcorne et al. (2003) provide a good overview of these models and observe that dynamic models (e.g., Farahani et al. 2009) that depend on sophisticated and accurate search heuristics are required to not only include the expected cost incurred when no suitable ambulance is available to answer a call, but also to incorporate variations in travel times during the day when computing shortest paths. The number of ambulance vehicles is determined and fixed at a certain level in the beginning,

and the location of each of these are dynamically varied as per various criteria that include instantaneous demand at various locations, forecasts based on the time of day as well as day of the week, among others.

3 Model Formulation & Structural Results

We begin our study by first fixing mobile and trackable shelves at candidate locations at the work site (e.g., manufacturing shop floor). These RFID-enabled shelves can be freely located/integrated at a set of candidate locations, creating dynamic local inventory capacities. When an item is picked up from the shelf, the information system immediately receives this information; the system also receives information in real-time when an item is dropped off. Unlike those defined in traditional inventory management literature, we don't limit the type of our inventory items. Instead, we consider a large variety of tools, machines, work-in-process, containers, and storage shelves that may change locations on site at any time.

A traditional inventory warehouse with a set of inventory items is illustrated in Figure 1. The inventory space is divided and identified by an inventory_ID (for example, the first row first column has ID "ABC123") and the items are identified respectively by a product_ID. A set of inventory_ID and product_ID association is defined so that we know what goes where. For example, item "ABC123DF" is always stored in the "ABC123" storage area. In this example, we identify items at the categorical level instead of the item level. It signifies that we may find multiple items that share the same product_ID.

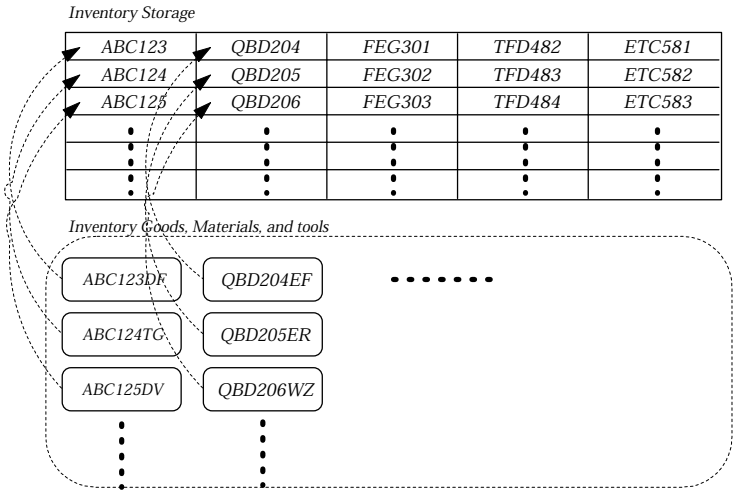

Figure 1: Product and Inventory Storage Association

Inventory demands can be generally categorized as inbound (to fetch) and outbound (to return). It is relatively much easier to manage the demands that are fulfilled at a few pre-assigned locations. Some of the recent advances in automated warehouse systems, such as those in Amazon and Lego, take advantage of this nice property when their inbound/outbound demands are clustered in a few facility locations (e.g. the collection/distribution port). In a manufacturing shop-floor context, the cost of physically associating product_ID with the inventory_ID is significant due to the time it takes to locate the correct storage location for retrieving or returning an item. Demands to retrieve an inventory item may occur at locations that are purely random. The search cost for an item includes the time and effort to locate an item that is available with its location information. The cost to fetch an item is the effort and lead time spent on the trip. The last cost is the returning cost, which is incurred when the jobs are completed and tools need to be returned to the original inventory location according to the association rule. When combined together, these three individual costs are generally significant.

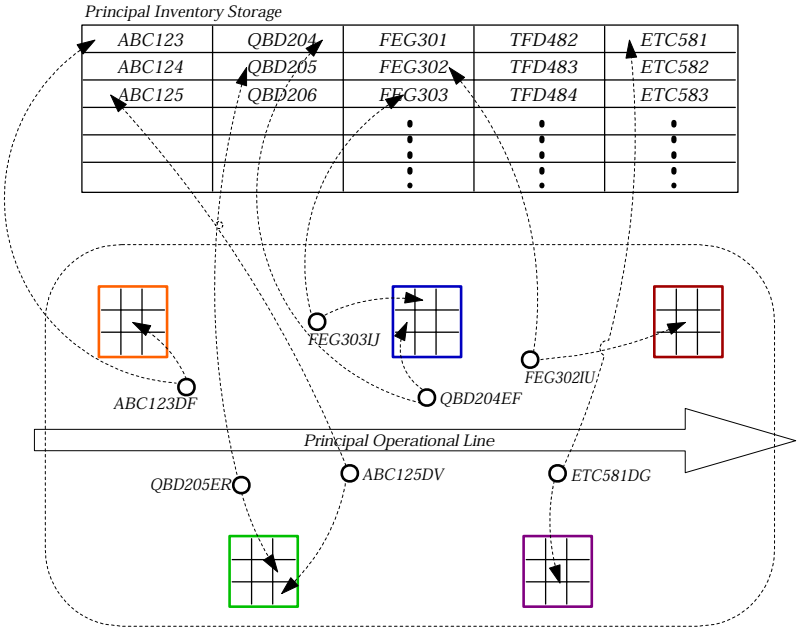

Figure 2: Free PicknDrop & Broken Product/Inventory Association

In the model/mechanism that we propose, we argue that if a worker is enabled to freely pick-up and drop-off items at the nearest convenient location it will greatly reduce the association cost as in the classical logistics system. In such a system, decisions on which silo to drop off items is decentralized, but the information is available in a centralized database and the decisions on the location and capacity of silos are also centralized. We illustrate the considered mechanism

by placing five drop-off silos (e.g., shelves) around the principal operational line (Figure 2). At each silo, there are designated drop-off areas with tracking antennas that communicate with the RFID tags. An item can be freely dropped off at the nearest location as per convenience, with a path as indicated by the dotted lines. Similarly, to fetch an item, information on the nearest storage area where this item is available can be readily displayed on a hand-held mobile device. The principal inventory storage is serviced by a distribution center in a remote area while the item/storage affiliation information is stored in the database. The inventory information can then be used to fetch the item either manually or through automated means. Through such a setting, the overall cost associated with operations as well as time required to search, fetch, and return are reduced.

While the first element in the considered system is to allow for free item pick-up and drop-off at the closest available location, the second element is to ensure a truly flexible warehouse operation where local capacity (i.e., capacity of silos at any given location) and inventory location (i.e., the location of the mobile silos themselves) are able to adapt according to the inventory service demand patterns. In what follows, we study the impact on cost and lead time with the considered warehousing mechanism. We also discuss the various conditions, their corresponding optimal strategies and performances. To operationalize this, we first consider a scenario where the location of all storage silos are fixed and the capacity is constrained. We then extend the static problem by allowing the trackable shelves to relocate and relaxing the fixed local capacity assumptions.

3.1 Once-for-all Transaction Support System

In the transaction support system, we consider a single stage inventory management problem that utilizes item level information to support individual transactions, such as to search the database to retrieve the inventory availability, to register a drop-off transaction, etc. It indicates a scenario where both the silos and the inventory items in this system are tracked in real-time at the item-level with minimal additional operational cost. Storage distribution is initially determined and then fixed, after which the individual silos are not allowed to relocate during operations. Items are tracked and traced and are allowed to be placed randomly at any available silo on site. Based on existing technology that is both cost-effective and feasible, all inventory items are RFID-tagged, which enables the items to remain visible in the system at all times with minimal additional operational cost. Once the fixed cost of the infrastructure (e.g., back-end systems, readers) are taken into account, the additional cost (e.g., RFID tags) are minimal.

