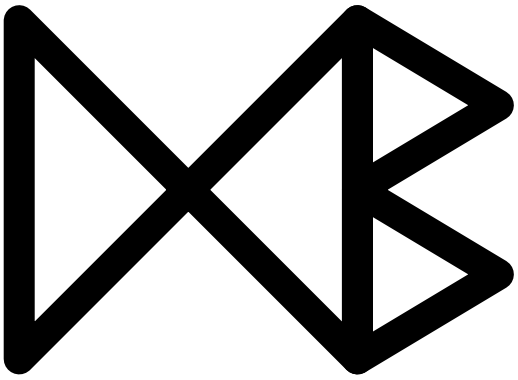


Unified Modeling and Reasoning in Outdoor and Indoor Spaces

Sari Haj Hussein, Hua Lu, and Torben Bach Pedersen

March, 2013

TR-30

A DB Technical Report

Title	Unified Modeling and Reasoning in Outdoor and Indoor Spaces
Author(s)	Sari Haj Hussein, Hua Lu, and Torben Bach Pedersen
Publication History	A DB Technical Report. Version 1, July 2012. Version 2, March 2013. Version 2 enhances over version 1 in the following: 1) It extends the model in version 1. 2) It offers a better motivation for the route observability concept and the static model-based reasoning. 3) It characterizes the relation between a route observability and the uncertainty in tracking moving objects in OI-spaces. 4) It enhances the probabilistic incorporation of RFID data through inferring the information gaps. 5) It concretizes and answers a new query type, namely the dynamic BP monitoring query. 6) It offers a functional analysis that illustrates the behavior of the route observability function. 7) It experimentally analyzes the quality of the inference carried out in order to infer the information gaps.

The DB TECH REPORTS icon is made from two letters in an early version of the Rune alphabet, which was used by the Vikings, among others. Runes have angular shapes and lack horizontal lines because the primary storage medium was wood, although they may also be found on jewelry, tools, and weapons. Runes were perceived as having magic, hidden powers. The first letter in the logo is "Dagaz," the rune for day or daylight and the phonetic equivalent of "d." Its meanings include happiness, activity, and satisfaction. The second letter is "Berkano," which is associated with the birch tree. Its divinatory meanings include health, new beginnings, growth, plenty, and clearance. It is associated with Idun, goddess of Spring, and with fertility. It is the phonetic equivalent of "b."

Abstract

In recent years, indoor spatial data management has started to attract attention partly due to the increasing use of receptor devices (e.g., RFID readers, and wireless sensor networks) in both outdoor and indoor spaces. Applications that employ these devices are expected to span uniformly and supply seamless functionality in both outdoor and indoor spaces. What makes this impossible is the current absence of a unified account of these two types of spaces both in terms of modeling and reasoning about the models. This paper reviews and extends a recent unified model of outdoor and indoor spaces and receptor deployments in these spaces. The extended model enables modelers to capture various information pieces from the physical world. On top of the extended model, this paper hones the route observability concept, derives its powerful, bounded information-theoretic function, and demonstrates its usefulness in enhancing the reading environment. Additionally, this paper establishes a conclusive relation between a route observability and the uncertainty in tracking moving objects. The extended model enables incorporating receptor data through a probabilistic trajectory-to-route translator. This translator first facilitates the tracking of moving objects enabling the search for them to be optimized, and second permits performing high-level reasoning about points of potential traffic (over)load in outdoor and indoor spaces, so-called bottleneck points. A functional analysis illustrates the behavior of the route observability function. An experimental evaluation follows to corroborate the competitive accuracy of the translator, the high quality of the inference, and the sensibleness of the reasoning, when applied to synthetic data, and to uncleansed, real-world data obtained from tracking RFID-tagged flight baggage.

Keywords: Outdoor space, indoor space, OI-space, modeling, reasoning, RFID, moving objects, spatio-temporal databases, uncertainty, dynamic Bayesian network, sampling.

1 Introduction

Ubiquitous receptor devices are increasingly deployed in outdoor and indoor spaces (OI-spaces [25]) to enable new classes of applications that enhance human ambient awareness about the physical world. A myriad of examples exist, of which are supply chain and product life cycle management, and asset and personnel tracking. In order to support these emerging applications, so-called receptor-based systems [9] are being built with a focus on managing and analyzing the data collected by receptors. A common assumption made in spatial data management systems is that spaces under consideration are outdoor spaces (O-spaces). As a matter of fact, a considerable portion of human lives is spent indoors – what increases the size and complexity of indoor spaces (I-spaces). Nonetheless, indoor spatial data management systems are less developed than their outdoor counterparts that have GIS at their core. The unification of these two types of spaces, both in terms of modeling and reasoning about the models, is lacked so far.

A variety of applications, facilitated by receptor-based systems, need to span seamlessly both O- and I- spaces. One application is tracking, i.e., determining the location of moving objects in OI-spaces. Another application is deciding the amount of OI-spaces that is covered by receptors. A third application is determining the locations of heavy traffic in OI-spaces. Supporting these applications and others (at various levels in OI-spaces) motivates this study which makes the following contributions:

- The study reviews and extends a recent unified model of OI-spaces and receptor deployments in these spaces [8].
- Based on the extension, the study investigates the route[1] observability concept with the aim of optimizing an RFID readers deployment and enhancing the reading environment.
- The study then advances a probabilistic translator of receptor data that offers a complete and more informative insight into the locations of moving objects in OI-spaces.

[1] A particular way moving objects follow (or are carried over) in an OI-space.

- Furthermore, the study uses the translated data in order to perform high-level reasoning about points of potential traffic load in OI-spaces, so-called bottleneck points (BPs).
- Last, the study extensively evaluates the proposals made via functional analysis and experimentation with a real-world RFID dataset.

The remainder of this paper is organized as follows. Section 2 reviews the authors' recent unified model of OI-spaces and receptor deployments in these spaces [8]. Section 3 extends this model via supplementing it with various properties from the physical OI-space environment. Using the coverage weight property introduced in Section 3, and building on some solid information-theoretic foundations, the paper hones the route observability concept in Section 4, derives its bounded function from the ground up, and establishes its lower and upper bound. Additionally, the paper offers a conclusive characterization of the relation between a route observability and the uncertainty in tracking moving objects in OI-spaces. The notion of a BP is realized in Section 5. Static reasoning about this notion (independently of timestamped RFID data streams) is performed in the same section on top of the OI-space model reviewed in Section 2. Probabilistic incorporation of RFID data is carried out in Section 6 using the probabilistic trajectory-to-route translator that has the extended model in its core. This incorporation paves the way for an over-time upgrade, performed in Section 7, of the static model-based reasoning done in Section 5. Section 8 follows with a comprehensive functional analysis and experimental evaluation that analyze the route observability function, the probabilistic trajectory-to-route translator, and the dynamic BP estimate algorithms under a variety of settings. Related work is reviewed in brief in Section 9, and the paper concludes in Section 10. The proofs are given in Appendix A, and Table 1 offers a summary of the notation used throughout the paper.

Table 1: Summary of notation

Symbol	Description
$l, l_i \vert l_j$	A semantic location (a location for short) and a connection point
$(l_i, l_j), r$	A binary sub-route (a sub-route for short) and an RFID reader
$\mathcal{W}_l, \mathcal{W}_c$	The sets of locations and connection points
$\mathcal{W}_o, \mathcal{W}_m$	The sets of moving objects and sub-routes
$\mathcal{W}_r$	The set of RFID readers in a deployment
$\mathcal{D}_{oi\text{-}space}, \mathcal{D}_{rfid}$	The OI-space and RFID deployment pseudographs
c	The edge label mapping in $\mathcal{D}_{oi\text{-}space}$
c_l, c_m, c_r	The vertex label mapping, the edge label mapping, and the coverage weight mapping in $\mathcal{D}_{rfid}$
$R = (l_1 \dots l_k), obs(R)$	A route in $\mathcal{D}_{rfid}$ and its observability
obj	A moving object
$TR(obj, T)$	The trajectory of obj over T
appear-ds	The data structure of appearance records
inter-ds	The data structure of intermediate records
prob-ds	The data structure of probabilistic records
infer-ds	The data structure of inferred records
synth-ds	The data structure of synthetic records
$E_{BP}(l)$	The static estimate that l is a BP
$E^T_{BP}(l)$	The dynamic estimate that l is a BP over T
$BPMQ^T$	A BP monitoring query over T

2 Model Review

An authors' recent work [8] proposes a unified model of OI-spaces and receptor deployments in these spaces. The work focuses on partially constrained outdoor and indoor motion common in receptor-based systems. The model is shown to be expressive, flexible, and invariant to the segmentation of a space plan, and the receptor deployment policy. The viability of this model is demonstrated via applying it to the

real-world baggage handling plan in Aalborg Airport. This plan comprises two sub-plans; the I-space and O-space plans in Aalborg Airport hall and apron[2] respectively. The former plan is shown in Figure 1.

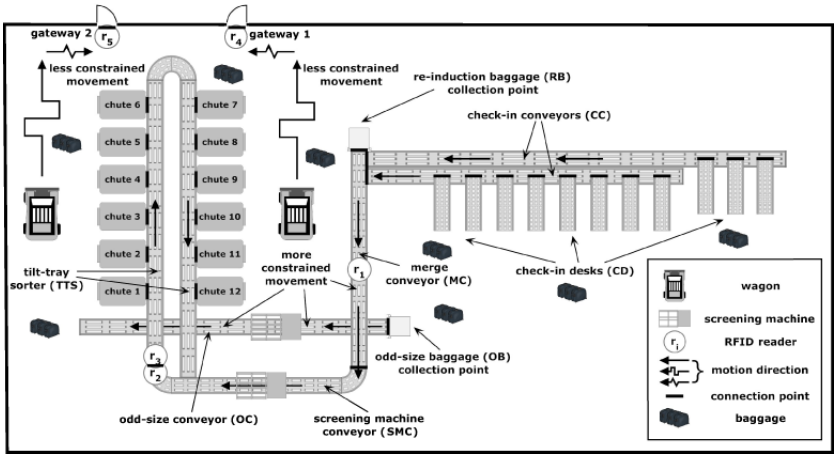


Figure 1: The example I-space plan in Aalborg Airport hall

To create the OI-space pseudograph model of the baggage handling example, the sets of *locations* ($\mathcal{W}_l$), *connection points* ($\mathcal{W}_c$), *moving objects* ($\mathcal{W}_o$), and *sub-routes* ($\mathcal{W}_m$) are identified. To give examples in Figure 1, the check-in desks (CD) and check-in conveyor (CC) are locations and (CD|CC) is their connection point. Moving objects are bags to which RFID tags are attached. An example bag route is CD $\rightarrow$ CC $\rightarrow$ MC $\rightarrow$ SMC $\rightarrow$ TTS (repeatedly in general) $\rightarrow$ CH. Two sub-routes along this route are $(\mathrm{CD}, \mathrm{CC})$ and $(\mathrm{TTS}, \mathrm{TTS})$. Next, the locations are converted into vertices and sub-routes into edges (an edge direction matches the motion direction and the order of the sub-route). Furthermore, the edges are labeled using sets taken from the power set of the connection points. For instance in Figure 1, the locations CD and CC are converted into vertices, and the sub-route $(\mathrm{CD}, \mathrm{CC})$ is converted into an edge connecting between these two vertices. The edge $(\mathrm{CD}, \mathrm{CC})$ is directed from CD to CC and labeled (CD|CC). The same identification, conversion, and labeling steps are carried out for Aalborg Airport apron which yields $\mathcal{D}_{oi\text{-}space} = (\mathcal{W}_l, \mathcal{W}_m, c)$ shown in Figure 2, where c is the edge label mapping. In this figure, CGS and GS1-GS4 are the apron geometric segments, BL1-BL3 are the belt loaders, and AP1-AP3 are the airplanes.

