ISSN 2288-4866 (Print)
ISSN 2288-4882 (Online)
http://www.jiisonline.org
J Intell Inform Syst 2014 September: 20(3): 1~18 http://dx.doi.org/10.13088/jiis.2014.20.3.001

A Smoothing Data Cleaning based on Adaptive Window Sliding for Intelligent RFID Middleware Systems

DongCheon Shin
Department of Business administration, Chung-Ang University
(dcshin@cau.ac.kr)

Dongok Oh
Department of Culture and Art Management, Chung-Ang University Graduate School
(jirox7@hanmail.net)

SeungWan Ryu
Department of Business administration, Chung-Ang University
(ryu@cau.ac.kr)

Seikwon Park
Department of Business administration, Chung-Ang University
(psk3193@cau.ac.kr)

Over the past years RFID/SN has been an elementary technology in a diversity of applications for the ubiquitous environments, especially for Internet of Things. However, one of obstacles for widespread deployment of RFID technology is the inherent unreliability of the RFID data streams by tag readers. In particular, the problem of false readings such as lost readings and mistaken readings needs to be treated by RFID middleware systems because false readings ultimately degrade the quality of application services due to the dirty data delivered by middleware systems. As a result, for the higher quality of services, an RFID middleware system is responsible for intelligently dealing with false readings for the delivery of clean data to the applications in accordance with the tag reading environment. One of popular techniques used to compensate false readings is a sliding window filter. In a sliding window scheme, it is evident that determining optimal window size intelligently is a nontrivial important task in RFID middleware systems in order to reduce false readings, especially in mobile environments. In this paper, for the purpose of reducing false readings by intelligent window adaption, we propose a new adaptive RFID data cleaning scheme based on window sliding for a single tag. Unlike previous works based on a binomial sampling model, we introduce the weight averaging. Our insight starts from the need to differentiate the past readings and the current readings, since the more recent readings may indicate the more accurate tag transitions. Owing to weight averaging, our scheme is expected to dynamically adapt the window size in an efficient manner even for non-homogeneous reading patterns in mobile environments. In addition, we analyze reading patterns in the window and effects of decreased window so that a more accurate and efficient decision on window adaption can be made. With our scheme, we can expect to obtain the ultimate goal that RFID middleware systems can provide applications with more clean data so that they can ensure high quality of intended services.

주제어 : data cleaning, RFID middleware system, window time slide, tag transition

Received : June 18, 2014 Revised : July 4, 2014 Accepted : July 12, 2014
Type of Submission : Concise Paper Corresponding Author : Seikwon Park

1. Introduction

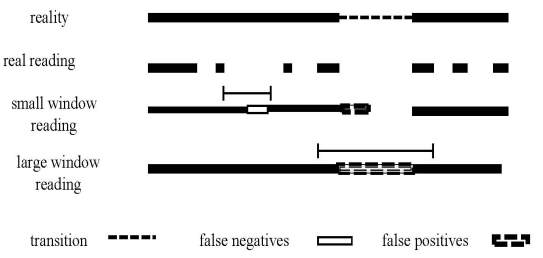
One of the most recent technology trends in computing technology is related to the widespread use of wireless and mobile technologies in many daily life environments. That is, one of the revolutionary trends is the widespread of wireless and mobile computing and communication devices, especially under Internet of Things. Owing to this trend, in the near future, the introduction of tags and sensors is expected to increase substantially in numerous applications. These applications include manufacturing and distribution logistics, pharmaceutical and healthcare, library, traffic and agricultural applications and so on (Ahsan et al., 2010; Mitrokotsa and Douligeris, 2009; Kim and Kim, 2008; Lee and Lee, 2005). From this perspective, there is no doubt about that RFID/SNs (Radio Frequency Identification/Sensor Networks) are common elementary technologies for our every daily life, especially in pervasive environments.