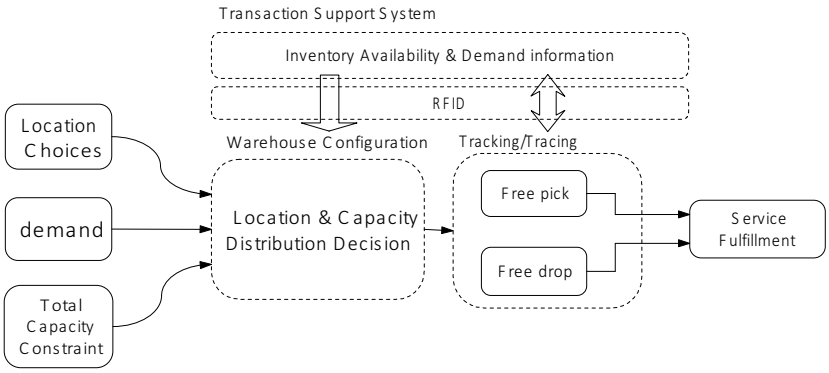

Figure 3: Flexible Warehouse Transaction Support System

In the base case model (Figure 3), we fix all silos in terms of number and location as well as individual capacity constraints. This scenario applies to an operational environment in which change in layout or movement of facility equipment (here, silos) are relatively expensive or infeasible. The number and location of silos are determined by equipment layout availability and other site-specific constraints. In this setting, a set of demand locations and a set of available candidate silo storage locations are given. A fixed location cost is incurred if a candidate location is used to accommodate silos. The shipping cost is assumed to be linear based on the distance between each demand location and candidate site. We optimize this problem given the set of silo locations, their capacities, and the pick&drop availability status. This is similar to the classical facility location problem, but is different in the sense of disregard for the product-inventory-position affiliation. In the classical model, if items are randomly dropped off, the system collapses and routines have to be setup. In our model, items can be freely placed and fetched with minimal additional effort. The routine in terms of where (among the silos) an item belongs, etc. is minimum and so the associated assumptions are different. We focus our research on the investigation of a novel inventory mechanism that enables a fully flexible warehousing configuration. Our intention is to simplify the model as much as possible by focusing on the key elements. With the explosion in demand for low-cost passive RFID tags and recent advances in RFID tag technology, the unit cost of these tags have dropped to the low single-digits. Moreover, the addition of linear RFID tag cost will not change the properties of the results in our model.

We use the following notation:

- I: set of demands, indexed by i
- J: set of candidate silo locations, indexed by j

- K: set of items, indexed by k
- D_{ik}: inbound demand for item $k \in K$ from demand $i \in I$
- A_{jk}: availability of item $k \in K$ at silo $j \in J$
- f_j: fixed cost of locating a facility at silo $j \in J$
- c_{ijk}: marginal distance cost of fetching/returning item $k \in K$ between demand location $i \in I$ and silo $j \in J$. It's strictly increasing with the distance between $i \in I$ and $j \in J$.
- s_{ijk}: search cost for item $k \in K$ at the $i \in I$ demand location, served by the $j \in J$ silo. It is independent of distance.
- R_{ik}: outbound demand from service location i for item k
- sr_{ijk}: the search cost of an available drop-off location i with silo j for an item k
- b_j: inventory capacity at $j \in J$
- A_{jk}: the initial state of availability of the k^{th} item at location j

The decision variables are:

- $X_j = \begin{cases} 1 & \text{if silo is assigned at } j \in J \\ 0 & \text{otherwise} \end{cases}$
- Y_{ijk}: Inbound demand for k^{th} item at location $i \in I$ to be served by silo $j \in J$
- Z_{ijk}: Outbound demand of k^{th} item at location i that is returned to silo j

Considering a single-period scenario with both inbound and outbound demands, the transactional objective is to find the optimal initial configuration of the silo location, item fractions, and the capacity control, as represented by equation (1).

$$Minimize \quad \sum_{j \in J} f_j X_j + \sum_{j \in J} \sum_{i \in I} \sum_{k \in K} (c_{ijk} + s_{ijk}) Y_{ijk} + \sum_{j \in J} \sum_{i \in I} \sum_{k \in K} (c_{ijk} + sr_{ijk}) Z_{ijk} \qquad (1)$$

Subject to:

$$\sum_{j\in J} Y_{ijk} = D_{ik}, \qquad \forall i \in I \tag{2}$$

$$\sum_{j\in J} Z_{ijk} = R_{ik} \qquad \forall i \in I; \forall k \in K \tag{3}$$

$$\sum_{i} Y_{ijk} \leq A_{jk} X_j, \forall j \in J, \forall k \in K \tag{4}$$

$$\sum_{k} (A_{jk} - \sum_{i} Y_{ijk}) + \sum_{k} \sum_{i} Z_{ijk} \leq b_j X_j, , \quad \forall j \in J \tag{5}$$

$$X_j \in \{0, 1\}, \qquad \forall j \in J \tag{6}$$

$$Y_{ijk} \in IN^+, \qquad \forall i \in I; \forall j \in J; \forall k \in K \tag{7}$$

$$Z_{ijk} \in IN^+, \qquad \forall i \in I; \forall j \in J; \forall k \in K \tag{8}$$

The objective function (1) minimizes the sum of the location costs and the shipment costs. Expressions (2) - (5) impose constraints on economic rationality, physical capacity, and inventory availability. Expression (2) states that all demands are fully assigned. Expression (3) is an availability constraint, stating that a demand can only be assigned to an available silo. (4) is integrality and non-negativity constraints. (5) represents the maximum capacity constraint. (6) to (8) specify the integer property of the problem, while (6) is binary. (7) and (8) become binary if we consider k as the item-level identification rather than the categorical parameter.

In what follows, we extend this single stage model by considering the warehouse operation optimization in multiple periods, enabled by a real-time decision support system.

3.2 Multiple-Period Decision Support System

In the multiple-period decision support system, we consider an inventory management problem that utilizes item-level information to support both individual transactions and warehouse configurations. Individual transactions of inventory items include activities such as searching the database to retrieve inventory availability, registering a drop-off transaction, and making recommendations of optimal item availability and storage availability. Warehouse configurations allow the warehouse to reconfigure the location and capacity distribution after each operational process. It indicates a scenario where both the silos and the inventory items in this system are tracked in real-time at the item-level. Storage distribution is flexible and individual silos are allowed to relocate during operations. Items are tracked and traced and are allowed to be placed randomly at any available silo on site. Figure 4 illustrates the process flow of this proposed flexible warehouse decision support system. In addition to the once-for-all system, the transaction support system also provides in real time with information on warehouse configuration. In

multi-period decision support system, we analyze the performance of service fulfillment and its quality along with the original configuration and decisions to discover the patterns among warehouse configuration, inventory availability and demand information. In the following period, the patterns are continuously refined for real-time decision making for each RFID traceable item.

We consider the flow of items in multiple stages where the Markovian state of any item depends on its previous state. A_{jkt} is the state parameter that represents the availability of the k^{th} item at the j^{th} site at the end of the t^{th} period. It evolves as a sequence of states over time, where $A_{jk}|_t$ depends on the previous state $A_{jk}|_{t-1}$. This problem is difficult to solve because it is a multiple-period problem. Without considering the state parameter $A_{jk}|_{t\in T}$ in multiple-period, the problem can be considered as a modified knapsack problem, which can only be solved optimally in pseudo-polynomial time. This revised problem is more complex than the knapsack problem. The state variable makes this problem different from any existing facility location models with the consideration of the flow patterns of individual items on site.