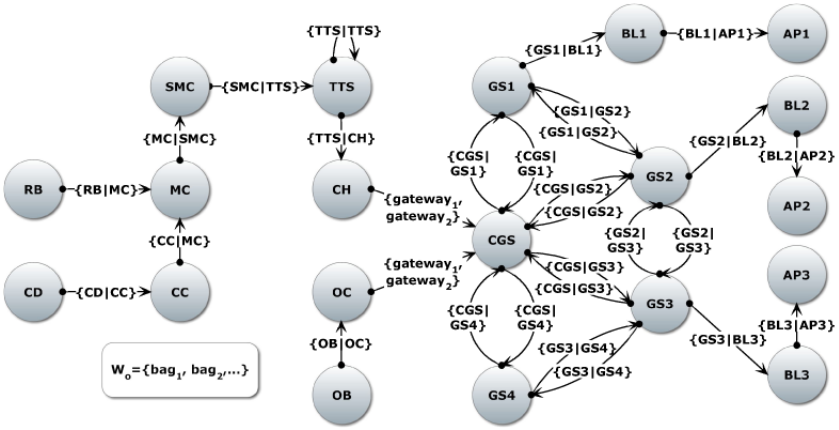


Figure 2: The example OI-space pseudograph

[2] An open part of an airport in which airplanes are parked, fueled, boarded by passengers, and loaded with baggage.

In the same work [8], Aalborg Airport RFID deployment (Figure 1) is modeled. An algorithm is applied in order to transform $\mathcal{D}_{oi\text{-}space} = (\mathcal{W}_l, \mathcal{W}_m, c)$ (Figure 2) into $\mathcal{D}_{rfid} = (\mathcal{W}_l, \mathcal{W}_m, c_l, c_m)$ (Figure 3), where c_l and c_m are the vertex and edge label mappings respectively. To give examples on vertex and edge labeling in Figure 3, the reader r_1 is positioned inside MC away from any connection point. Therefore $r_1 \in c_l(\mathrm{MC})$. On the other hand, r_2 and r_3 are adjacently positioned at $\mathrm{SMC}|\mathrm{TTS}$, and r_2 reads before r_3 when moving from SMC to TTS across $\mathrm{SMC}|\mathrm{TTS}$. Thus, $(r_2, r_3) \in c_m(\mathrm{SMC}, \mathrm{TTS})$.

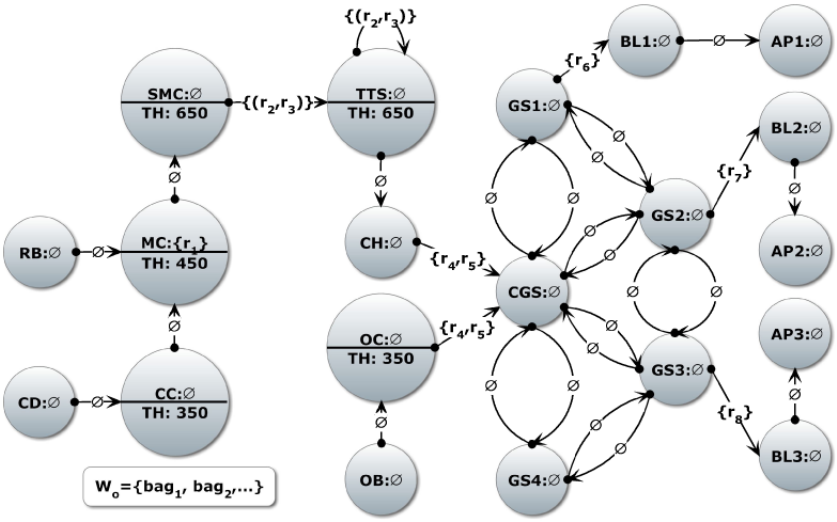


Figure 3: The example RFID readers deployment pseudograph extended with the conveyor throughput property TH (measured in bags/hour)

3 Model Extension

This work extends $\mathcal{D}_{rfid}$ (Figure 3) into a property pseudograph [19] by allowing the vertices and edges to have various properties (key/value pairs) from the physical hall and apron environments. The advantage of this extension is threefold. First, a property pseudograph gives the freedom to modelers in expressing their awareness of various information pieces from the physical world. Intuitively, the more the information gathered about the physical world, the wider the scope of the questions that can be asked about it. Second, a property pseudograph contains most of the pieces used in graph modeling, which makes it a malleable structure that can be easily transformed into other common graph structures. Third, a property pseudograph is the typical data model used in graph databases. Therefore, the extension to this type of graph enables benefiting from the proven efficiency of graph databases in processing dense and interrelated datasets and quickly traversing along the edges between vertices [19].

Various properties can be obtained from the hall (Figure 1) and apron environments and subsequently added to the vertices and edges of $\mathcal{D}_{rfid}$ (Figure 3). A few example properties are conveyor type (with values chain and curve conveyors, etc), conveyor throughput (measured in bags/hour), speed limit on the apron geometric segments (measured in m/s), in addition to connection point type (with values actual or virtual). An important property to the route observability application (Section 4) is the coverage weight of RFID readers among locations. This vertex property can be captured in a mapping $c_r : \mathcal{W}_l \to (\mathcal{W}_r \to [0, 1])$, which is effectively a mapping to a mapping in the sense that it maps any location $l \in \mathcal{W}_l$ to a set of assignments each of which specifies the coverage weight of a reader $r \in \mathcal{W}_r$ whose reading zone is joint

(overlapping/nested) with the area of l. Strictly speaking:

$$c_r : \mathcal{W}_l \rightarrow w; w : \mathcal{W}_r \rightarrow [0,1]$$
$$c_r(l) = \{r \rightarrow w(r) = \frac{ZONE(r) \cap AREA(l)}{AREA(l)} : ZONE(r) \cap AREA(l) \neq \emptyset\}$$

It is also convenient to use the simplifying notation:

$$\overline{c_r(l)} = \sum_{w(r) \in c_r(l)} w(r) \tag{1}$$

The coverage weights of the locations seen in Figure 3 are approximated based on the RFID deployment (Figure 1) and listed in Table 2. For instance, $c_r(\mathrm{SMC}) = \{r_2 \rightarrow .8, r_3 \rightarrow .2\}$ means that roughly 80% of r_2 and 20% of r_3 reading zones overlap with $AREA(\mathrm{SMC})$. The notation of the RFID pseudograph becomes $\mathcal{D}_{rfid} = (\mathcal{W}_l, \mathcal{W}_m, c_l, c_m, c_r)$, augmented by c_r; the new coverage weight mapping.

Table 2: The coverage weights of the locations in Figure 3

l	$c_r(l) = \{r \rightarrow w(r)\}$	l	$c_r(l) = \{r \rightarrow w(r)\}$
CD, CC, GS4, AP1, OB, RB, AP2, AP3	$\emptyset$	GS1	$\{r_6 \rightarrow .1\}$
MC	$\{r_1 \rightarrow 1\}$	GS2	$\{r_7 \rightarrow .1\}$
SMC	$\{r_2 \rightarrow .8, r_3 \rightarrow .2\}$	GS3	$\{r_8 \rightarrow .1\}$
TTS	$\{r_2 \rightarrow .2, r_3 \rightarrow .8\}$	BL1	$\{r_6 \rightarrow .9\}$
OC	$\{r_4 \rightarrow .95, r_5 \rightarrow .95\}$	BL2	$\{r_7 \rightarrow .9\}$
CH	$\{r_4 \rightarrow .95, r_5 \rightarrow .95\}$	BL3	$\{r_8 \rightarrow .9\}$
CGS	$\{r_4 \rightarrow .05, r_5 \rightarrow .05\}$		

4 Route Observability

A route observability is a measure of the extent to which a given route is covered by RFID readers. The study of a route observability is motivated as follows. Some physical approaches to RFID deployment in OI-spaces attempt to correct RFID anomalies by enhancing the reading environment. This can be attained through either installing additional readers or (more economically) adjusting the positioning of already-installed readers. In both cases, the aim is to cover a more substantial amount of an OI-space [1]. The number of RFID readers positioned along a route does not accurately reflect this route observability. For instance, route_2 in Figure 4 is more observable than route_1. The reason is that 100% of r_5 reading zone overlaps with $AREA(\mathrm{gateway}_2)$ compared to the 50% of r_4 reading zone that overlaps with $AREA(\mathrm{gateway}_1)$. The need for a precise route observability measure is thus overwhelming.

Figure 4: route_2 is more observable than route_1 albeit both routes are covered by one reader.

The route definition can be made more formal by saying that a route $R = (l_1 \ldots l_k)$ in $\mathcal{D}_{rfid}$ is an alternating sequence of locations and sub-routes from $\mathcal{D}_{rfid}$ that starts in l_1 and ends in l_k. The sets of locations and sub-routes in R are denoted as $\mathcal{V}(R)$ and $\mathcal{A}(R)$ respectively. Hence, one may write $R =$

$(\mathcal{V}(R), \mathcal{A}(R))$. The starting point in deriving an observability measure is to consider a single location observability and then generalize it into routes with arbitrary number of locations. The observability of a location l can be denoted as follows:

$$\Gamma : [0, 1] \rightarrow [0, \infty)$$

To measure the observability in a meaningful way, the function Γ should satisfy the following properties:

P1. Nonnegativity: $\forall w(r) \in c_r(l) : \Gamma(w(r)) \geq 0$.

P2. Increasing monotonicity: $\forall w(r_1), w(r_2) \in c_r(l) : w(r_1) \leq w(r_2) \Rightarrow \Gamma(w(r_1)) \leq \Gamma(w(r_2))$.

P3. Normalization: If the whole coverage of a reader r is contained within the location l, then the observability should be 1, that is $\forall w(r) \in c_r(l) : w(r) = 1 \Rightarrow \Gamma(w(r)) = 1$.

Intuitively, a location observability should be expressed by an increasing function of the coverage weights: the higher these weights, the higher the observability. This justifies including P2, and makes P1 convenient to have. The property P3 is a requirement for the measurement unit and it can be modified accordingly. One class of functions that satisfy P1-P3 is defined for each $w(r) \in [0, 1]$ by the formula:

$$\Gamma(w(r)) = a \log_b(w(r) + 1)$$

where a is an arbitrary constant and b is a nonnegative constant different from 1. Adding 1 to $w(r)$ satisfies P1. Since the logarithmic function is increasing, satisfying P2 entails a nonnegative a. P3 can be formally expressed by the equation:

$$a \log_b(1 + 1) = 1$$

This equation can be satisfied by choosing $a = 1$ and $b = 2$ making bits the measurement unit of a location observability. The function becomes: (Here and hereafter, all logarithms are to the base 2).

$$\Gamma(w(r)) = \log(w(r) + 1)$$

One more desirable property is the finite additivity which can be expressed as follows:

P4. Finite additivity: For every finite sequence of pairwise disjoint routes, the observability of a union of these routes equals the sum of the individual observabilities.