Over the past years, RFID has replaced the traditional barcode systems, incurring a significant change in our lives. RFID is a technology which allows a sensor (reader) to read from a distance, and without line of sight, a unique product identification code (EPC) associated with a tag (Han et al., 2008). With a tag RFID can track a number of items in a cost effective way, generating a large amounts of identification data together with other related data. An RFID system usually consists of 3 components: RF tags, readers with antennas, and host computers. A reader reads tag data attached to objects using RF signals and then data is transmitted to the host computer on which application systems run. Hence, an RFID middleware plays important role in connecting RFID devices (e.g., readers) and applications. Basically, it is responsible for controlling readers and managing data.

RFID data management, due to the nature of seamless and dynamic massiveness, has confronted with many challenges to overcome for the widespread adoption of RFID systems in more applications (Aggarwal and Han, 2013; Arivarasi and Anand, 2013; Derakhshan et al., 2007; Xingyi et al., 2008). Some challenges are as follows: data cleaning, data warehousing, event processing, and privacy. One of the main obstacles to overcome such challenges stems from the inherent inaccuracy by RFID readers; A reader can produce the redundant, unreliable, and missed readings. Data cleaning is associated with such readings in order to maintain data integrity by cleansing RFID data streams and then to delivery we call clean data to applications. Many related works on data cleaning in the middleware level can be founded in the literatures (Bashier et al., 2011; Mahdin and Abawajv, 2011; Chen et al., 2010; Liao et al., 2011, Shen and Zhang, 2008; Xingyi et al., 2008; Wang et al., 2014).

One of the primary issues related to data cleaning is false readings by a reader due to the unreliability of a reader and mobility of tags. The unreliability loses some readings in the reader's detection area. It causes so called false negatives meaning that tags are recognized to be out of reader’s detection range. The tag mobility causes

so called false positives meaning that tags are recognized to be in reader's detection range. It is evident that the incorrect and inaccurate data by false readings have significant ill-effects on the high service quality of applications. Consequently, RFID middleware systems need to be responsible for correcting false readings. Many data cleaning works commonly use a temporal smoothing filter in which false readings from each tag are interpolated within a sliding time window over the reader's data stream. In this approach, the window size is a very critical factor because it leads to tradeoffs between false negatives and false positives. Whereas a smaller size may increase false negatives (missed readings) because of low reading possibility (rate) by a reader, a larger size may increase false positives (mistaken readings) because of high possibility of tag movement. Figure 1 depicts two opposing effects due to the smoothing window size. A small window's readings include false negatives which drop readings of reality. Instead of avoiding false negatives, a large window's readings include more false positives which fail to capture tag transition of reality. Therefore, to set the window size is a crucial, non-trivial task for reducing false readings in the intelligent RFID middleware systems. Some works on data cleaning addressed this problem by adapting the window size automatically and continuously based on observed readings (Jeffery et al., 2008; Massawe et al., 2012).



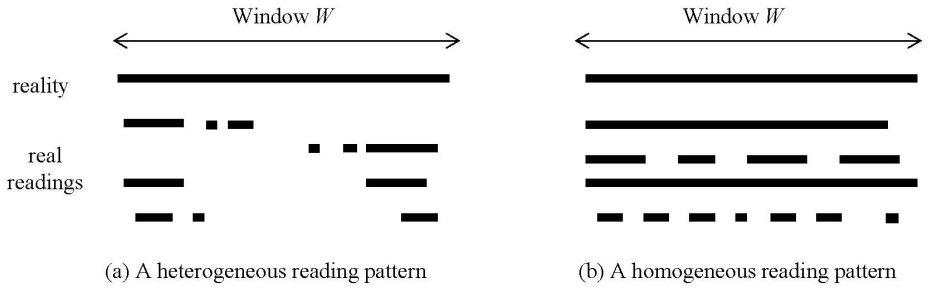
〈Figure 1〉 Effects of the smoothing window size

The first declarative, adaptive smoothing filter for RFID data cleaning, SMURF, has been proposed to compensate for the inherent unreliability of RFID data streams in (Jeffery et al., 2008). SMURF automatically adapts the window size based on the observed tag readings, by modeling RFID data streams as a statistical sample of tags and exploiting sampling theory such as binomial sampling and π-estimators (Lohr, 1999). To distinguish between missed readings and tag transitions, SMURF uses binomial sampling techniques for the single tag cleaning process, and for the multi-tags it additionally uses π-estimators for aggregate cleaning. Another adaptive data cleaning scheme based on some concepts from SMURF called WSTD has been proposed with an improved transition detection mechanism especially for the mobile tag environments (Massawe et al., 2012). In order to detect when transitions occur within a window, WSTD compares the two window sub-range observations or estimated tag counts.