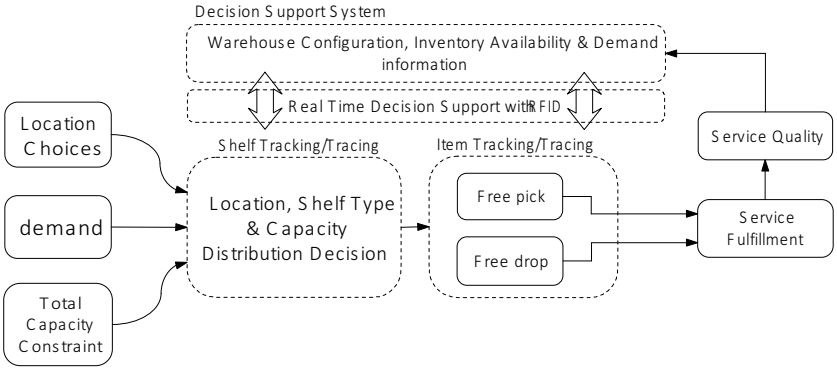

Figure 4: Flexible Warehouse Real-time Decision Support System

To model this, we extend the previous setup by considering the following variables:

- R_{ikt}: outbound demand from service location i for item k at time t
- sr_{ijk}: the search cost of an available drop-off location for an item
- b_j: inventory capacity at $j \in J$
- $t \in T$: the period index
- A_{jkt}: the availability of the k^{th} item at location j at the end of period t
- A_{jk0}: the initial state of inventory

We now have the following revised decision variables:

- $X_{jt} = \begin{cases} 1 & \text{if silo is located at } j \in J \text{ at state } t \\ 0 & \text{otherwise} \end{cases}$
- Y_{ijkt}: Inbound demand of k^th item at location $i \in I$ that is served by silo $j \in J$
- Z_{ijkt}: Outbound demand of k^{th} item at location i that is served by silo j at time t

The problem becomes:

$$Minimize \quad \Upsilon(X_{jt}, Y_{ijkt}, Z_{ijkt}) = \sum_{j \in J} f_j X_{jt} + \sum_{j \in J} \sum_{i \in I} \sum_{k \in K} [(c_{ijk} + s_{ijk}) Y_{ijkt} + (c_{ijk} + sr_{ijk}) Z_{ijkt}] \tag{9}$$

Subject to:

$$A_{jkt} = A_{jk(t-1)} + \sum_i Z_{ijkt} - \sum_i Y_{ijkt} \quad \forall i \in I; \forall j \in J; \forall k \in K \tag{10}$$

$$\sum_k A_{jkt} \leq b_j X_{jt} \quad \forall j \in J; \forall t \in T \tag{11}$$

$$\sum_j Z_{ijkt} = R_{ikt} \quad \forall i \in I; \forall k \in K; \forall t \in T \tag{12}$$

$$\sum_j Y_{ijkt} = D_{ikt} \quad \forall i \in I; \forall k \in K; \forall t \in T \tag{13}$$

$$\sum_i Y_{ijkt} \leq A_{jkt} X_{jt} \quad \forall j \in J; \forall k \in K; \forall t \in T \tag{14}$$

$$X_{jt} \in \{0, 1\}, \quad \forall j \in J \tag{15}$$

$$Y_{ijkt} \in IN^+, \quad \forall i \in I; \forall j \in J; \forall k \in K; \forall t \in T \tag{16}$$

$$Z_{ijkt} \in IN^+, \quad \forall i \in I; \forall j \in J; \forall k \in K; \forall t \in T \tag{17}$$

Equation (10) represents that flow conservation constraint. Expression (11) specifies the capacity constraint $\sum A_{jkt} \leq b_j$ when $X_j = 1$. It also excludes the possibility of having any inventory when $X_j = 0$. Equation (12) specifies that all outbound demands are serviced. Equation (13) specifies that all inbound demands are serviced. $\sum_i \sum_k Z_{ijkt} \leq b_j X_j \quad \forall j \in J$ creates the same boundary as expression (14) to eliminate the possibility to use a zero-sum activity at a non-existing location and to eliminate the possible dead loop in this model.

In order to find the optimal strategy in multiple periods, we consider a dynamic program with control variables that adjust both local inventory capacity and their locations. We assume that local inventory capacities are manageable by periodic global evaluation and relocation of silos on site. Prior state stochastic characteristics are used to facilitate decision making at the

next period. We now extend previous models by dynamic programming that generates global optimal policy over time.

The global optimization is based on expression (18), which is to maximize profit and to minimize the total operational cost δ for the entire discrete time period.

$$\max_{v_t} \{\delta(\Upsilon_0, \Upsilon_1, \cdots, \Upsilon_T; v_0, v_1, \cdots, v_{T-1})\} \tag{18}$$

where

- $\Upsilon_t(X_j, Y_{ij}, Z_{ij}, A_{jk}, b_j)$: the cost function at any time point that equals $\sum_{j\in J} f_j X_{jt} + \sum_{j\in J}\sum_{i\in I}\sum_{k\in K}[(c_{ijk} + s_{ijk})Y_{ijkt} + (c_{ijk} + sr_{ijk})Z_{ijkt}]$
- δ: the production function
- v_i: vector of control variables

In expression (18), δ is the production function that considers both cost and revenue variables to account for the total operational profit. It is a linearly decreasing function of the operational cost at t, Υ. If we assume that all demands can be satisfactorily serviced, to maximize profit is the same as to minimize the total cost Υ. v_t is a vector of control variables that take place at the beginning of each period, such that $v_t = \psi_t\{\Upsilon_t\}$, where ψ_t includes the adjustment function based on the current performance at each stage. We consider the accumulated costs over a finite discrete time period $\{t_0, t_1, \cdots, t_{T-1}, t_T\}$. Then, $\delta(\Upsilon_0, \Upsilon_1, \cdots, \Upsilon_T; v_0, v_1, \cdots, v_{T-1}) = \delta_0(\Upsilon_0, v_0) + \delta_1(\Upsilon_1, v_1) + \cdots + \delta_{T-1}(\Upsilon_{T-1}, v_{T-1}) + S(\Upsilon_T)$ where $S(\Upsilon_T)$ is a remaining value function at the end of finite time period. We define $\xi(\cdot)$ as an inter-temporal function that connects the state and control variables such that $\Upsilon_T = \xi_{T-1}(\Upsilon_{T-1}, v_{T-1})$. Using Bellman's method, the original optimization problem presented in expression (18) is the same as the following recursive function:

$$V(\Upsilon_{T-k}, k) \tag{19}$$

$$= \max_{v_{T-k}} \{\delta_{T-k}(\Upsilon_{T-k}, v_{T-k}) + V(\Upsilon_{T-k-1}, k+1)\} \tag{20}$$

$$\equiv \max_{\psi_{T-k}} \{\delta_{T-k}(\Upsilon_{T-k}, \psi_{T-k}(\Upsilon_{T-k})) + V(\xi_{T-k}(\Upsilon_{T-k}, \psi_{T-k}(\Upsilon_{T-k})), k+1)\} \tag{21}$$

subject to:

$$\Upsilon_{T-k+1} = \xi_{T-k}(\Upsilon_{T-k}, v_{T-k}, g(v_{T-k})) \tag{22}$$

$$\Upsilon_0 = \tilde{\Upsilon_0} \tag{23}$$

$$v_{T-k} = \psi_{T-k}(\Upsilon_{T-k}) \tag{24}$$

$$v_t \in \Theta \text{ for all } t = 0, 1, \cdots, T-1 \tag{25}$$

Constraint (22) represents the state transition from the previous period; constraint (23) represents the original state at time 0; constraint (24) determines the control function at each period; in constraint (25), Θ is the feasible set for the control variables that is assumed to be closed and bounded. Because Υ_0 is a given value at the origin of the overall dynamic programming problem, we are able to solve for v_0 as a number that is independent of the Υs. It is easy to compute Υ_1, and hence v_1, from the control rule of that period, and then Υ_2, v_2, etc. This process can be repeated until all the Υ_i and v_i values are known. Hence we have the set of optimal control function and its production or cost outcome.

By controlling the position of each silo unit at time t, we are now able to optimize the finite time performance in our problem setting. In a supervised environment, we are interested in the control of both inventory silo positions and local capacities. With varying demand patterns, we expect to control these two parameters to optimize the overall performance. We let m_{ij} represent the operational cost of moving a silo from location i to location j. V_{ij} is a $\{0, 1\}$ decision variable, where 1 represents the movement from location i to j. The cost of dynamic capacity control is expressed in the control vector v_t for supervised Markovian optimization. In this model, we incorporate:

- b_{jt}: the maximum demand that can be assigned to a location $j \in J$ at time $t \in T$.
- m_{ij}: cost to relocate a silo from location i to j.
- V_{ijl}: movement of the l^{th} silo from i^{th} to the j^{th} location, where $l \in 1, 2, \cdots, L$ represents a silo in the silo set L.