This property enables measuring the observability of routes with arbitrary number of locations. Furthermore, it enables the concatenation of routes. In order to satisfy P4, one takes the expected value function of $\Gamma(w(r))$, and then sums for all $l \in \mathcal{V}(R)$, which yields the novel route observability function:

$$obs(R) = \sum_{l \in \mathcal{V}(R)} \sum_{w(r) \in c_r(l)} w(r) \log(w(r) + 1) \tag{2}$$

The obs function has solid bounds that are sought in Theorem 1. These bounds are important in that they delimit the optimization that can be introduced into an RFID readers deployment.

Theorem 1 *Given an RFID readers deployment pseudograph $\mathcal{D}_{rfid} = (\mathcal{W}_l, \mathcal{W}_m, c_l, c_m, c_r)$, the observability of any route R in $\mathcal{D}_{rfid}$ has the bounds:*

$$0 \leq obs(R) \leq \sum_{l \in \mathcal{V}(R)} \log\left(\overline{c_r(l)} + |c_r(l)|\right)$$

Table 3: Some baggage routes in Figure 3, their observabilities, and bounds

R baggage type	(CD . . . AP1) normal-size	(RB . . . AP2) security-checked	(OB . . . AP3) odd-size
$obs(R)$	5.1468	5.1468	2.6848
$bounds(R)$	$[0, 8.2673]$	$[0, 8.2673]$	$[0, 4.0974]$

Revisiting $\mathcal{D}_{rfid}$ in Figure 3 and the coverage weights in Table 2, some baggage routes, their observabilities, and bounds are listed in Table 3. Notice in this table that the observabilities of the chosen routes are less than their maximum, attainable values. This suggests a possibility to adjust Aalborg Airport RFID deployment in order to better the reading environment and thereby reduce or even eliminate the occurrence of RFID anomalies. A modeler can experiment with different scenarios of RFID readers positioning, and monitor the change in the observabilities until optimum values (those closer to the upper bound) are obtained. Three such scenarios are chosen and analyzed in Section 8.1. As a matter of fact, the adequacy of the observabilities obtained using the obs function is dependent on the purpose of building the RFID-based system. For instance, higher route observabilities (and less uncertainty in tracking moving objects) are more crucial to have in safety- and business-critical RFID-based systems. Indeed, the relation between a route observability and the uncertainty in tracking moving objects[3] along this route can be characterized and proved as follows.

Lemma 1 *The higher a route observability, the less the uncertainty in tracking moving objects along this route.*

5 Static Model-Based Reasoning

Since the OI-space model is graph-based (Figure 2), one can rely on the fundamentals of graph theory to perform high-level reasoning that gives a better insight into the physical world. The model-based reasoning, described in this section, deals with the concepts of a BP in a static, time-independent fashion, i.e., independently of timestamped RFID data streams. This reasoning is hence important at planning stages that precede the actual RFID deployment.

A BP is a location in an OI-space where there is potentially a lot of traffic. Given an OI-space pseudograph (Figure 2), one can postulate a static estimate about BPs by borrowing the concept of a vertex pseudodegree. A vertex pseudodegree is the number of all directed edges (including loops) whose head or tail is this vertex. The pseudodegree of a vertex v is denoted as $d(v)$. A very basic result in graph theory tells that the sum of pseudodegrees in a directed pseudograph $\mathcal{D} = (\mathcal{V}, \mathcal{A})$ is twice the number of edges in $\mathcal{D}$. Strictly speaking:

$$\sum_{v \in \mathcal{V}(\mathcal{D})} d(v) = 2|\mathcal{A}(\mathcal{D})|$$

In the case of $\mathcal{D}_{oi\text{-}space} = (\mathcal{W}_l, \mathcal{W}_m, c)$, the formula becomes:

$$\sum_{l \in \mathcal{W}_l} d(l) = 2|\mathcal{W}_m| \tag{3}$$

Definition 1 characterizes the static estimate about BPs.

[3] A quantity that emerges due to partial observability in an RFID-based system, nondeterminism in interpreting raw RFID data streams, or a combination of the two.

Definition 1 *(Static BP Estimate) Given an OI-space pseudograph $\mathcal{D}_{oi\text{-}space} = (\mathcal{W}_l, \mathcal{W}_m, c)$, the static support degree that $l \in \mathcal{W}_l$ is a BP is estimated by the ratio of l's pseudodegree to twice the number of edges in $\mathcal{D}_{oi\text{-}space}$. Formally speaking:*

$$\forall l \in \mathcal{W}_l : E_{BP}(l) = \frac{d(l)}{2|\mathcal{W}_m|}$$

The nature of E_{BP} is explored in Lemma 2.

Lemma 2 *Given an OI-space pseudograph $\mathcal{D}_{oi\text{-}space} = (\mathcal{W}_l, \mathcal{W}_m, c)$, E_{BP} is consistent as a probability distribution on a random variable whose alphabet is $\mathcal{W}_l$.*

Revisiting $\mathcal{D}_{oi\text{-}space}$ (Figure 2), the locations, their pseudodegrees, and the static estimates that they are BPs are listed in Table 4. It is important to notice that the sorter loop is counted twice when deciding $d(\mathrm{TTS})$. Thus, CGS has the highest static support degree of being a BP in the hall and apron of Aalborg Airport, followed equally by GS2 and GS3, then by GS1, and after it by GS4 and TTS with equal static support. The relatively high likelihood $(4/60)$ of suffocation by baggage in TTS suggests a need for careful deployment of RFID readers in this location.

Table 4: The locations, their pseudodegrees, and static BP estimates in $\mathcal{D}_{oi\text{-}space}$ (Figure 2)

l	CD	CC	MC	SMC	TTS	CH	RB
d	1	2	3	2	4	2	1
E_{BP}	1/60	2/60	3/60	2/60	4/60	2/60	1/60
l	OB	OC	CGS	GS1	GS2	GS3	GS4
d	1	2	10	5	7	7	4
E_{BP}	1/60	2/60	10/60	5/60	7/60	7/60	4/60
l	BL1	BL2	BL3	AP1	AP2	AP3	
d	2	2	2	1	1	1	
E_{BP}	2/60	2/60	2/60	1/60	1/60	1/60	

6 Probabilistic Incorporation of RFID Data

A probabilistic account of RFID data is crucial to compensate for the missing information that is inherent in this data. This section deals with the probabilistic incorporation of RFID data streams. The incorporation attained in this section offers complete and more informative knowledge about the locations of moving objects in OI-spaces. This knowledge facilitates the tracking of these objects and enables the search for them to be optimized (this will be explained later in this section). Preliminaries are offered in section 6.1 and the novel probabilistic translator follows in section 6.2.

6.1 Preliminaries

A raw RFID reading can be denoted as a triple of the form $\mathrm{rd} \equiv \langle \mathrm{obj\text{-}id}, \mathrm{reader\text{-}id}, \mathrm{time} \rangle$ which indicates that the tag affixed to $\mathrm{obj\text{-}id}$ was detected by $\mathrm{reader\text{-}id}$ at timestamp time. An RFID data stream produced by all the readers in a deployment is then a stream of triples $\mathrm{S} \equiv \langle \mathrm{rd}_1, \mathrm{rd}_2, \ldots \rangle$. Minding the efficiency of query processing, one does not want to store and persistently manipulate raw RFID readings at the timestamp level. Instead one would like to store the first and last detection of a tag by a reader, i.e., the appearance of a moving object in a reader's reading zone over a closed time period. Thus, the level of raw RFID readings is lifted by employing a pre-processing module (the details of which can be found elsewhere [10]) that condenses these readings into so-called appearance records. Each appearance record has the form

$\text{ar} \equiv \langle \text{ar-id}, \text{obj-id}, \text{reader-id}, \text{s-time}, \text{e-time} \rangle$ where s-time and e-time are the start and end time of an appearance. These appearance records are stored in the data structure appear-ds. A definition of a moving object trajectory can be given at this stage.

Definition 2 *(Moving Object Trajectory) A trajectory [2] of a moving object obj inside an RFID data stream S over a time period T is the sequence of appearance records whose detected object is obj and detection time is in T.*

$$TR(obj, T) = ar_1, ar_2, \ldots, ar_n : ar_i.obj\text{-}id = obj \wedge [ar_i.s\text{-}time, ar_i.e\text{-}time] \subseteq T$$

For instance, the trajectory of bag_1 in Figure 3 during $[t_1, t_{37}]$ is $TR(bag_1, [t_1, t_{37}]) = ar_1, ar_2, \ldots, ar_7$. Table 5 lists the corresponding appearance records.

Table 5: The trajectory of bag_1 during $[t_1, t_{37}]$

ar-id	obj-id	reader-id	s-time	e-time
ar_1	bag_1	r_1	t_1	t_2
ar_2	bag_1	r_2	t_5	t_6
ar_3	bag_1	r_3	t_7	t_8
ar_4	bag_1	r_2	t_{11}	t_{12}
ar_5	bag_1	r_3	t_{13}	t_{14}
ar_6	bag_1	r_4	t_{19}	t_{29}
ar_7	bag_1	r_7	t_{32}	t_{37}

6.2 Probabilistic Trajectory-to-Route Translator

Given a trajectory $TR(obj, T)$ of a moving object obj over a time period T (Table 5), one wants to infer the route that this object followed (or was carried over) between locations over the same period. This is achieved through the probabilistic trajectory-to-route translator in Algorithm 1 which comprises three stages:

Stage 1. Translation based on $\mathcal{D}_{rfid}$: Based on the vertex and edge labels in $\mathcal{D}_{rfid}$, this stage translates the appearance records in $TR(obj, T)$ into intermediate records in the inter-ds. Notice that the loop in line 3 terminates at the completion of one insertion, since it is not expected that a reader is positioned inside more than one location, neither is it expected to be simultaneously positioned inside a location and at a connection point. On the contrary, the loop in line 7 does not terminate at the completion of one insertion, since it is possible for more than one sub-route to have shared elements in their labels (Figure 3).

Stage 2. Transformation: This stage condenses the intermediate records in the inter-ds into a smaller number of probabilistic records that are pushed into the prob-ds. The transformation is done using a simple SQL query (this query effect will be shown later in an example).

Stage 3. Inferring the information gaps: The gaps in RFID data streams are unavoidable due to RFID anomalies as well as the economical and practical intractability of covering a whole OI-space with RFID readers. Stage 3 aims at inferring the information gaps in the prob-ds by borrowing from the prior knowledge available about the RFID-based system. The inference is based on the dynamic Bayesian network (DBN) in Figure 5a [20, 12]. In this DBN, the location L_t and reader R_t are two random variables whose alphabets are $\mathcal{W}_l$ and $\mathcal{W}_r$ respectively. L_t denotes an obj's location, whereas R_t denotes a reader detecting obj's tag. In probabilistic reasoning texts, L_t and R_t are referred to as the state and evidence variables respectively. Additionally, R_t is observable while L_t is not. The DBN world in Figure 5a is viewed as a series of time slices each of which contains L_t and R_t. The interval between these slices depends on the problem considered, and it is parameterized as inv in the input to Algorithm 1. Three kinds of information specify the DBN in Figure 5a:

Algorithm 1 Probabilistic Trajectory-to-Route Translator

Input: $\mathcal{D}_{rfid} = (\mathcal{W}_l, \mathcal{W}_m, c_l, c_m, c_r)$, $TR(obj, T)$, a dynamic Bayesian network (DBN), the interval between time slices inv (a positive integer), and the data structures of intermediate, probabilistic, and inferred records:

- inter-ds $\equiv \langle$ar-id, obj-id, loc, s-time, e-time$\rangle$
- prob-ds $\equiv \langle$obj-id, prob-loc, s-time, e-time$\rangle$
- infer-ds $\equiv \langle$obj-id, infer-loc, s-time, e-time$\rangle$

where loc is a location, prob-loc and infer-loc are probability distributions on a random variable whose alphabet is $\mathcal{W}_l$.