Data cleaning algorithms based on a binomial sampling model such as above, however, may suffer from the inaccurate tag transition detection by modeling the number of successful observations of a tag in the window as a binomial distribution. This implies that averaging based on

the binomial distribution should be a valid estimate of the actual probability of successful observations. It may be true in case that the probability is relatively homogeneous within a given window size. On the contrary, it is not true if tag dynamics are considered because the transition can occur anytime within a time window sliding, especially at the first half of the window. For example, if tag transition occurs in the first half, since the transition time, tag is never observed by a reader to the end of the window. This indicates that averaging based on the binomial distribution is not a valid estimate any more. Figure 2 illustrates two broad categories of reading pattern in a window. In case of homogeneous pattern, the assumption that the reading probability is relatively uniform may be valid. However, it is obvious that the assumption is not valid any more in case of heterogeneous pattern. As a result, a simple averaging can hinder the accurate transition detection and thereby right adaption the of window size. Rather, an inappropriate adaption can generate more false readings, aggravating the problem of false readings. Note that tag transition detection is about to decrease the window size to reduce false positives negatives, which also causes a large number of false negatives accordingly.

In this paper, in order to improve the adaption of window size intelligently for reducing false readings, we propose a new RFID data cleaning scheme for a single tag based on the weight averaging, instead of a simple averaging based on the binomial distribution. The essential insight of our works is to estimate the reading probability (rate) within the window by differentiating the past tag readings and the most recent tag reading. That is, in order to estimate the reading probability of at time *t*, the reading at time *t-1* is separately given a relative weight to the other past ones. With the weight averaging, we can cope with tag transition more accurately by considering the recent tag status to some extent, especially in mobile tag environments. This implies that we can estimate more valid tag reading probabilities within a window. We believe the insight is fairly reasonable intuition in that the more recent reading indicates the more significance in the tag movement within a window. By



(a) A heterogeneous reading pattern (b) A homogeneous reading pattern

〈Figure 2〉 Broad categories of reading pattern

differentiating weights according to the recent degree of readings, our scheme can capture more accurately a tag transition especially in mobile environments, thereby contributing to the efficient and intelligent adaption of window size. For gaining more intelligence in window adaption we further analyze reading patterns in the window and effects of window adaption, which results in a more accurate and efficient decision on window adaption. We believe that our scheme can contribute to the improved quality of application services because RFID middleware systems can provide applications with more accurate data.

The remainder of this paper is as follows. Section 2 introduces a brief review of previous works closely related to our work. Section 3 describes the weight averaging used in our scheme. The RFID data cleaning scheme for the single tag is proposed in Section 4. The conclusions and future works appear in Section 5.

2. Preliminaries

2.1 RFID Basics

RFID technology allows unique identification through an electronic tagging and tracking. Typically, a reader with antennae communicates with tags using RF signals to read IDs. The unique code from the tag is transmitted to one or more readers which then deliver it to one or more servers (applications). Passive RFID tags without an onboard battery are usually powered by the radio signal reading them. Though active tags with an onboard battery provide larger read ranges, passive tags are the most widespread ones in typical large applications such as retail and supply chain management.

To read tag data, a reader interrogates tags in its detection regions by sending RF signal. The responding tags to the signal are to be identified in the corresponding interrogation cycle to attempt to read tags in the reader's region. A reader can read multiple tags simultaneously. An epoch, which is a unit of reading time, may compromise a number of interrogation cycles. For each epoch, the reader maintains a tag list containing tag IDs and other necessary information such as the number of interrogation responses and the last time to be read. For more information on RFID technology, see (Want 2004).