At any time point t, we have the following optimization problem:

$$Minimize \quad \sum_{t=1}^{T} \{\Upsilon_t(X_{jt}, Y_{ijkt}, Z_{ijkt}, V_{ijl})\} \tag{26}$$

$$= \sum_{t=1}^{T} \left\{ \sum_{j \in J} f_j X_{jt} + \sum_{j \in J} \sum_{i \in I} \sum_{k \in K} \left[(c_{ijk} + s_{ijk}) Y_{ijkt} + (c_{ijk} + sr_{ijk}) Z_{ijkt} + \sum_{l \in L} V_{ijl} \cdot m_{ij} \right] \right\} \tag{27}$$

Subject to:

$$\sum_k A_{jkt} = A_{jk(t-1)} + \sum_i Z_{ijkt} - \sum_i Y_{ijkt} \quad \forall i \in I; \forall j \in J; \forall k \in K \tag{28}$$

$$\sum_k A_{jkt} \leq b_{jt} X_{jt} \quad \forall j \in J; \forall t \in T \tag{29}$$

$$\sum_j Z_{ijkt} = R_{ikt} \quad \forall i \in I; \forall k \in K; \forall t \in T \tag{30}$$

$$\sum_j Y_{ijkt} = D_{ikt} \quad \forall i \in I; \forall k \in K; \forall t \in T \tag{31}$$

$$\sum_i \sum_k Y_{ijkt} \leq b_{jt} X_j \quad \forall j \in J; \forall t \in T \tag{32}$$

$$b_{jt} = b_{j,t-1} + \sum_{l \in L} \left(\sum_i V_{ijl} - \sum_i V_{jil} \right) \quad \forall l \in L; \forall t \in T \tag{33}$$

$$X_{jt} \in \{0, 1\}, \quad \forall j \in J \tag{34}$$

$$Y_{ijkt} \in IN^+, \quad \forall i \in I; \forall j \in J; \forall k \in K; \forall t \in T \tag{35}$$

$$Z_{ijkt} \in IN^+, \quad \forall i \in I; \forall j \in J; \forall k \in K; \forall t \in T \tag{36}$$

$$V_{ijl} \in \{0, 1\}, \quad \forall l \in L \tag{37}$$

where the control function V_{ijl} is dependent on the $(t-1)^{th}$ state (33). Therefore, the local capacities are adjustable according to the total number of storage silos at a certain location.

3.3 RFID-Enabled Real-time Knowledge-Based System: Practical Example

A knowledge-based system is a natural fit to implement the flexible warehousing concept due to its capability for real-time data collection, processing, and decision making at a fine level of granularity. Instead of attempting to find a closed-form solution, we propose the practical use of a knowledge-based self-adjusting mechanism (Figure 5) for the dynamic silo location/capacity problem and to satisfy all inventory demands at the lowest cost over multiple periods. The core of this knowledge-based system is the knowledge base that is kept updated and current through machine learning. The learning algorithm used is chosen based on system-specific parameters (e.g., data type, volume). Knowledge in the knowledge base is evaluated based on system performance, which is used to identify deficit (e.g., situation where poor generalizability occurs when a heretofore unseen pattern is observed in the system). Such deficits are addressed through learning. As required or necessary, the system continually or continuously monitors for changes in demand patterns and triggers a new iteration when a significant pattern change is observed. At the start of this new iteration, operators relocate inventory silos to adjust warehouse location and capacity according to demand change. Demands then are satisfied in a revised warehousing

setup. Real-time data on local inventory, demand fulfillment, local capacities, and inventory flow are continuously measured, evaluated, and learned to create the knowledge-base that is used to facilitate inventory decision-making in the next iteration.

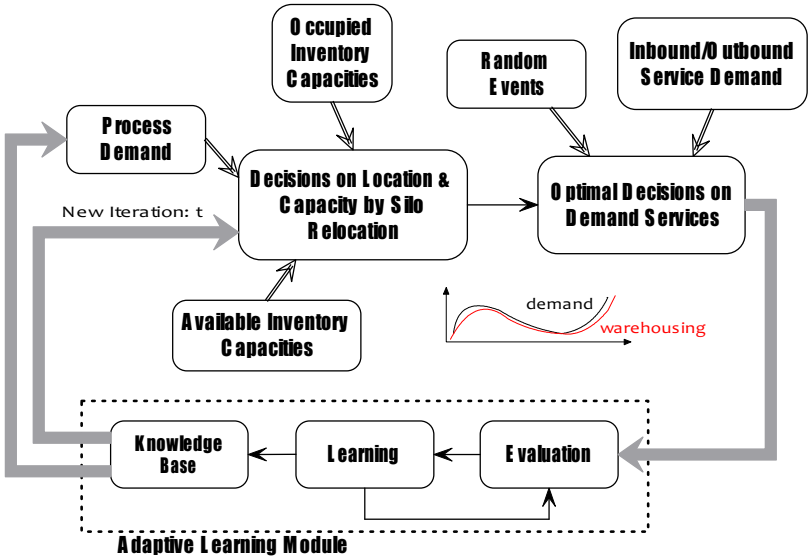

Figure 5: Knowledge-Based Warehousing System

3.3.1 Warehouse Pattern Recognition: An Example on Capacity Distribution

Patterns in a warehouse are formed from routine business processes. The discovery of and decision-making based on these patterns are largely subject to the physical constraints (capacity, location, handling rules, etc.) and the priority of these constraints. Learning and decision making with respect to the local capacities b_{jt}, for example, are controlled by a series of silo movements, depending on the previous state $\{v_{T-k}\} = \psi_{T-k}(\Upsilon_{T-k})$. Our objective is to relocate silos at the end of each state (or the beginning of the next state) according to the current state inventory demand dynamics. In the most straightforward scenario, we can use the K-nearest neighbor (KNN) algorithm to discover the demand patterns and the best location/capacity decision. A simple mathematical alternative of this problem is to minimize the overall imbalance of neighborhood capacity considering that we have more inventory capacity than what is needed to fulfill all demands. This problem is the same as the one to find the least square of excessive inventory capacity in all the found neighborhoods:

$$minimize \sum_{j \in J} \left(b_{j,t+1} - \sum_{k \in K} A_{jkt} \right)^2 \tag{38}$$

Subject to:

$$\sum_{j \in J} b_{j,t+1} = B \tag{39}$$

where B is the the total inventory capacity of the warehouse. After the optimal capacities are allocated, demands can be serviced by the k nearest neighbors if we incorporate more constraining perspectives, such as tool comparability, etc. KNN-SVM (KNN-Support Vector Machines) can be used to handle more complex problems that involve not only temporal-spatial dimensions but also task-specialized classifications. With our ultimate goal to create a completely flexible warehousing mechanism that not only considers the basic demand, capacity, location, and cost constraints but also constraints embedded in the characteristics of tools, inventory items, warehousing conditions, process requirements, etc., we illustrate in the next section a self-learning mechanism that automatically enables the system to create new rules and discover patterns from prior activities.

Knowledge-based system is essential in the flexible warehousing setup. The intertwined complexity of today's and the immediate near future's inventory management system demands a deep involvement of ICT tracking, data analytics, and decision support. A direction of future research could be an in-depth investigation of a knowledge-based system along with its associated data mining techniques and optimization algorithms. From a transaction support system, to the decision support system, and eventually the example of knowledge based system, we develop this study based on the degree of site complexity/requirement and depth of data analytics in a flow that we consider to be logical and coherent. We do not discuss in detail the learning part since it is context-specific and any appropriate learning algorithm would serve this purpose. Additional effort to enrich the warehouse knowledge-based system is needed in search for the most appropriate/optimal/effective methods and algorithms for different industry and business setups.

3.3.2 A Note on Item-Level Information

RFID-generated information enables the operator to monitor the real-time operational flow at the item level on all tagged inventory items and storage silos. After an initial investment, the operational and maintenance cost of such a system is very low. In the dynamic programming, we made an assumption on time t by fixing the period length. This rigid assumption in reality is not very practical especially when demand fluctuates at a changing frequency. We relax this assumption by changing time period to algorithmical iterations based on real-time data.

The above-described problem in the conventional setup is difficult to implement with existing technology without the use of automated monitoring and a learning mechanism.