Output: Records in the inter-ds, prob-ds, and infer-ds.

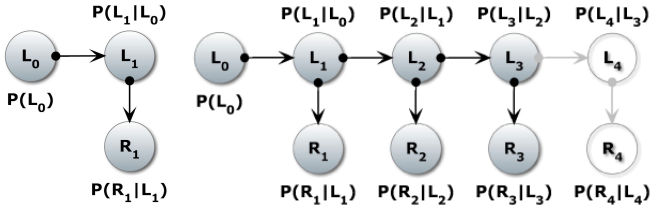
inter-ds $\leftarrow \emptyset$; prob-ds $\leftarrow \emptyset$; infer-ds $\leftarrow \emptyset$;
for each $ar_i \in TR(obj, T)$ **do**
// Stage 1. *Translation based on* $\mathcal{D}_{rfid}$.
  for each $l \in \mathcal{W}_l$ **do**
    if $ar_i.reader\text{-}id \in c_l(l)$ **then**
      insert $\langle ar_i, ar_i.obj\text{-}id, l, ar_i.s\text{-}time, ar_i.e\text{-}time\rangle$ into inter-ds
      break
  for each $m = (l_i, l_j) \in \mathcal{W}_m : l_i, l_j \in \mathcal{W}_l$ **do**
    if $(ar_i.reader\text{-}id \in c_m(m)$ or $(ar_i.reader\text{-}id, ar_{i+1}.reader\text{-}id) \in c_m(m))$ **then**
      insert $\langle ar_i, ar_i.obj\text{-}id, l_i, ar_i.s\text{-}time, ar_i.e\text{-}time\rangle$ and
      $\langle ar_i, ar_i.obj\text{-}id, l_j, ar_i.s\text{-}time, ar_i.e\text{-}time\rangle$ into inter-ds
// Stage 2. *Transformation.*
Transform inter-ds into prob-ds.
// Stage 3. *Inferring the information gaps.*
for each $p\text{-}rec_i \in$ prob-ds **do**
  inject $p\text{-}rec_i.prob\text{-}loc$ and $p\text{-}rec_{i+1}.prob\text{-}loc$ as evidence into DBN
  update DBN beliefs using EPIS-BN
  $bel1 \leftarrow first\text{-}DBN\text{-}belief$
  $beln \leftarrow last\text{-}DBN\text{-}belief$
  insert $\langle p\text{-}rec_i.obj\text{-}id, bel1, p\text{-}rec_i.s\text{-}time, p\text{-}rec_i.e\text{-}time\rangle$ and
  $\langle p\text{-}rec_{i+1}.obj\text{-}id, beln, p\text{-}rec_{i+1}.s\text{-}time, p\text{-}rec_{i+1}.e\text{-}time\rangle$ into infer-ds
  $start \leftarrow p\text{-}rec_i.e\text{-}time + inv$
  $end \leftarrow p\text{-}rec_{i+1}.s\text{-}time - 1$
  if $start <= end$ **then**
    evolve infer-loc from DBN
    insert $\langle p\text{-}rec_i.obj\text{-}id$, infer-loc, $start, end\rangle$ into infer-ds

1. The transition model $P(L_{t+1}|L_t)$: It describes the likelihood of obj's location at the next slice given its location at the current slice.

2. The sensor model $P(R_t|L_t)$: It describes the likelihood of detecting obj's tag given its location at the current slice.

3. The prior model $P(L_0)$: It describes the likelihood of obj's location at slice 0.

From this specification, a complete DBN with an unbounded number of slices can be constructed as needed by copying the first slice. The unrolled DBN over five slices is shown in Figure 5b. Observer in Figure 5b that the current state depends only on the previous state and not on any earlier states. This is due to the first-order Markov process assumption which is commonly made when reasoning over time. Another important assumption that is made in Figure 5b is that the changes in the DBN world are caused by a process whose laws are static over time. With this assumption in place, only one $P(L_{t+1}|L_t)$ and $P(R_t|L_t)$ has to be specified albeit the unrolled DBN may have infinitely many slices.



(a) DBN (b) Unrolled DBN

Figure 5: Figure (a) shows the DBN used for inferring the information gaps; and Figure (b) shows the unrolled DBN over five time slices.

The distributions $P(L_0)$ and $P(L_{t+1}|L_t)$ can be specified based on $\mathcal{D}_{rfid}$ in Figure 3 as follows. Regarding $P(L_0)$, uniform probabilities are ascribed to entry locations and zero probabilities are ascribed to the rest. For instance in Figure 3, CD, RB, and OB are given a probability $1/3$ each, while the rest of the locations are given a probability 0. Turning to $P(L_{t+1}|L_t)$, it is generically sufficient to consider the location at slice t and the two locations that follow it along a route in Figure 3. If any of these three locations has a loop, then 50% probability is ascribed to it while the remainder 50% is uniformly distributed to the rest two locations. If none of these locations has a loop, then uniform probabilities are ascribed to all of them. For instance in Figure 3, if the location at t is SMC, then the location at $t+1$ is SMC, TTS, or CH with probabilities .25, .5, and .25 respectively. The distribution $P(R_t|L_t)$, on the other hand, can be specified based on the coverage weights in Table 2. For instance, if the location at t is SMC, then r_2 or r_3 detects obj's tag at t with probabilities .8 and .2 respectively.

The pieces of evidence available in the prob-ds (outcome of stage 2) are injected into the DBN's L_t nodes in line 12 of Algorithm 1. The purpose of evidence injection is to incorporate the prior knowledge about moving objects available in the prob-ds. Incorporation of prior knowledge has the desirable impact of amplifying the inference quality. Following the evidence injection, the DBN beliefs can be updated. DBN Belief updating is computationally complex that several algorithms were developed to cope with this complexity. These algorithms fall into two categories; exact and approximate belief updating. Algorithms for exact belief updating (e.g., variable elimination [20, 12], polytree [17] and clustering [11]) were shown to have exponential space and time complexities in the number of state variables when applied to a DBN. Therefore we must fall back on approximate algorithms. The most widely used algorithm in the database literature [18, 24, 26] for approximate belief updating is sequential importance sampling (also known as

particle filtering) [4]. The approximate algorithm used in line 13 of Algorithm 1 is the Estimated Posterior Importance Sampling algorithm for Bayesian Networks (EPIS-BN) [28]. EPIS-BN uses loopy belief propagation to compute an estimate of the posterior probability over all DBN nodes and then refines this estimate via importance sampling. EPIS-BN is quite likely the best approximate algorithm available to date. In addition to being faster, it produces results that are an order of magnitude more precise than other algorithms. The beliefs that evolve from applying EPIS-BN are used to populate the infer-ds in lines 14-21 of Algorithm 1.

Before exemplifying the operation of Algorithm 1, it is good to stress that a DBN is quite likely the best Bayesian filtering method for location estimation [3]. A DBN, for instance, surpasses a Hidden Markov Model (HMM) in its ability to model domains with many state variables. A DBN also outperforms a Kalman filter [21] in which very strong Gaussian assumptions are made. These assumptions limit the applicability of a Kalman filter to location estimation using accurate RFID readers (readers that do not produce many RFID anomalies in the reported data).

Applying stage 1 of Algorithm 1 to the trajectory of bag_1 in Table 5, yields the intermediate records in Table 6. Stage 2 transforms the content of Table 6 into the probabilistic records in Table 7. As an example that demonstrates the effect of the SQL query applied in stage 2, note that TTS appears three times in records 3-5 in Table 6, therefore TTS probability is .75 in record 2 of Table 7. Next, stage 3 has to be applied in order to infer the information gaps $[t_3, t_4]$, $[t_9, t_{10}]$, $[t_{15}, t_{18}]$, and $[t_{30}, t_{31}]$ in Table 7. Parameterizing stage 3 with $inv = 1$ and proceeding with the computations yield the inferred records in Table 8. These records correspond to the inferred route of bag_1. The temporal evolution of the probabilities in this route is conveniently plotted in Figure 6.

Table 6: The intermediate records of bag_1 during $[t_1, t_{37}]$

ar-id	obj-id	loc	s-time	e-time
ar_1	bag_1	MC	t_1	t_2
ar_2	bag_1	SMC	t_5	t_6
ar_2	bag_1	TTS	t_5	t_6
ar_2	bag_1	TTS	t_5	t_6
ar_2	bag_1	TTS	t_5	t_6
ar_3	bag_1	SMC	t_7	t_8
ar_3	bag_1	TTS	t_7	t_8
ar_3	bag_1	TTS	t_7	t_8
ar_3	bag_1	TTS	t_7	t_8
ar_4	bag_1	SMC	t_{11}	t_{12}
ar_4	bag_1	TTS	t_{11}	t_{12}
ar_4	bag_1	TTS	t_{11}	t_{12}
ar_4	bag_1	TTS	t_{11}	t_{12}
ar_5	bag_1	SMC	t_{13}	t_{14}
ar_5	bag_1	TTS	t_{13}	t_{14}
ar_5	bag_1	TTS	t_{13}	t_{14}
ar_5	bag_1	TTS	t_{13}	t_{14}
ar_6	bag_1	CH	t_{19}	t_{29}
ar_6	bag_1	CGS	t_{19}	t_{29}
ar_6	bag_1	OC	t_{19}	t_{29}
ar_6	bag_1	CGS	t_{19}	t_{29}
ar_7	bag_1	GS2	t_{32}	t_{37}
ar_7	bag_1	BL2	t_{32}	t_{37}

Contemplating the content of Table 8 in comparison to Table 5 enables one to realize that the knowledge obtained from the translator is both (1) complete and (2) more informative about the locations of baggage in transit. To clarify (1), note that Table 8 communicates full observability of bag_1 during $[t_1, t_{37}]$, whereas the observability delivered is only partial in Table 5 during the same period (note the information gaps $[t_3, t_4]$, $[t_9, t_{10}]$, $[t_{15}, t_{18}]$, and $[t_{30}, t_{31}]$). To give an example on (2), ar_5 in Table 5 tells that bag_1 passed under r_3 during $[t_{13}, t_{14}]$. Due to the adjacent positioning of r_2 and r_3 (Figure 1), this information piece is deficient

Table 7: The probabilistic records of bag_1 during $[t_1, t_{37}]$

obj-id	prob-loc	s-time	e-time
bag_1	[MC : 1]	t_1	t_2
bag_1	[SMC : .25, TTS : .75]	t_5	t_6
bag_1	[SMC : .25, TTS : .75]	t_7	t_8
bag_1	[SMC : .25, TTS : .75]	t_{11}	t_{12}
bag_1	[SMC : .25, TTS : .75]	t_{13}	t_{14}
bag_1	[CH : .25, OC : .25, CGS : .5]	t_{19}	t_{29}
bag_1	[GS2 : .5, BL2 : .5]	t_{32}	t_{37}