As you know, unfortunately, the unreliability of RFID readings is one of the primary causes to hinder the widespread use of RFID technology. To deal with the unreliability caused by false readings, in temporal sliding window smoothing filters, a proper window size is of importance for ensuring completeness of readings and tag dynamics. A reader transfers tag information to the servers every time interval of window size. In fact, completeness and tag dynamics are two opposing properties which need to be carefully balanced. A large window size is good for ensuring completeness, but is not good for tag dynamics. In comparison, a small window size is good for tag dynamics, but is not good for ensuring completeness. Therefore, adapting the window size automatically can be one of good directions.

2.2 A Brief Reviews of Previous Statistical Modeling for Adaptive Cleaning

It is well known that the raw RFID data streams typically do not represent a precise population of tags in the physical world. That is, a significant fraction of tag readings is missed due to some obstacles in detection region or tag dynamics causing to be out of detection region. A statistical modeling views this observed readings as a random sample of the population of tags in the physical world. Therefore, readings observed in each epoch are viewed as a random sampling trials and then readings observed in the smoothing window is viewed as repeated random sampling trials. Based on this insight, a statistical sampling theory is exploited to model the corresponding physical world.

An epoch is an atomic time unit of reading and multiple epochs constitute a smoothing time window which is a sequence of consecutive epochs. The time window is slid by one epoch. Furthermore, the slide point is set to the middle of the window. In other words, after the entire window has been read, reading corresponding to the midpoint of the window is produced. Figure 3 depicts 3 scenarios as a result of adapting the window size. The adjustment of window size uses the common Addictive-Increase/Multiplicative-Decrease (AIMD) paradigm (Chui and Jain, 1989). That is, the current window size is increased additively and is decreased multiplicatively.

Let N_t denote the unknown size of the underlying tag population at each epoch t, and let $S_t \subseteq \{1,2,\cdots, N_t\}$ denote the subset of tags observed(i.e., sampled) during that epoch. S_t is viewed as an unequal probability random sample of the tag population. A per-epoch sampling probability $p_{i,t}$ for each tag i at epoch t is derived as in (1). It is obvious that $p_{i,t}$ differs across tags time.

$$p_{i,t} = N_{response} / N_{inteerogation} \quad (1)$$

$N_{response}$ and $N_{interrogation}$ are number of readings(responses) and the known number of interrogations(requests) for tag i at epoch t, respectively. Each epoch is viewed as an independent Bernoulli trial (i.e., a sampling draw

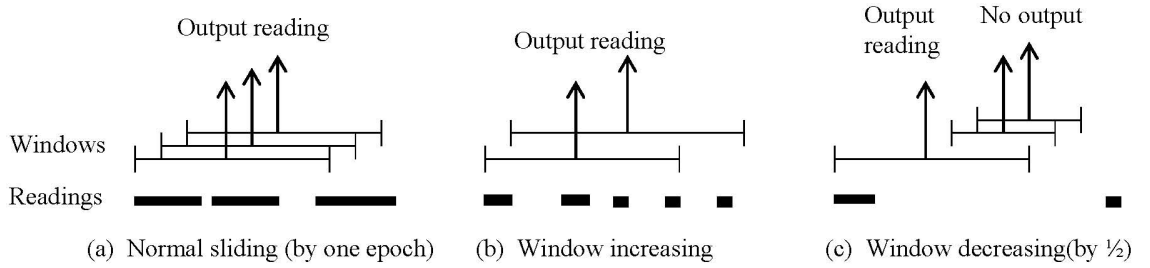


Figure 3 Example scenarios in adaption of window size

for tag i) with successful probability p_i. This implies that the number of successful observations of tag i in the window W_i of size w_i epochs (i.e., $W_i = (t - w_i, t)$) is a random variable with a binomial distribution $B(w_i, p_i)$. The most important assumption is that within an appropriate window size the s will be relatively homogeneous and thus averaging is a valid estimate of the actual . The average empirical read rate in the window W_i is derived as in (2).