Information at the item level [40] appears to be very important in our proposed flexible warehouse setup. If we consider batch operation conditions such as those in mass production, the flexible system does not seem to offer a better (if not worse) operational performance than the traditional rigid system. If the operation requires batch-fetching in a large scale, a rigid system still benefits from the economic of operating on large quantity. Because of this reason, it limits the usage and application of flexible warehouse only to "item level" operations, which fits very well with modern operation concepts of just-in-time and mass-customization.

One distinctive characteristic of mass customization is component variations such that each item is individually ordered and processed even for the same part component in a final product. In a traditional rigid warehouse, part items for mass production are grouped together without further differentiation and are able to be delivered in batches. It has become an emerging logistic problem when components are customized, making it impossible to batch process the demands without first sequencing the inventory. In a flexible system, this problem diminishes thanks to direct item-level visibility.

4 Numerical Analysis

We now discuss numerical results based on various scenarios that are considered to be impactive in actual management of such a system. We conduct numerical experiments that enable us to obtain important managerial insights that are otherwise difficult to develop analytically. In particular, we attempt to answer the following questions:

- How beneficial is the flexible warehousing mechanism in production logistics? With what spectrum of variables does it achieve the largest value and lowest cost?
- Is dynamic scenario necessary, or can we capture most of the potential benefits by using a static silo arrangement strategy whose location and capacity are determined optimally?
- How important is the silo distribution network topology? Does dynamic control generate costs that differ considerably for specific silo distribution configurations?

We applied our proposed mechanism and policies to a total of 3000 cases. In each experiment, the dynamics of optimal policy variables were captured following the models presented in Section 3. We use a uniform distribution of demand in the simulation experiments. We use unity

iteration to represent a standard operational time period. In Section 4.1, we consider a basic system with 10 fixed position silos with fixed capacity constraints, using the policy proposed in the previous section and identify environments where flexible warehousing is most beneficial. We use this setup to illustrate how decision support can be achieved for this problem scenario. The goal here is not to determine the initial silo location for optimal system performance. In Section 4.2, we investigate the effect of the capacity constraints of the silos in the network. In Section 4.3, we explore the environments where a dynamic control and flexible warehousing topology works the best.

4.1 Non-relocatable Storage & Flexible Item-Operation

In the experimental setup listed in Figure 6, we design a basic operational environment where all core operational tasks are confined to a one unit by one unit space. There is a warehouse located in a nearby spot that is 1.5 by 2 units away from the origin, as illustrated in Figure 7. The spatial unit, which can be 100 meters in reality for example, is exemplified in order to concentrate on the problem itself. Within the one-by-one operational space, a demand can be initiated at any spot. In the first experiment, we randomly place 10 silo location candidates to serve demands that are randomly initiated. All containing silos (shelves) are capable of accurately reporting the number of each item and its associated item-level identifications. These silos are very different from the traditional line-side buffer inventory because they accept random pick&drop activities of RFID-tagged items at any time within availability, while a traditional line-side buffer is designed to accommodate planned inventory management, such as the scheduling of work-in-progress (WIP) and feeding raw materials.

RFID-tagged items can be stored either in the warehouse or in any of the silos in the operational site. In order to serve the needs, which can be either a retrieval or a return request, operators may choose to use either storage choice - the warehouse or any of the available silos. When a return request is initiated, the least expensive option for the operator is to drop the item at the nearest storage location with available space. In order to fetch an item, either direct line-of-sight or a hand-held device can be used to locate the nearest silo that contains the item of interest. If the nearest silo is not available for service, operators can attempt to locate another silo that is a little farther away or directly return the item to the distribution warehouse. Search cost is incurred if a silo can not be located through direct line-of-sight.

Compared to the operational performance with a single distribution center, the results presented in Figure 7 show the cost (reduction) in percentage with consideration of the total number of iterations and demands. Results show that the reduction in overall trip cost increases with an

	X	Y
1	0.8406	0.0975
2	0.6058	0.9847
3	0.0800	0.6203
4	0.8651	0.6269
5	0.5936	0.0604
6	0.0233	0.6445
7	0.4687	0.8812
8	0.5153	0.7953
9	0.3521	0.9252
10	0.3657	0.2501
11	1.5000	2.0000

Figure 6: Coordinates of 10 silo locations and a remote distribution center

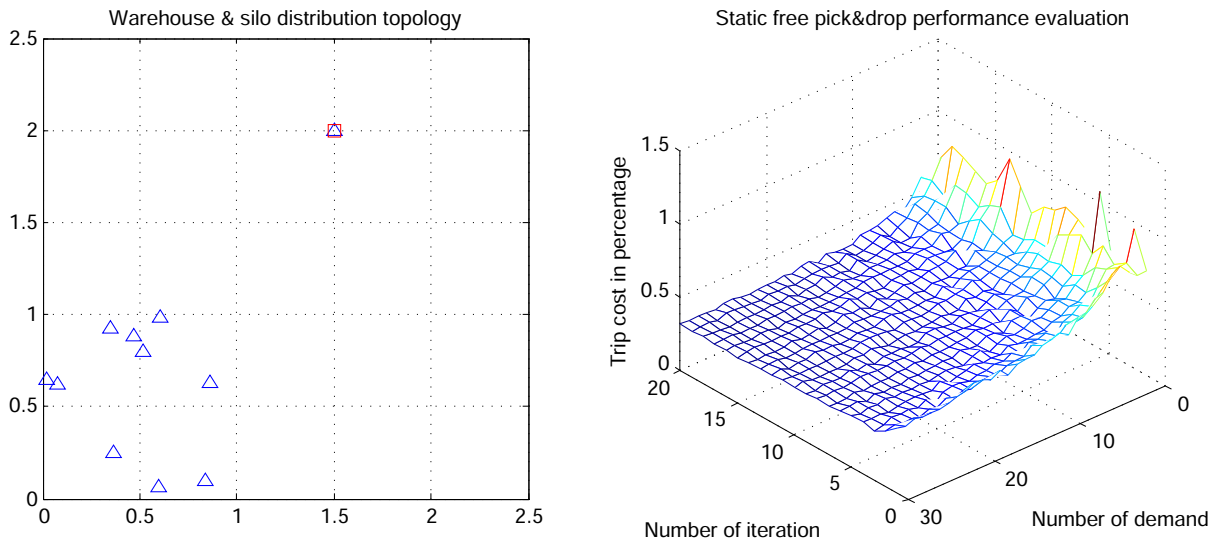

Figure 7: Number of iteration, number of demand & cost impact

increasing demand. Cost reduction also increases with the number of iterations but at a slower rate. It also shows that the trip cost reduction gradually stabilizes at a certain level, which is 32.16% in this specific example.

Table 1 summarizes the percentage of trip cost that enables flexible pick and drop at the nearest available location in percentage compared to the scenario with one distribution center. The performance gradually improves with higher demand density because of the availability of items that flow in nearby locations. The cost performance also improves with increasing number

Table 1: Trip cost reduction of flexible warehousing compared to the rigid warehousing case

	Iteration									
Demand	2	4	6	8	10	12	14	16	18	20
2	72.49%	79.43%	73.09%	72.61%	74.30%	56.49%	64.49%	69.32%	60.48%	53.56%
4	66.68%	61.95%	58.65%	55.13%	49.35%	50.96%	47.07%	45.95%	46.10%	43.66%
6	57.97%	56.57%	49.53%	47.03%	49.91%	42.05%	40.16%	40.18%	38.71%	42.51%
8	54.57%	45.17%	41.56%	42.66%	40.25%	38.92%	39.03%	37.00%	38.37%	39.02%
10	49.03%	44.54%	45.27%	38.35%	39.67%	39.40%	37.53%	38.88%	38.37%	36.28%
12	46.00%	40.41%	39.75%	39.44%	39.17%	37.22%	39.01%	37.48%	36.50%	35.57%
14	44.87%	39.17%	39.92%	37.59%	37.05%	34.87%	36.05%	36.94%	34.38%	34.30%
16	44.58%	40.54%	35.98%	36.85%	36.73%	34.19%	34.84%	35.11%	34.53%	35.00%
18	44.12%	37.06%	37.24%	35.06%	34.26%	35.63%	33.74%	33.77%	34.53%	32.59%
20	41.34%	36.94%	35.61%	35.14%	35.13%	35.11%	33.71%	32.32%	34.00%	33.24%
22	41.11%	36.22%	35.55%	34.94%	34.34%	33.65%	34.84%	32.69%	31.12%	32.21%
24	41.60%	35.95%	35.93%	36.09%	33.00%	33.19%	32.24%	32.26%	32.53%	32.68%
26	37.53%	36.72%	34.70%	33.07%	33.49%	31.81%	33.39%	32.65%	32.58%	31.90%
28	39.83%	34.74%	35.35%	32.97%	33.75%	32.34%	31.78%	32.41%	31.08%	32.79%
30	36.15%	34.75%	33.26%	33.17%	33.27%	33.61%	32.14%	32.64%	31.69%	30.97%

of iterations because of the clustering effect that moves the same items to the neighborhood with higher demand. This result is also plotted in Figure 7, from which we find that the improvement converges to a lower bound. This lower bound specifies the maximum performance given the demands and available inventory location even without capacity constraint.