Table 8: The inferred route of bag_1 during $[t_1, t_{37}]$

obj-id	infer-loc	s-time	e-time
bag_1	[MC : 1]	t_1	t_2
bag_1	[MC : .39, SMC : .40, TTS : .21]	t_3	t_4
bag_1	[SMC : .30, TTS : .70]	t_5	t_6
bag_1	[SMC : .14, TTS : .86]	t_7	t_8
bag_1	[SMC : .07, TTS : .93]	t_9	t_{10}
bag_1	[SMC : .02, TTS : .98]	t_{11}	t_{12}
bag_1	[SMC : .01, TTS : .99]	t_{13}	t_{14}
bag_1	[SMC : .01, TTS : .57, CH : .28, CGS : .14]	t_{15}	t_{18}
bag_1	[CH : .29, CGS : .71]	t_{19}	t_{29}
bag_1	[CH : .04, CGS : .18, GS1 : .18, GS2 : .13, GS3 : .14, GS4 : .10, BL2 : .23]	t_{30}	t_{31}
bag_1	[GS2 : .46, BL2 : .54]	t_{32}	t_{37}

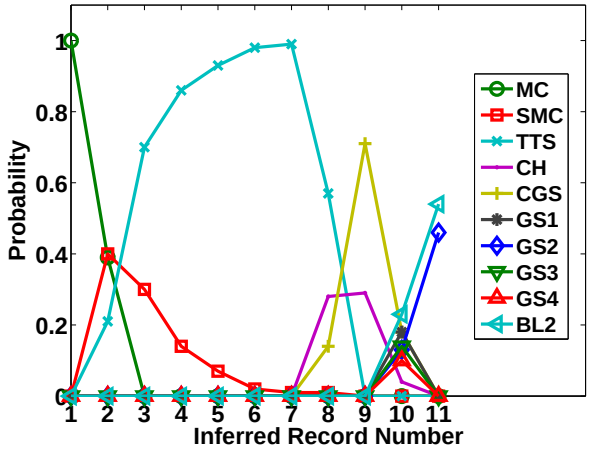


Figure 6: The temporal evolution of the probabilities in the inferred route of bag_1 given in Table 8.

and possibly inaccurate. Contrary to this, the seventh row in Table 8 tells that bag_1 is highly likely to be at TTS and less likely to be at SMC during $[t_{13}, t_{14}]$. All in all, the translator better facilitates the tracking of baggage in Aalborg Airport and enables the search for lost baggage to be optimized.

7 Dynamic Model-Based Reasoning

The dynamic nature of moving objects in RFID-based systems that evolves over time makes it useful to model time explicitly. The static reasoning (conducted in Section 5) is meant for pre-RFID-deployment phases and hence does not consider the growth of moving objects trajectories over time. Having incorporated RFID data and inferred the routes of moving objects in Section 6, dynamic, time-dependent reasoning

about BPs can be done in this section.

Definition 3 *(Dynamic BP Estimate) Given an RFID readers deployment pseudograph $\mathcal{D}_{rfid} = (\mathcal{W}_l, \mathcal{W}_m, c_l, c_m, c_r)$, the* infer-ds*, and a monitoring period T of a location $l \in \mathcal{W}_l$, the dynamic support degree over T that l is a BP is estimated by the joint probability distribution on all the random variables of the inferred records in the* infer-ds *whose detection time is joint (overlapping/nested) with T. Formally speaking:*

$$\forall l \in \mathcal{W}_l : E^T_{BP}(l) = Pr(obj_1 \; at \; l, \ldots, obj_n \; at \; l) : obj_i \in \mathcal{W}_o$$

Definition 4 *(Dynamic BP Monitoring Query) A dynamic BP monitoring query (BPMQ) takes as input an RFID readers deployment pseudograph $\mathcal{D}_{rfid} = (\mathcal{W}_l, \mathcal{W}_m, c_l, c_m, c_r)$, the* infer-ds*, and a monitoring period T. It then reports $E^T_{BP}(l)$ for all $l \in \mathcal{W}_l$. To put in symbols:*

$$BPMQ^T = \{E^T_{BP}(l) : l \in \mathcal{W}_l\}$$

It is known that inference using joint distributions has prohibitive time complexity [20]. This complexity can be coped with in two ways. First, the specification of a monitoring period T limits the inference to only a subset $I\text{-}REC(T)$ of the infer-ds:

$$I\text{-}REC(T) = \{i\text{-}rec \in \text{infer-ds} : [i\text{-}rec.s\text{-}time, i\text{-}rec.e\text{-}time] \cap T \neq \emptyset\}$$

Second, the absolute independence assertion, imposed on random variables, radically reduces the amount of information necessary to encode the joint distribution by enabling the factoring of this distribution into separate, smaller distributions.

An algorithm for answering a BPMQ is given in Algorithm 2. The probability tweaking parameter η, in the input to this algorithm, is fed as a percentage, and it specifies the quotient by which all E^T_{BP} are (de)concentrated in accordance with the detection time. The normalization function ψ normalizes E^T_{BP} generated by the algorithm by dividing each $E^T_{BP}(l)$ by the sum of all E^T_{BP}. This normalization transforms E^T_{BP} and ensures its consistency as a probability distribution on a random variable whose alphabet is $\mathcal{W}_l$. Capturing the dynamic estimates in a probability distribution facilitates comparing them with the static estimates that were also represented as a probability distribution in Section 5.

Suppose that the inferred route of bag_2 during $[t_3, t_{28}]$ is as given in Table 9. Imagine further that bag_1 (whose inferred route is given in Table 8) and bag_2 are the only bags that are handled during $[t_1, t_{37}] \cup [t_3, t_{28}] = [t_1, t_{37}]$. If one would like to answer $BPMQ^{[t_1,t_7]}$, one follows Algorithm 2 steps extracting $I\text{-}REC([t_1, t_7])$ from the infer-ds to get the records in Table 10, and then proceeding with the calculations given $\eta = 10\%$ to obtain:

$$E^{[t_1,t_7]}_{BP}(\text{MC}) = 4 \times .39 \times .32 \times .9^2 \times 1.1^5 = .6512$$
$$E^{[t_1,t_7]}_{BP}(\text{SMC}) = 4 \times .40 \times .30 \times .14 \times .45 \times .9^2 \times 1.1^5 = .0394$$
$$E^{[t_1,t_7]}_{BP}(\text{TTS}) = 4 \times .21 \times .70 \times .86 \times .23 \times .9^2 \times 1.1^5 = .1517$$
$$E^{[t_1,t_7]}_{BP}(\text{rest of locations}) = 0$$

A final normalization of $E^{[t_1,t_7]}_{BP}$ yields respectively the values:

$$\langle .7731, .0468, .1801, 0 \rangle$$

In this example, MC has the highest dynamic support degree of being a BP in the hall and apron of Aalborg Airport, followed by TTS, and then by SMC. The dynamic support for the rest of the locations is zero, due to the complete absence of inferred records in these locations as seen in Table 10.

Algorithm 2 Answering a BPMQ

Input: $\mathcal{D}_{rfid} = (\mathcal{W}_l, \mathcal{W}_m, c_l, c_m, c_r)$, the infer-ds, a monitoring period T, a probability tweaking parameter η, and a normalization function ψ to $[0, 1]$.
Output: $\psi(E^T_{BP})$.

extract $I\text{-}REC(T)$ from the infer-ds
$increase = 1.0 + \eta/100.0$
$decrease = 1.0 - \eta/100.0$
for each $l \in \mathcal{W}_l$ **do**
  $E^T_{BP}(l) = |\{i\text{-}rec \in I\text{-}REC(T) : l \in i\text{-}rec\}|$
  for each $i\text{-}rec \in I\text{-}REC(T)$ **do**
    $t = i\text{-}rec.e\text{-}time - i\text{-}rec.s\text{-}time$
    if $i\text{-}rec.pr(obj\ at\ l) > 0$ **then**
      $E^T_{BP}(l) = E^T_{BP}(l) \times i\text{-}rec.pr(obj\ at\ l)$
      repeat t times
        $E^T_{BP}(l) = E^T_{BP}(l) \times increase$
    else
      repeat t times
        $E^T_{BP}(l) = E^T_{BP}(l) \times decrease$
return $\psi(E^T_{BP})$

Table 9: The inferred route of bag_2 during $[t_3, t_{28}]$

obj-id	infer-loc	s-time	e-time
bag_2	[MC : 1]	t_3	t_4
bag_2	[MC : .32, SMC : .45, TTS : .23]	t_5	t_7
bag_2	[SMC : .06; TTS : .94]	t_8	t_{10}
bag_2	[CH : .31, CGS : .69]	t_{11}	t_{25}
bag_2	[GS1 : .41; BL1 : .59]	t_{26}	t_{28}

Table 10: $I\text{-}REC([t_1, t_7])$ extracted from the infer-ds

obj-id	infer-loc	s-time	e-time
bag_1	[MC : 1]	t_1	t_2
bag_1	[MC : .39, SMC : .40, TTS : .21]	t_3	t_4
bag_1	[SMC : .30, TTS : .70]	t_5	t_6
bag_1	[SMC : .14, TTS : .86]	t_7	t_8
bag_2	[MC : 1]	t_3	t_4
bag_2	[MC : .32, SMC : .45, TTS : .23]	t_5	t_7

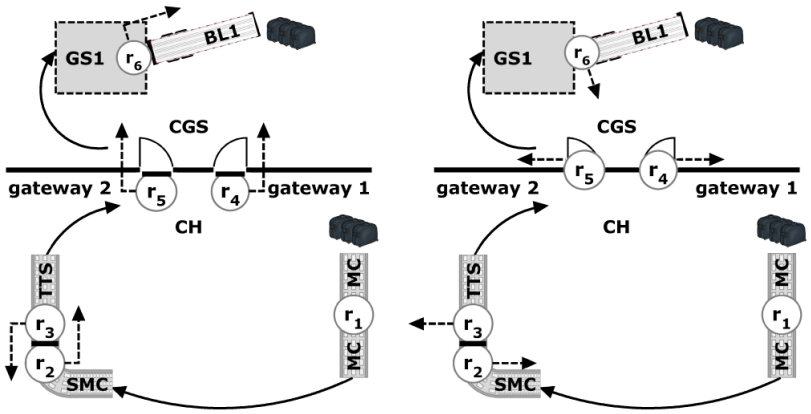
8 Functional Analysis and Experimental Evaluation

This section offers an analysis of the obs function (Formula 2) in addition to two groups of experiments that evaluate the accuracy and performance of algorithms 1 and 2. The experiments are conducted on actual, uncleansed data that is gathered from Aalborg Airport RFID deployment over the period between 2011-08-10 and 2012-09-17. The actual deployment in the hall differs from the one shown in Figure 1 in that a single reader is deployed at (SMC|TTS). Readers deployment in the apron is only planned, therefore RFID readings from the apron are currently unavailable, and the outdoor locations and readers are accordingly excluded from $\mathcal{D}_{rfid}$ (Figure 3). The pre-processing module (mentioned in Section 6.1) reduces the number of raw RFID readings from around 3.3 million down to $845,000$ that are stored in the appear-ds. The overall number of RFID-tagged bags for which these readings are reported is around

$270,000$. All experiments are implemented in Java SE version $1.7.0\_10$ and MATLAB version 7.14. The DBMS used is Oracle 11g Release 2 version $11.2.0.2.0$. The desktop machine, on which the experiments are conducted, has an Intel(R) Core(TM) i7-2600 processor with clock speed 3.40 GHz and 8.00 GB memory, supporting a 64-bit installation of Microsoft Windows 7 Enterprise version $6.1.7601$.