$$P_i^{avg} = \sum_{t \in S_i} p_{i,t} / |S_i| \qquad (2)$$

S_i is a subset of all epochs in W_i ($S_i \subseteq W_i$). Viewing S_i as a binomial sample of epochs in W_i and $|S_i|$ as a $B(w_i, P_i^{avg})$ binomial random variable, the expected value and variance of $|S_i|$ is given as in (3).

$$\mathrm{E}[|S_i|] = w_i \cdot p_i^{avg}, \quad \mathrm{Var}[|S_i|] = w_i \cdot p_i^{avg} \cdot (1 - p_i^{avg}) \qquad (3)$$

The binomial sampling model makes it possible to adjust window size w_i aiming at complete readings within a window W_i. Setting the number of epochs within the smoothing window as in (4) ensures that tag i is observed within W_i with probability > 1 - δ

$$w_i \geq \ln(1/\delta) / p_i^{avg} \qquad (4)$$

Basically, the window size is increased if the estimated window size using (4) is greater than the current window size. The less current size implies that enlarging the window size is likely to increase the possibility of tag detection, resulting in decreasing the false negatives.

On the other hand, to avoid false positives due to tag transitions, it needs to be capable of accurately determining when a tag is out of the detection range within a window. This transition detection also employs the binomial sampling model as statistically significant deviations in the observed binomial sample size from its expected value. More formally, assuming that the current window size w_i and sampling probability are not too small, it follows from a Central Limit Theorem (CLT) argument that the value of $|S_i|$ is within ±2 $\sqrt{Var[|S_i|]}$ of its expectation with probability close to 0.98 with the assumption that no transition occurs in the current window. As a result, it can be said that tag transition is detected if the two conditions in (5) hold. That is, the number of observed readings is less than the expected number of readings and statistically a significant variation in the tag observations.

$$(|S_i| - w_i \cdot p_i^{avg}) < 0 \quad \text{and}$$
$$\left||S_i| - w_i \cdot p_i^{avg}\right| > 2\sqrt{w_i \cdot p_i^{avg} \cdot (1 - p_i^{avg})} \qquad (5)$$

Once a tag transition is detected, the window size is decreased to give more opportunities for reducing the false positives.

According to the common Addictive-Increase/ Multiplicative-Decrease (AIMD) paradigm, the current window size is increased additively and is decreased multiplicatively.

To alleviate the problems of mobile tags, a simple heuristic to detect a mobile tag can be devised. If the tag is detected with consistently falling $P_{i,t}$ within the window, it is likely to moving away from the reader and then may be exiting the detection range soon. For the purpose of determining the tag movement away from the detection region, the slope of the best-fit line using the least squares can be employed. The negative slope (in unit of $\frac{\Delta P_{i\ t}}{\text{epochs}}$) of the best-fit line indicates the tag movement. Such readings can be filtered before the main processing. In addition, based on guidance from the binomial sampling model reviewed here, several concrete algorithms can be devised by establishing either a mechanism for size adjustment and/or conditions to be held in order to reduce false readings (either negatives or positives).

3. Weight Averaging

To guarantee both the completeness for reducing false negatives and the tag dynamics for reducing false positives, setting the appropriate window size is of importance. However, this problem is inherently not trivial one because of the opposing effects in relation to the window size. In this respect, the adaptive approaches based on the statistically binomial sample model taken in (Jeffery et al., 2008; Massawe et al., 2012) can be considered as good attempts. In those approaches, the accurate calculation of p_i^{avg} is a critical task in order to attain the intended goal of effectively reducing false readings. In order for p_i^{avg} to be reliable and reasonable, the assumption that the probability of successful tag reading is uniform over each epoch within the window must be valid. Unfortunately, in real environments, either tag movements or other reasons such as presence of some materials hindering detection threat this assumption because it is inevitable for them to cause the unequal distribution of reading probability. This implies more false readings can be generated due to the inaccurate decision on the window adaption.