Figure 8: Randomly Generated Facility Layouts For Benchmark Testing

We now consider benchmarking and performance comparison between the flexible system and the rigid system. With a focus on the flexible system, we test various capacity, location, and demand settings. We simplify the rigid system by assuming one fixed location where all the items should be retrieved and returned. This setup can be generalized to a more complex rigid system with multiple locations. As long as the capacity, location, and association rules are fixed, the statistical performance of the simplified and expanded rigid systems display the same operational properties. We extend the above simulation (Figure 7) to four different flexible location setups, all benchmarked by the same one rigid system. At a practical level, it would be easy to add a set of rigid systems, but much more complicated to present the results. And,

extending the simulation on the side of the rigid system will not generate more meaningful results, so we focus on the variations of the flexible system. Figure 8 shows the geographical distribution of four warehouse setups with 5, 10, 15 and 20 randomly generated locations.

Figure 9 reports the performance benchmarks on these four setups with location choices. We find that simply based on location choices, the performance demonstrates similar patterns with respect to the number of demand and number of iterations, although slight systematic differences exist. One of the reasons is that in these tests we have not considered the capacity constraints. In what follows, we extend these tests by incorporating both global and local capacity constraints and examine their impacts on the proposed flexible warehouse system.

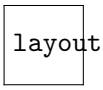

Figure 9: Static Performances of Randomly Generated Facility Layouts

4.2 Effects of Various Capacity Constraints

So far, we have generalized the problem without considering the total capacity constraint or the individual location capacity constraints. With a realistic capacity constraint, the optimization problem is different because of the added search cost when an item or inventory is not immediately available at the closest convenient location.

For the new experiments, we revise the setups from Section 4.1 by adding local capacity constraints, with the assumption that the remote distribution center has infinite storage capacity. The revised experiment is repeated with three different warehousing configurations that include 10, 15 and 20 silo location choices respectively. In each experiment, the individual location capacity takes on values in the zero to fifty range.

Figure 10 summarizes the results with pre-configured capacity constraint. We assume that the setup cost is linear to the inventory capacity. It shows that large inventory capacity helps the flexible warehouse to reduce its trip cost and search cost. With limited inventory capacity, our results show that the performance of our proposed flexible warehouse is close to the traditional setup. More location choices also help reduce the operational costs. It seems that the marginal benefit of introducing more flexibility (more location choices and increased inventory capacity) differs with different levels of demand dynamics. The results also show a lower bound of cost performance with capacity constraints. The trip cost and search cost decrease with additional flexibility but the improvement margin becomes narrower.

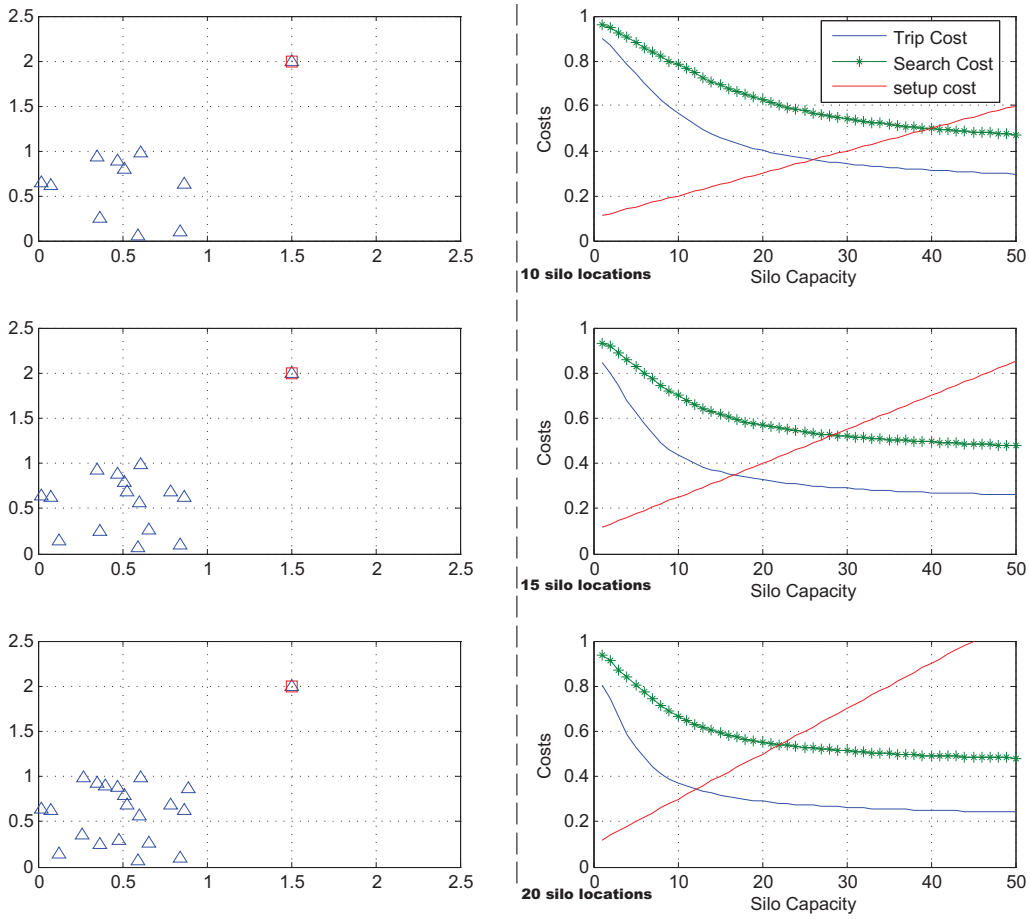

Figure 10: Trip cost & search cost with various capacity constraints

4.3 Dynamic Control

In the previous section, we investigated the performance when a set of fixed capacity is imposed on the system. We now study the impact of dynamic control that allows the system to adjust the local capacities according to the demand dynamics and the patterns discovered on site.

Table 2 summarizes the results that compare the performance of fixed capacity constraint with flexible capacity in the warehouse system for both fixed capacity and dynamic capacity scenarios. We find that dynamic control performs extremely well with limited inventory capacity with more than 50% of cost reduction compared to the fixed capacity scenario. When capacity increases, the dynamic control still performs better but the marginal benefit decreases. When the overall inventory capacity exceeds the overall demand, we also find that the dynamic control performs very close to the static scenario, as shown (22.85%) compared to 24.22% in the case

Table 2: Trip cost performance comparisons on number of location, average local capacity, static and dynamic control

	Average Local Capacity									
Location number										
Static	5	10	15	20	25	30	35	40	45	50
10	72.98%	55.76%	45.35%	39.74%	36.29%	33.98%	32.35%	31.11%	30.21%	29.44%
15	62.37%	43.85%	36.25%	32.54%	30.32%	28.86%	27.82%	27.00%	26.41%	25.93%
20	52.34%	36.81%	31.39%	28.79%	27.28%	26.25%	25.54%	25.00%	24.55%	24.22%
Dynamic										
10	36.17%	35.46%	35.15%	35.08%	35.02%	34.91%	34.82%	34.79%	34.80%	34.78%
15	35.47%	34.42%	34.21%	34.06%	34.01%	33.90%	33.91%	33.87%	33.91%	33.91%
20	23.77%	23.40%	23.18%	23.05%	22.99%	22.95%	22.94%	22.91%	22.87%	22.85%

of 20 location choices and 50 average local capacity.