8.1 Analysis of the obs Function

In order to analyze and discern the behavior of the obs function, the baggage route $(\mathrm{MC}\ldots\mathrm{BL1}) \equiv \mathrm{MC} \rightarrow \mathrm{SMC} \rightarrow \mathrm{TTS} \rightarrow \mathrm{CH} \rightarrow \mathrm{CGS} \rightarrow \mathrm{GS1} \rightarrow \mathrm{BL1}$ is chosen (Figure 3) due to the considerable number of readers positioned along it when compared to other routes. The readers positioning and number of locations along this route are varied following three scenarios. In the first scenario (Figure 7a), readers positioning is varied along a line parallel to $(\mathrm{MC}\ldots\mathrm{BL1})$. The initial coverage weights (before the variation starts) are listed in Table 11. This scenario is meant to understand the impact of varying the distribution of coverage weights along a route on this route's observability. In the second scenario (Figure 7b), readers positioning is varied along a line at a right angle to $(\mathrm{MC}\ldots\mathrm{BL1})$. The initial coverage weights (before the variation starts) are listed in Table 12. This scenario is meant to understand the impact of decreasing the coverage weights along a route on this route's observability. In the third and final scenario, the number of locations along $(\mathrm{MC}\ldots\mathrm{BL1})$ is increased as shown in Table 13. Together with this increase, the readers positioning along $(\mathrm{MC}\ldots\mathrm{BL1})$ is varied at a right angle to this route (in a similar fashion to Figure 7b). This scenario is meant to understand the impact of increasing a route length and simultaneously decreasing the coverage weights on this route's observability. In all the aforementioned scenarios, the variation in readers positioning changes (increases/decreases) the coverage weights of the locations along $(\mathrm{MC}\ldots\mathrm{BL1})$ by 5% at a time.



(a) Parallel variation (b) Right-angle variation

Figure 7: Figure (a) shows the variation in readers positioning along a line parallel to the route $(\mathrm{MC}\ldots\mathrm{BL1})$; and Figure (b) shows the variation along a line at a right angle to the same route. Dashed arrows depict the variation, and solid ones depict the route.

Table 11: The initial coverage weights of the locations along $(\mathrm{MC}\ldots\mathrm{BL1})$ in Figure 7a

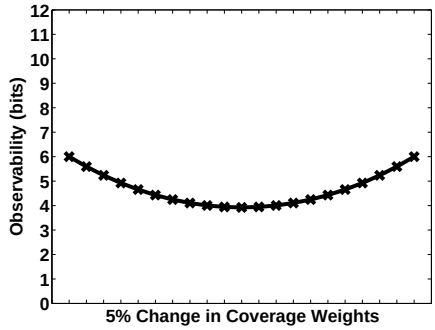
l	$c_r(l) = \{r \rightarrow w(r)\}$	l	$c_r(l) = \{r \rightarrow w(r)\}$
MC	$\{r_1 \rightarrow 1\}$	CGS	$\{r_4 \rightarrow 0, r_5 \rightarrow 0\}$
SMC	$\{r_2 \rightarrow 1, r_3 \rightarrow 0\}$	GS1	$\{r_6 \rightarrow 1\}$
TTS	$\{r_2 \rightarrow 0, r_3 \rightarrow 1\}$	BL1	$\{r_6 \rightarrow 0\}$
CH	$\{r_4 \rightarrow 1, r_5 \rightarrow 1\}$		

Table 12: The initial coverage weights of the locations along $(\mathrm{MC}\ldots\mathrm{BL1})$ in Figure 7b

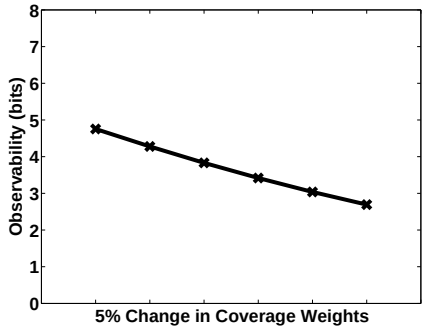
l	$c_r(l) = \{r \to w(r)\}$	l	$c_r(l) = \{r \to w(r)\}$
MC	$\{r_1 \to 1\}$	CGS	$\{r_4 \to .5, r_5 \to .5\}$
SMC	$\{r_2 \to 1, r_3 \to 0\}$	GS1	$\{r_6 \to .5\}$
TTS	$\{r_2 \to 0, r_3 \to 1\}$	BL1	$\{r_6 \to .5\}$
CH	$\{r_4 \to .5, r_5 \to .5\}$		

Table 13: Increasing the number of locations along $(\mathrm{MC}\ldots\mathrm{BL1})$

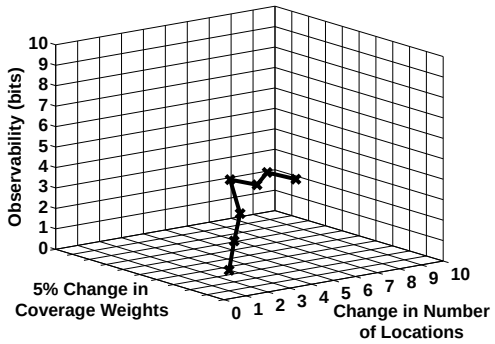
$(\mathrm{MC}\ldots\mathrm{BL1})$
MC
MC $\to$ SMC
MC $\to$ SMC $\to$ TTS
MC $\to$ SMC $\to$ TTS $\to$ CH
MC $\to$ SMC $\to$ TTS $\to$ CH $\to$ CGS
MC $\to$ SMC $\to$ TTS $\to$ CH $\to$ CGS $\to$ GS1
MC $\to$ SMC $\to$ TTS $\to$ CH $\to$ CGS $\to$ GS1 $\to$ BL1



(a) $obs(\mathrm{MC}\ldots\mathrm{BL1})$



(b) $obs(\mathrm{MC}\ldots\mathrm{BL1})$



(c) $obs(\mathrm{MC}\ldots\mathrm{BL1})$

Figure 8: Figures (a), (b), and (c) respectively show the effect of the first, second, and third variation scenarios on $obs(\mathrm{MC}\ldots\mathrm{BL1})$

The impact of the first, second, and third variation scenarios on $obs(\mathrm{MC}\ldots\mathrm{BL1})$ is respectively depicted in Figures 8a, 8b, and 8c. Figure 8a tells that $obs(\mathrm{MC}\ldots\mathrm{BL1})$ is maximized when the coverage weights along $(\mathrm{MC}\ldots\mathrm{BL1})$ are unevenly distributed (i.e., when the reading zones of readers positioned at connection points non-uniformly cover adjacent locations). The short term *uneven positioning* can be used

to refer to this kind of readers positioning in an RFID deployment. The interpretation of the opposite term *even positioning* follows in a similar fashion. Even positioning yields the undesirable impact of minimizing $obs(\mathrm{MC}\ldots\mathrm{BL1})$. Notice again in Figure 8a that $obs(\mathrm{MC}\ldots\mathrm{BL1})$ attains its maximum value (6 bits) under the following uneven positioning:

$$c_r(\mathrm{SMC}) = \{r_2 \rightarrow 1, r_3 \rightarrow 0\}, c_r(\mathrm{TTS}) = \{r_2 \rightarrow 0, r_3 \rightarrow 1\}$$
$$c_r(\mathrm{CH}) = \{r_4 \rightarrow 1, r_5 \rightarrow 1\}, c_r(\mathrm{CGS}) = \{r_4 \rightarrow 0, r_5 \rightarrow 0\}$$
$$c_r(\mathrm{GS1}) = \{r_6 \rightarrow 1\}, c_r(\mathrm{BL1}) = \{r_6 \rightarrow 0\}$$

and the following uneven positioning:

$$c_r(\mathrm{SMC}) = \{r_2 \rightarrow 0, r_3 \rightarrow 1\}, c_r(\mathrm{TTS}) = \{r_2 \rightarrow 1, r_3 \rightarrow 0\}$$
$$c_r(\mathrm{CH}) = \{r_4 \rightarrow 0, r_5 \rightarrow 0\}, c_r(\mathrm{CGS}) = \{r_4 \rightarrow 1, r_5 \rightarrow 1\}$$
$$c_r(\mathrm{GS1}) = \{r_6 \rightarrow 0\}, c_r(\mathrm{BL1}) = \{r_6 \rightarrow 1\}$$

Observe in the values listed above that the full reading zones of r_2 and r_3 are utilized to cover the areas of SMC and TTS, thus leading to maximization of $obs(\mathrm{MC}\ldots\mathrm{BL1})$. On the other hand, $obs(\mathrm{MC}\ldots\mathrm{BL1})$ attains its minimum value (3.9248 bits) in Figure 8a under the following even positioning:

$$c_r(\mathrm{SMC}) = c_r(\mathrm{TTS}) = \{r_2 \rightarrow .5, r_3 \rightarrow .5\}$$
$$c_r(\mathrm{CH}) = c_r(\mathrm{CGS}) = \{r_4 \rightarrow .5, r_5 \rightarrow .5\}$$
$$c_r(\mathrm{GS1}) = c_r(\mathrm{BL1}) = \{r_6 \rightarrow .5\}$$

Observe in the values listed above that half the reading zones of r_2 and r_3 is utilized to cover the areas of SMC and TTS, thus leading to minimization of $obs(\mathrm{MC}\ldots\mathrm{BL1})$. Figure 8b exhibits that $obs(\mathrm{MC}\ldots\mathrm{BL1})$ decreases with the decrease in the coverage weights along $(\mathrm{MC}\ldots\mathrm{BL1})$ (i.e., with the decrease in the overlap between readers reading zones and the areas of locations). This observation is inline with the obs function definition given in Formula 2. Last, the fluctuation in Figure 8c (between the last four obs values 3.9186, 3.5526, 3.7944, and 3.3725) tells that extending $(\mathrm{MC}\ldots\mathrm{BL1})$ (by adding locations observed by readers) does not necessarily increase $obs(\mathrm{MC}\ldots\mathrm{BL1})$ if the readers along $(\mathrm{MC}\ldots\mathrm{BL1})$ are improperly positioned. Put equivalently, the number of readers and their positioning are equally important to achieve a desirable route observability (recall from Section 4 that the number of readers positioned along a route does not accurately reflect this route observability).