The unequal distribution means that reading probability in each epoch within a window is different from each other. To reflect this observation, we employ so called the weight averaging with which separately different weights are given to the recent reading probability in order to calculate the average reading probability in each epoch within the window. By giving separate weights to the recent epochs rather than former ones in the window, we can capture more meaningful tag movement because the more recent reading reflects the more recent tag position. A simple averaging used in previous works have the possibility of capturing less exact tag transition because it equivalently treats the old readings and the recent

readings. With the weight averaging we can properly deal with heterogeneity of reading probabilities in the window. As a result, we expect the approach using weight averaging instead of binomial sampling model can detect tag transitions more accurately, reducing false readings accordingly.

The weight averaging estimates the average reading probability of each epoch within the window *W* as in (6), giving a separate weight on the recent reading probability.

$$\varphi_t = \alpha \cdot \varphi_{t-1} + (1-\alpha) \cdot p_{t-1} \quad (0 \le \alpha \le 1,\ 1 \le t \le w) \quad (6)$$

$\varphi_w$and p_t denote the estimated average reading probability and the actual reading probability at time *t* (i.e., at epoch *t*), respectively. The window of size *w* consists of *w* epochs. Note that p_t is $N_{response}$ / $N_{inteerogation}$ at epoch *t* as in (1). The constant *a* is a weighting factor. If *a* = 0, the next estimated probability is the same as the current actual one. This implies that the estimated reading probability at each epoch is entirely dependent on the current actual reading probability. If *a* =1, the next estimated reading probability at each epoch is same as the current one. This implies that the estimated reading probability is assumed be uniform without separate consideration of the recent readings. By establishing the value of *a* suitable for the tag environments (e.g., 0.8), we can devise various strategies with a different favor of the reading recentness

The estimation of average uses the previous reading probabilities. Without a generality, the initial values of φ_0 and p_0 can be set as the estimated reading probability and the actual reading probability of the last epoch of the just before window, respectively. From (6) we can easily know that φ_w of the last epoch *w* is calculated recursively as shown in $\widetilde{(6)}$.

$$\varphi_w = a^w + \varphi_0 + a^{w-1} \cdot (1-a) \cdot p_0 + \cdots a^i \cdot (1-a) \cdot p_{w-i-1} \cdot + a^2(1-a) \cdot p_{w-3} + a \cdot (1-a) \cdot p_{w-2} + (1-a) \cdot p_{w-1} \quad \widetilde{(6)}$$

Since both *a* and 1-*a* is less than or equal to 1, each successive term has larger than its predecessor. This implies the more recent reading probability is given to more weight in calculation of estimated reading probability of each epoch in the window. As a result, we can detect tag transition more accurately by treating the recent readings separately. Once the estimated reading probability of each epoch in the window is calculated from (6), the estimated average reading probability, φ_t^{avg}, of the window W_i of size w_i for the tag *i* is calculated as in (7).

$$\varphi_t^{avg} = \Sigma_{t \in wi} \varphi_t / w_i \quad (7)$$

4. A Single RFID Data Cleaning

As described earlier, to reduce false readings (both false negatives and false positives) due to

unreliability of readings and tag mobility is of importance in data cleaning based on window time slide. It is obvious that false readings degrade the quality of application services due to the incorrect data. In this section, for the purpose of reducing false readings more intelligently, we present an adaptive smoothing filter for a single tag data cleaning, WASWIN (Weight Averaging Smoothing WINdow for rfid data), based on the weight averaging, not following a binomial distribution. We adopt some concepts of statistical modeling used in previous related works such as SMURF (Jeffery et al., 2008) and WSTD (Massawe et al., 2012). However, WASWIN fundamentally has a distinguished feature of weighting on the more recent reading probability at each epoch in the window. This feature basically enables for WASWIN to detect tag transition more accurately, especially in heterogeneous (non-uniform) distribution of readings. The accurate tag transition detection is a very important issue in adapting window for reducing false readings, especially in mobile environments. That is, to differentiate between missed readings and tag transition is the most essential task. For example, in a warehousing environment some tagged objects may be moved while others are static. Hence, it is evident that the accurate detection of tag transition significantly affects the performance of adaptive window sliding mechanism.