These results are also illustrated in Figure 11. It shows drastic performance improvement under dynamic control that allow silos to be relocated according to the actual demands on site. For the same demand pattern, dynamically controlled inventory system seems to be able to lower the lower bound of trip cost. This is intuitively explainable because when the warehouse can adjust itself to accommodate on-site demand change, it reduces the overall trip cost.

5 Concluding Remarks

We introduce a novel concept of flexible warehousing based on a real-time decision support system, enabled by RFID-generated data. This smart mechanism enables completely flexible warehousing shelf locations, local capacities, and transportation routes. With a periodically renewable fixed global warehousing capacity, we discard the location constraint, local capacity constraint, and demand assignments routines that typically exist in facility location problems. Dynamic decisions on location and local capacity are made based on the stochastic Markovian demand states. In this scenario, warehousing shelves and equipment are modified according to demand flow over time. All inventory items and warehousing equipment are trackable at a fixed setup cost and minimal operational cost so that all items and equipment are visible in real-time. In the second stage, we optimize the process and routine constraints by allowing free pick-up and drop-off mechanism in which inventory items at hand are dropped off at the closest available location and items are picked up at the closest location. We compared the performance of the considered flexible warehousing mechanism with classical models, through various degrees of constraint relaxations. Our results show significant performance improvement and demonstrate

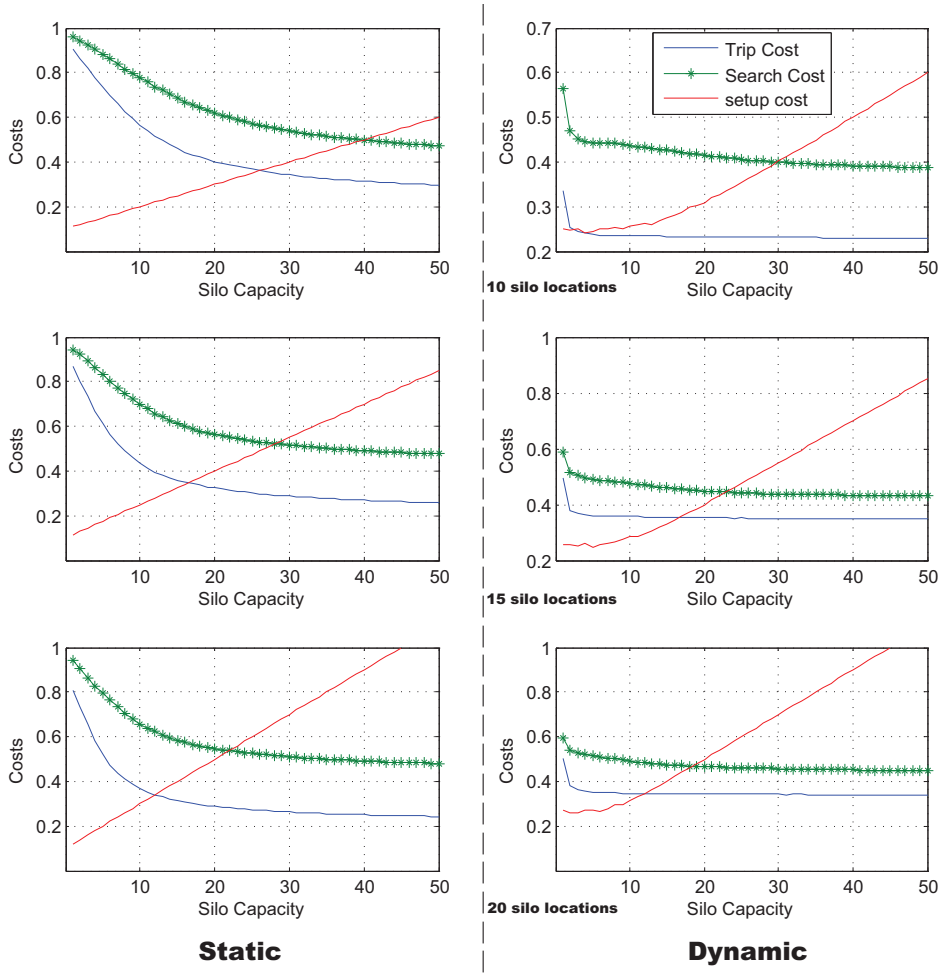

Figure 11: Comparison of static and dynamic controls

that (1) "free pick-n-drop" combined with fluid warehousing mechanism greatly reduces trip costs and lead time for single trip demand, (2) there exist a lower performance bound in such a setup with fixed local capacities, and (3) the lower bound can be further improved when inventory capacity and location are adjusted according to actual demand patterns.

Overall, the flexible warehousing mechanism considered in this study represents a type of "smart" logistics system. We expect to see more research in this general field of intelligent logistics that are able to adopt more effective heuristics and more efficient algorithms to approach optimization problems. Although the proposed inventory system is flexible as compared to existing ones, this study is limited because we simplify the configurations of a warehousing environment on selected parameters of location and capacity. With the existence of modern sensor network and big data capability, data collected from various operational parameters can be further utilized, including vibration, temperature, humidity, light, noise, human factor, etc.

These additional parameters create a big data environment that the knowledge-based system can analyze and contribute to optimized warehouse management. For future research, it would be also interesting to investigate the many classical problems in facility location and transportation decisions in the context of completely flexible warehousing, such as the one with multiple batch-fetching demands. Future research can also be directed to find the impact of imperfect tracking information, such as read error, shrinkage, and theft, on the smart inventory system.

Acknowledgement

The work of Chengbin Chu is partially supported by the Natural Science Foundation of China under the Grant #71428002.

References

[1] Albareda-Sambola, M., Fernandez, E., Hinojosa, Y., Puerto, J. (2009). The multi-period incremental service facility location problem. Journal of Computer and Operations Research, 36(5)1356-1375.

[2] Atasoy, B., Gullu, R., & Tan, T. (2012). Optimal inventory policies with non-stationary supply disruptions and advance supply information. Decision Support Systems, 53(2), 269-281.

[3] Balakrishnan, J., Cheng, C.H. (1998). Dynamic layout algorithms: A state-of-the-art survey. Omega, 26(4), 507-521.

[4] Basu, P., & Nair, S. K. (2014). A decision support system for meanvariance analysis in multi-period inventory control. Decision Support Systems, 57, 285-295.

[5] Benchimol, M., Benchimol, P., Chappert, N., Taille, A.d.l., Laroche, F., Meunier, F., Robinet, L. (2011). Balancing the stations of a self service "Bike Hire" system. RAIRO-Operations Research, 45, 37-61.

[6] Bose, I., Lui, A.K.H., and Ngai,E.W.T. (2011). The Impact of RFID Adoption on the Market Value of Firms: An Empirical Analysis, Journal of Organizational Computing and Electronic Commerce, Vol. 21, No. 4, pp. 268-294

[7] Bose I., and Leung, A. C. M. (2008) Radio Frequency Identification for Customer Relationship Management, RFID in Operations and Supply Chain Management: Research and Applications, eds. T. Blecker and G. Huang, Erich Schmidt Verlag Publishing, pp. 273-288.

[8] CAkiCi, O. E., Groenevelt, H., & Seidmann, A. (2011). Using RFID for the management of pharmaceutical inventory-system optimization and shrinkage control. Decision Support Systems, 51(4), 842-852.

[9] Charikar, M., Chekuri, C., Feder, T., Motwani, R. (2004). Incremental clustering and dynamic information retrieval. SIAM Journal of Computing, 33(6), 1417-1440.

[10] Chow, H.K., Choy, K.L., Lee, W.B., Chan, F.T. (2007). Integration of web-based and RFID technology in visualizing logistics operations - a case study. Supply Chain Management: An International Journal, 12(3), 221-234.