8.2 Evaluation of Algorithm 1

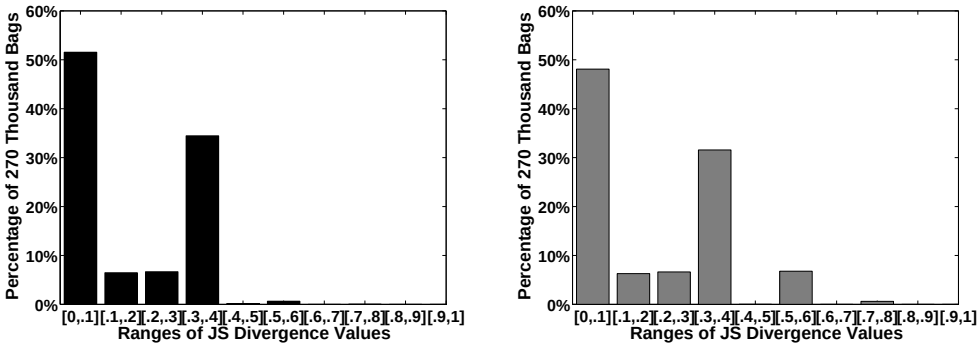
Accuracy: In order to evaluate the accuracy of the translation done by Algorithm 1, one needs to decide how far the translated distribution of baggage is from the real one (from the ground truth). This distance is measured via Jensen-Shannon (JS) divergence measure [15], which is defined by the formula:

$$JS(p_1, p_2) = \sum_{x \in \mathcal{X}} p_1(x) \cdot \log \frac{p_1(x)}{\frac{p_1(x)+p_2(x)}{2}}$$

where p_1 and p_2 are two probability distribution functions on a discrete random variable X whose alphabet is $\mathcal{X}$. JS enjoys a number of salient properties that the commonly-used Kullback-Leibler (KL) divergence lacks [26]. Of these properties are the finiteness and boundness ($0 \leq JS \leq 1$). The reader is referred to [6] for a detailed comparison between these two divergence measures. Another work [7] replaces KL with JS to offer a seminal refinement of an information flow metric. Returning to the accuracy evaluation, the translated distribution of each bag can be easily identified by looking at the infer-ds. The real distribution on the other hand has to be constructed recalling the intuition that reality occurs with certainty, i.e., with a probability of 1. Minding the fairness of this evaluation, synthetic RFID readings are generated for all the $270,000$ bags for which actual data is available. Naturally, the synthetic data is generated under the virtual assumptions of optimal coverage and read rates of RFID readers, and optimal baggage handling in Aalborg

Airport hall. This implies that RFID anomalies are not expected. Moreover, it means that baggage delivered at the check-in desks is properly handled until it is loaded into the designated airplanes. The synthetic data is stored in the synth-ds whose records have the same form as those of the infer-ds. One can then proceed to identify the real distribution of each bag (in the synth-ds) and compute JS divergence between it and the corresponding translated distribution (in the infer-ds). Thus, there is one JS divergence value per bag, i.e., $270,000$ JS values in total. Due to this large number of JS values, the known range $[0, 1]$ of JS is partitioned into 10 smaller ranges, the length of each is 0.1. Then the distribution of the $270,000$ bags across these ranges is reported in Figure 9a. As seen in this figure, the translated distribution is generally no further than $[0, 0.4]$ (99.14% of the bags) from reality, which substantiates the competitive translator accuracy.

Inference Quality: Another aspect of evaluating Algorithm 1 is to judge the quality of the inference carried out in stage 3. For this, one needs to compare between two distances; the distance between the DBN-based infer-ds and real synth-ds distributions, and the distance between the naive prob-ds and real synth-ds distributions. Figure 9a demonstrates the former of these two distances. The latter distance can be similarly determined using JS divergence; it is plotted in Figure 9b. Figure 9a tells that for the DBN-based distribution, only 0.86% of the bags falls outside the range $[0, 0.4]$, as opposed to 7.42% for the naive distribution. Thus, the DBN-based inference of Algorithm 1 improves by around 8.6 times over the naive translation that does not utilize a DBN.



(a) JS for the DBN-based infer-ds (b) JS for the naive prob-ds

Figure 9: Figure (a) [Figure (b)] shows JS between the the DBN-based infer-ds [the naive prob-ds distribution] and real synth-ds distributions.

Performance: In order to evaluate the performance of Algorithm 1, the number of appearance records is varied in the input, and then the time needed by the algorithm to populate the infer-ds is plotted in Figure 10. Clearly, the execution time increases with the increase of the processed records.

8.3 Evaluation of Algorithm 2

First, the translator (Section 6.2) is utilized on the full content of the appear-ds in order to populate the infer-ds, input to Algorithm 2. Then the input parameters to this algorithm are varied as shown in Table 14 (the number of inferred records corresponding to each setting appears within parentheses).

Effect of varying the day: Figures 11a, 11b, and 11c show E_{BP}^{T} for 30 minutes in the morning, afternoon, and evening of the three days listed in Table 14. Notice in these figures that MC, SMC, and TTS have high dynamic support degrees of being BPs in Aalborg Airport hall in most of the day periods. Occasionally, CGS witnesses high dynamic support of being a BP. Looking closely tells that the results obtained answer 9 BPMQs posed in different day periods, and targeting baggage handling quality at Aalborg Airport. Thus, it is noteworthy that the dynamic estimation done constitutes a good model that not only highlights points of

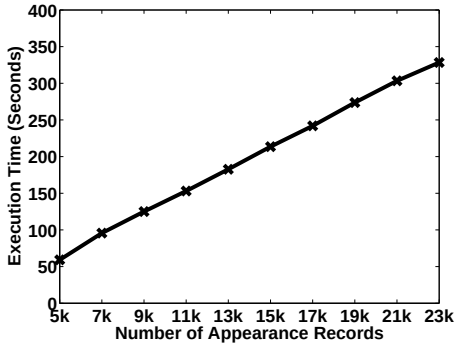


Figure 10: The performance evaluation of Algorithm 1

Table 14: The input parameters to Algorithm 2

Parameter	Values
Day	2012-02-01 (375), 2012-04-15 (164), 2012-06-01 (602)
T in minutes	10 (74), 20 (151), 30 (223), 40 (317)
η	1% (223), 2% (223), 3% (223), 4% (223)

potential suffocation by baggage, but also ranks the standard of each responsibility area in Aalborg Airport. Figure 11d shows that the execution time decreases, for all days, in the order morning, afternoon, and evening, which reflects a corresponding decrease in Aalborg Airport hall traffic. Recall that the execution time is dependent on the number of processed inferred records.

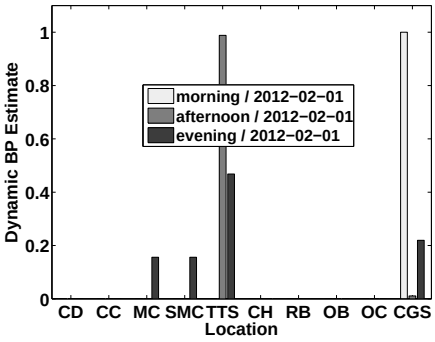
Effect of varying T: Results for varying T for one day, according to the values in Table 14, are shown in Figure 12a. The variation of E_{BP}^{T} in this figure does not follow a noticeable pattern. One can however reason about the histograms by saying that expanding T does not necessarily lead to an increase in the dynamic support for a specific location at the expense of others. Instead, this expansion may introduce support for locations that were not BPs prior to the expansion. Figure 12b tells that the execution time increases proportionally to T, primarily due to the corresponding increase in the number of inferred records that Algorithm 2 processes.

Effect of varying η: The impact of varying η (as prescribed in Table 14) on E_{BP}^{T} and the execution time is shown depicted in Figures 13a and 13b respectively. The slight fluctuation seen in these figures is attributed to the time the Java Virtual Machine needs, and the manner it handles the multiplication and rounding of floating-point numbers.

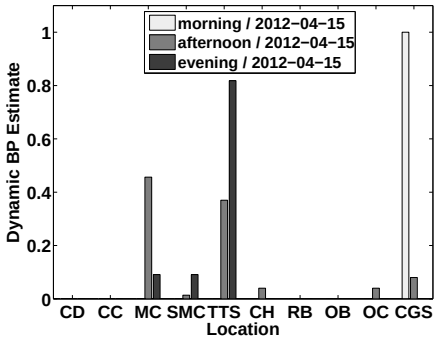
Performance: It is also interesting to study the execution time of Algorithm 2 when varying the number of inferred records in the input. This is shown in Figure 14.

9 Related Work

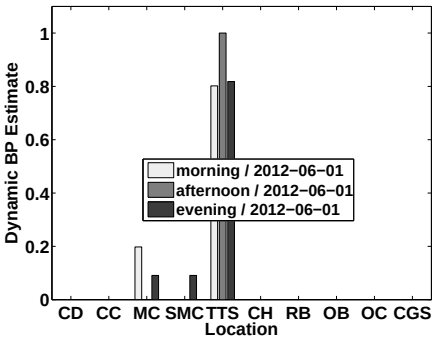
Although it falls into several categories, related work has by far focused on the modeling of indoor spaces. An integrated indoor model [5] covers different information dimensions of indoor models including thematic, geometric, and routing-related information. It is based on classifying indoor objects and structures while taking geometry, appearance, and semantics into account. A lattice-based location model for indoor navigation [13] is capable of preserving semantic relationships and distances, e.g., the nearest neighbor relationship among indoor entities. A grid graph-based model for indoor environments [14] combines the structural properties of these environments with the continuous metric properties that might be of interest to some applications. Another work [22] employs this grid model for evacuation planning. A distance-aware



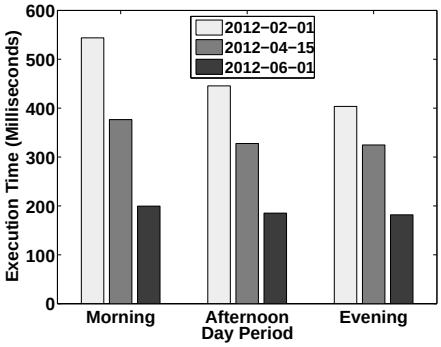
(a) E_{BP}^{T}



(b) E_{BP}^{T}

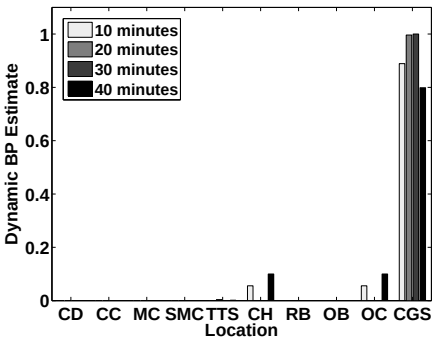


(c) E_{BP}^{T}

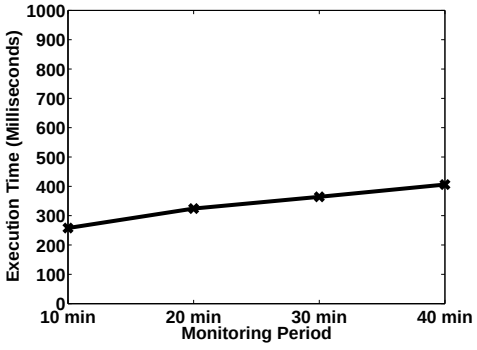


(d) Execution time

Figure 11: Figures (a), (b), and (c) show the effect of varying the day with $T = 30$ minutes, and $\eta = 0.1\%$; and Figure (d) shows the execution time for the former.



(a) E_{BP}^{T}

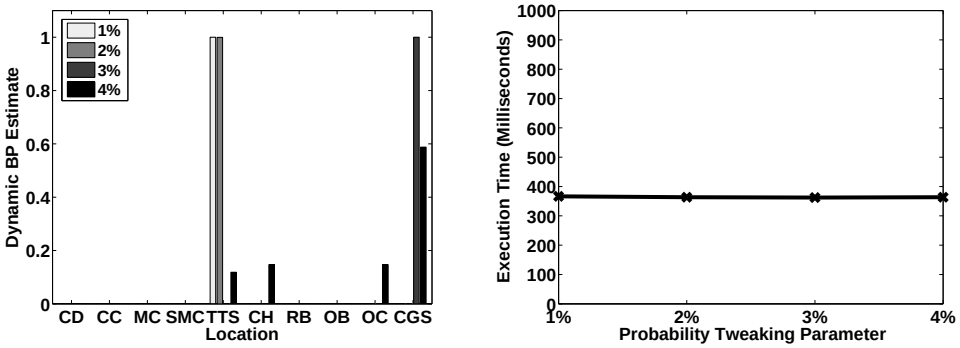


(b) E_{BP}^{T}

Figure 12: Figure (a) shows the effect of varying T with $\mathrm{day} = 2012\text{-}02\text{-}01$, and $\eta = 0.1\%$; and Figure (b) shows the execution time for the former.

indoor space model [16] accompanies a set of indoor distance computation algorithms and an indexing framework in order to enable the processing of indoor queries over indoor objects. This work distinguishes itself from those aforementioned by capturing both O- and I-spaces in a unified model.