[11] Current, J., Ratick, S., ReVelle, C. (1998). Dynamic facility location when the total number of facilities is uncertain: A decision analysis approach. European Journal of Operational Research, 110(3), 597-609.

[12] Dasgupta, S., Long, P. (2005). Performance guarantees for hierarchical clustering. Journal of Computer and System Sciences, 70, 555-569.

[13] DiMaio, P. (2009). Bike-sharing: History, Impacts, Models of Provision, and Future. Journal of Public Transportation, 12(4), 41-56.

[14] Farahani, R.Z., Abedian, M., Sharahi, S. (2009). Dynamic facility location problem. in Facility Location: Contributions to Management Science, 347-372.

[15] Fotakis, D. (2004). Incremental algorithms for facility location and k-median, Lecture Notes in Computer Science (LNCS), 3221, 347-358.

[16] Friggstad, Z., Salavatipour, M.R. (2011). Minimizing movement in mobile facility location problems. ACM Transactions on Algorithms, 7(3), 28.

[17] Gonzalez, T. (2005). Clustering to minimize the maximum inter-cluster distance. Theoretical Computer Science, 38, 293-306.

[18] Hanebeck, C., Lunani, M. (2008). RFID-enabled Returnable Container Management: Solution to a Chronic and Wasteful Automotive Industry Problem. A white paper from IBM Global Business Services web available through ftp://ftp. software. ibm. com/common/ssi/sa/wh/n/gbw030 46usen/GBW03046USEN.

[19] Hasle, P., Bojesen, A., Jensen, P.L., Bramming, P. (2012). Lean and the working environment: a review of the literature, International Journal of Operations & Production Management, 32(7), 829-849.

[20] Iravani, S. M., Kolfal, B., Van Oyen, M. P. (2014). Process flexibility and inventory flexibility via product substitution. Flexible Services and Manufacturing Journal, 26(3), 320-343.

[21] Kim, H.S., Sohn, S.Y. (2009). Cost of ownership model for the RFID logistics system applicable to u-city. European Journal of Operational Research, 194(2), 406-417.

[22] Lin, G., Nagarajan, C., Rajaraman, R., Williamson, D.P. (2010). A general approach for incremental approximation and hierarchical clustering. SIAM Journal of Computing, 39, 3633-3669.

[23] Martin, A., Darpy, D. (2015). L'importance du projet de design dans l'appropriation des objets partags: le cas d'Autolib. In AFM, May.

[24] McCutchen, R.M., Khuller, S. (2008). Streaming algorithms for k-center clustering with outliers and with anonymity. Proceedings of the Workshop on Approximation Algorithms(APPROX), LNCS 5171, 165-178.

[25] Meiller, Y., S. Bureau, W. Zhou, and S. Piramuthu, (2011) "Adaptive Knowledge-based System for Health Care Applications with RFID-generated Information, Decision Support Systems, 51(1), pp. 198-207.

[26] Mettu, R.R., Plaxton, C.G. (2003). The online median problem. SIAM Journal on Computing, 32, 816-832.

[27] Nair, R., Miller-Hooks, E., Hampshire, R.C., Busic, A. (2013). Large-Scale Vehicle Sharing Systems: Analysis of Vlib'. International Journal of Sustainable Transportation, 7(1), 85-106.

[28] Plaxton, C.G. (2006). Approximation algorithm for hierarchical location problems. Journal of Computer & System Sciences, 70, 425-443.

[29] Poon, T.C., Choy, K.L., Chow, H.K., Lau, H.C., Chan, F.T., Ho, K.C. (2009). A RFID case-based logistics resource management system for managing order-picking operations in warehouses. Expert Systems with Applications, 36(4), 8277-8301.

[30] Rosenblatt, M.J., Roll, Y., Zyser, D.V. (1993). A combined optimization and simulation approach for designing automated storage/retrieval systems. IIE Transactions, 25(1), 40-50.

[31] Sana, S. S. (2011). A production-inventory model of imperfect quality products in a three-layer supply chain. Decision Support Systems, 50(2), 539-547.

[32] Shang, J., Tadikamalla, P. R., Kirsch, L. J., & Brown, L. (2008). A decision support system for managing inventory at GlaxoSmithKline. Decision Support Systems, 46(1), 1-13.

[33] Simon, H.A. (1959). Theories of decision-making in economics and behavioral science. The American Economic Review. 49(3), 253-283.

[34] Snyder, L.V. (2006). Facility location under uncertainty: A review. IIE Transactions, 38(7), 547-564.

[35] Thiétart, R.A., Forgues, B. (1995). Chaos theory and organization. Organization Science, 6(1), 19-31.

[36] Tironi, M. (2015). (De) politicising and Ecologising Bicycles: The history of the Parisian Vlib'system and its controversies. Journal of Cultural Economy, 8(2), 166-183.

[37] Tu, Y.-J., W. Zhou, S. Piramuthu, (2009). Identifying RFID-Embedded Objects in Pervasive Healthcare Applications. Decision Support Systems, 46(2), pp. 586-593.

[38] Viani, F., Salucci, M., Robol, F., Oliveri, G., Massa, A. (2012). Design of a UHF RFID/GPS fractal antenna for logistics management. Journal of Electromagnetic Waves and Applications, 26(4), 480-492.

[39] Zhou, W., Tu, Y. J., Piramuthu, S. (2009). RFID-enabled item-level retail pricing. Decision Support Systems, 48(1), 169-179.

[40] Zhou, W. (2009). RFID and item-level information visibility. European Journal of Operational Research, 198(1), 252-258.

Highlights

- The proposed system tracks both inventory items and mobile warehouse equipment at the item level.
- A flexible warehouse scenario where items are dropped and picked as per convenience.
- Warehouse configuration are flexible by relaxing both location constraint and local (e.g., item type-level) capacity constraints with a periodically renewable fixed global capacity.
- The proposed flexible storage system performs better than static systems.

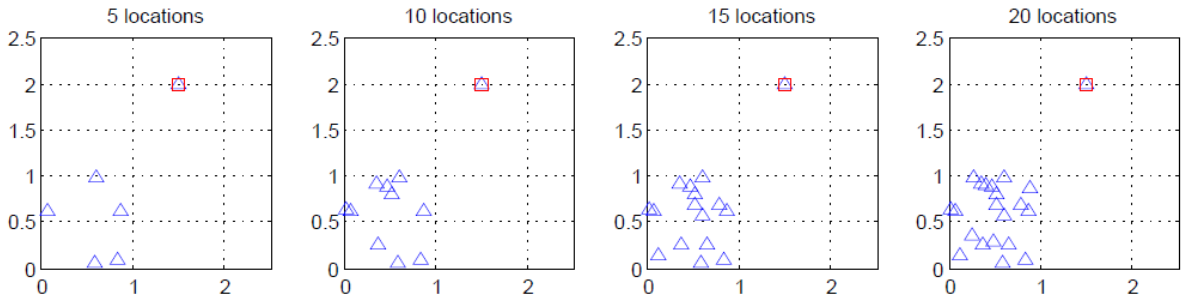

Figure 8

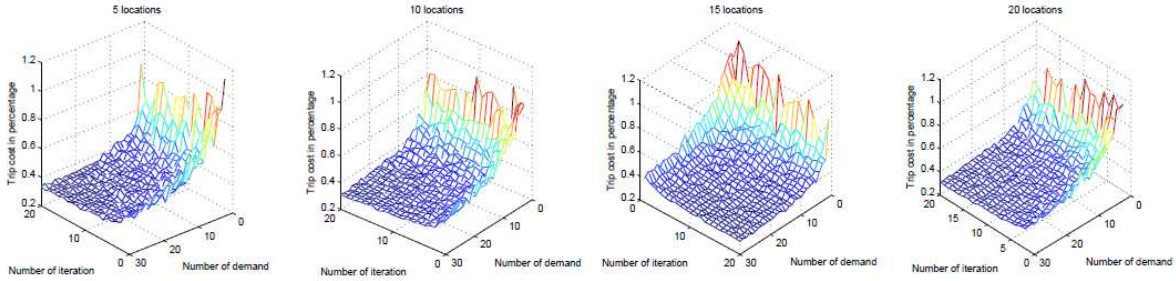

Figure 9