A model of built environments [23] uses bigraphs in order to understand the relationships between



(a) E^T_{BP} (b) Execution time

Figure 13: Figure (a) shows the effect of varying η with $\mathrm{day} = 2012\text{-}02\text{-}01$, and $T = 30$ minutes; and Figure (b) shows the execution time for the former.

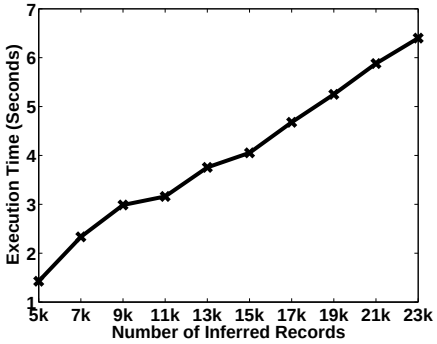


Figure 14: The performance evaluation of Algorithm 2

entities in these environments. The authors' development of inference tools on top of their bigraphs is however ongoing. In contrast, this paper offers a self-contained set of modeling and reasoning techniques for O- and I-spaces. The development of a navigation ontology for outdoor and indoor environments [27] is in progress. This ontology is based on so-called shared microworlds between these two environments. These microworlds are learnt through the Affordance Theory, which enables the identification of functions that entities in outdoor and indoor spaces have or have not in common. This paper's model differs from [27] as follows. First, this paper's model is designed for reasoning about moving objects rather than navigation which is the theme of [27]. Second, this paper's model accommodates receptor deployments which are not considered in [27].

10 Conclusions

This paper reviews and extends a recent unified model of OI-spaces and receptor deployments in these spaces [8]. It shows that the proposed extension enables modelers to express their awareness of various information pieces from the physical world. Based on the extended model, the paper studies the route observability concept, derives its bounded function, and demonstrates its potential for enhancing the reading environment. Furthermore, the paper clearly relates a route observability to the uncertainty in tracking moving objects. The paper defines the notion of a BP. It then performs static reasoning about this notion using the extended model and borrowing from the fundamentals of graph theory. The paper describes a trajectory-

to-route translator that utilizes the extended model. This translator performs probabilistic incorporation of RFID data that compensates for missing information in this data, and enables the search for moving objects in OI-spaces to be optimized. The translated RFID data permits reasoning about BPs in a dynamic, time-dependent fashion. The functional analysis and experimental evaluation (conducted on synthetic and uncleansed, real-world data) validate the proposals made in this paper. In particular, they recognize the behavior of the route observability function, they substantiate the competitive accuracy of the translator and the high quality of the inference, and they demonstrate the sensibleness of the reasoning made about BPs.

Acknowledgment

This work was supported by the BagTrack project sponsored by the Danish National Advanced Technology Foundation under grant 010-2011-1.

References

[1] Y. Bai, F. Wang, and P. Liu. Efficiently filtering rfid data streams. In *CleanDB*, 2006.

[2] P. Bakalov and V. Tsotras. A generic framework for continuous motion pattern query evaluation. In *ICDE*, 2008.

[3] V. Fox, J. Hightower, L. Liao, D. Schulz, and G. Borriello. Bayesian filtering for location estimation. *Pervasive Computing, IEEE*, 2(3):24–33, 2003.

[4] N. Gordon, D. Salmond, and A. Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. *Radar and Signal Processing, IEE Proceedings F*, 140(2):107–113, 1993.

[5] B. Hagedorn, M. Trapp, T. Glander, and J. Döllner. Towards an indoor level-of-detail model for route visualization. In *MDM*, 2009.

[6] S. H. Hussein. A precise information flow measure from imprecise probabilities. In *SERE*, 2012.

[7] S. H. Hussein. Refining a quantitative information flow metric. In *NTMS*, 2012.

[8] S. H. Hussein, H. Lu, and T. B. Pedersen. Towards a unified model of outdoor and indoor spaces. In *ACM SIGSPATIAL GIS*, 2012.

[9] S. Jeffery, G. Alonso, M. Franklin, W. Hong, and J. Widom. A pipelined framework for online cleaning of sensor data streams. In *ICDE*, 2006.

[10] C. S. Jensen, H. Lu, and B. Yang. Graph model based indoor tracking. In *MDM*, 2009.

[11] F. V. Jensen, K. G. Olesen, and S. K. Andersen. An algebra of bayesian belief universes for knowledge-based systems. *Networks*, 20(5):637–659, 1990.

[12] D. Koller and N. Friedman. *Probabilistic graphical models : principles and techniques*. MIT Press, 2009.

[13] D. Li and D. L. Lee. A lattice-based semantic location model for indoor navigation. In *MDM*, 2008.

[14] X. Li, C. Claramunt, and C. Ray. A grid graph-based model for the analysis of 2d indoor spaces. *Computers, Environment and Urban Systems*, 34(6), 2010.

[15] J. Lin. Divergence measures based on the shannon entropy. *Information Theory, IEEE Transactions on*, 37(1), 1991.

[16] H. Lu, X. Cao, and C. S. Jensen. A foundation for efficient indoor distance-aware query processing. In *ICDE*, 2012.

[17] J. Pearl. Fusion, propagation, and structuring in belief networks. *Artif. Intell.*, 29(3):241–288, 1986.

[18] C. Ré, J. Letchner, M. Balazinksa, and D. Suciu. Event queries on correlated probabilistic streams. In *SIGMOD*, 2008.

[19] M. A. Rodriguez and P. Neubauer. Constructions from dots and lines. *Bulletin of ASIS&T*, 36(6), 2010.

[20] S. J. Russell and P. Norvig. *Artificial intelligence : a modern approach*. Pearson Education, 3. ed. edition, 2010.

[21] S. Sathe, H. Jeung, and K. Aberer. Creating probabilistic databases from imprecise time-series data. In *ICDE*, 2011.

[22] J. Sun and X. Li. Indoor evacuation routes planning with a grid graph-based model. In *Geoinformatics*, 2011.

[23] L. Walton and M. Worboys. An algebraic approach to image schemas for geographic space. In *COSIT*, 2009.

[24] E. Welbourne, N. Khoussainova, J. Letchner, Y. Li, M. Balazinska, G. Borriello, and D. Suciu. Cascadia: A system for specifying, detecting, and managing rfid events. In *MobiSys*, 2008.

[25] M. Worboys. Modeling indoor space. In *ISA*, 2011.

[26] J. Xie, J. Yang, Y. Chen, H. Wang, and P. Yu. A sampling-based approach to information recovery. In *ICDE*, 2008.

[27] L. Yang and M. Worboys. A navigation ontology for outdoor-indoor space: (work-in-progress). In *ISA*, 2011.

[28] C. Yuan and M. J. Druzdzel. An importance sampling algorithm based on evidence pre-propagation. In *UAI*, 2003.

A Proofs

A.1 Proof of Theorem 1

The absolute lower bound $obs(R) \geq 0$ is a result of $w(r)$ falling in the range $[0, 1]$. As for the dynamic upper bound, it is noted that for a convex function f and a random variable X, Jensen's inequality gives:

$$Ef(X) \leq f(EX)$$

The function $w(r)\log(w(r)+1)$ is convex (i.e., it lies below any chord), therefore:

$$\begin{aligned}
obs(R) &= \sum_{l\in\mathcal{V}(R)}\sum_{w(r)\in c_r(l)} w(r)\log(w(r)+1) \quad \text{(Formula 2)}\\
&\leq \sum_{l\in\mathcal{V}(R)} \log \sum_{w(r)\in c_r(l)} w(r)(w(r)+1)\\
&\leq \sum_{l\in\mathcal{V}(R)} \log \sum_{w(r)\in c_r(l)} (w(r)+1) \quad (w(r)\in[0,1])\\
&= \sum_{l\in\mathcal{V}(R)} \log \left(\sum_{w(r)\in c_r(l)} w(r) + \sum_{w(r)\in c_r(l)} 1 \right)\\
&= \sum_{l\in\mathcal{V}(R)} \log \left(\sum_{w(r)\in c_r(l)} w(r) + |c_r(l)| \right)\\
&= \sum_{l\in\mathcal{V}(R)} \log \left(\overline{c_r(l)} + |c_r(l)| \right) \quad \text{(Formula 1)}
\end{aligned}$$

This gives the bounds of $obs(R)$:

$$0 \leq obs(R) \leq \sum_{l\in\mathcal{V}(R)} \log \left(\overline{c_r(l)} + |c_r(l)| \right)$$

and proves the theorem.

A.2 Proof of Lemma 1

Suppose that one is commissioned to track an object obj moving along an arbitrary route R in a chosen OI-space monitored by an RFID-based system. The appearance of obj at an arbitrary location along R can be thought of as an event x that occurs with a probability $p(x)$. The information conveyed by this event (self-information) is defined as follows:

$$SI(x) = -\log(x)$$

Intuitively, the higher $obs(R)$, the higher $p(x)$, and so the lower $SI(x)$. In information theory texts, $SI(x)$ also summarizes the uncertainty about the occurrence of x. Therefore, the higher $obs(R)$, the lower the uncertainty about the occurrence of x.

A.3 Proof of Lemma 2

Proving the consistency of E_{BP} as a probability distribution is carried out in two steps. In the first step, E_{BP} is shown to have proper bounds as follows.

$$\begin{aligned}
&\sum_{l\in\mathcal{W}_l} d(l) = 2|\mathcal{W}_m| \quad \text{(Formula 3)}\\
&\sum_{l\in\mathcal{W}_l} \tfrac{d(l)}{2|\mathcal{W}_m|} = 1 \quad \text{(Division properties, } \mathcal{W}_{\mathrm{m}} \text{ is nonempty)}\\
&0 \leq \tfrac{d(l)}{2|\mathcal{W}_m|} \leq 1 : \forall l \in \mathcal{W}_l \quad \text{(Inequality properties)}\\
&E_{BP}(l) \in [0,1] : \forall l \in \mathcal{W}_l \quad \text{(Definition 1)}
\end{aligned}$$

In the second step, it is ensured that no intermediate value of E_{BP} falls outside the range $[0,1]$ by showing that E_{BP} is a monotonically increasing function, i.e., it is shown that:

$$\forall l_1, l_2 \in \mathcal{W}_l : d(l_1) \leq d(l_2) \Rightarrow E_{BP}(l_1) \leq E_{BP}(l_2)$$

as follows:

$$
\begin{aligned}
&d(l_1) \leq d(l_2) \quad \text{(Assumption)} \\
&\tfrac{d(l_1)}{2|\mathcal{W}_m|} \leq \tfrac{d(l)}{2|\mathcal{W}_m|} \quad \text{(Division properties, } \mathcal{W}_{\mathrm{m}} \text{ is nonempty)} \\
&E_{BP}(l_1) \leq E_{BP}(l_2) \quad \text{(Definition 1)}
\end{aligned}
$$

Thus, E_{BP} is invariably consistent as a probability distribution.