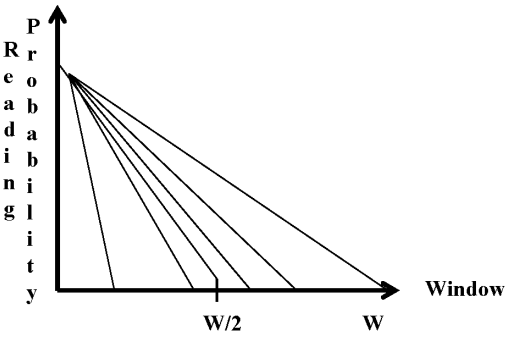
WASWIN sets to one epoch as an initial window size for each tag. The window size is dynamically adjusted on basis of the actual readings. During each new epoch, for each tag *i* WASWIN performs window processing (***Window Processing(W$_i$)***). This process includes estimation of several parameters such as $|S_i|$ and φ_t^{avg} using tag-list information by observed reading of tag i. Of course, if $|S_i|>0$, which means that there exists at least one reading within the window, the reading of tag *i* is identified (observed). In addition, it also includes emitting an output reading for the identified tag i. Then, WASWIN tries to determine the window size for completeness with high probability (***FindComplete Size(φ_i^{avg}, δ)***). According to (4), a window size of $w_i^* = \ln(1/\delta)p_i^{avg}$ for tag *i* is sufficient to guarantee the correct reading of tag with high probability 1-δ. In our work, for tag *i* φ_i^{avg} is used to calculate w_i^* on behalf of p_i^{avg} as in (8).

$$w_i^* = ln(1/\delta)/\varphi_i^{avg} \quad (8)$$

If the current window size w_i is smaller than the estimated window size w_i^*, it needs to increase the current window size in order to increase the possibility of tag readings in the larger window. Of course, together with the above condition, a preceding condition to increase the window size is that the number of actually observed readings is less than the expected number of readings (($|S_t|$ - $W_i \cdot \varphi_i^{avg}$) < 0). Otherwise, since it indicates that the current window size is enough to maintain the reading performance, there is no need to increase the window size.

On the other hand, as a window size is

increased the possibility of false positives also increases because a tag *i* is presumed to be present in the reader's detection region in spite of tag transition due to the interpolation of readings in the window. Consequently, it is very crucial to accurately detect the time when a tag transition occurs and then to decrease the window size to reduce false positives. To detect transitions, as a first step, WASWIN also uses the slope of the best-f using the least squares (***NegativeSlopeOfLine*** $(P_{i,t})$**).** If the tag is detected with consistently falling reading probability within the window, it is likely to leave from the reader in the near future. That is, we can decide that the negative slope (in unit of $\frac{\triangle \Pi, t}{epochs}$) of the best-fit line implies the tag transition soon. Figure 4 shows some possible negative slopes. The reading probability is declined drastically or gradually. Every negative line indicates the tag transition. Though slope is negative, within the window some tags may be present in the reading region. This implies that there needs to see statistically-significant deviations in the actual sample size from its expected value.

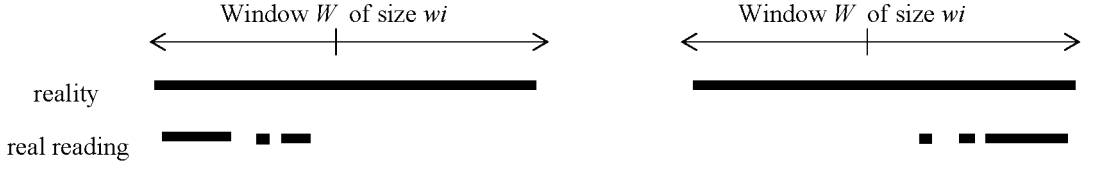


〈Figure 4〉 Negative slopes of the best-fit line

To see the significant deviations, WASWIN tries to determine whether the conditions from CLT(Central Limit Theorem) shown in (9) hold (***SeeSinificantDeviation(|Si|, Wi,*** φ_i^{avg}***)***).

$$(|S_i| - w_i \cdot \varphi_i^{avg}) < 0 \text{ and}$$
$$||S_i| - w_i \cdot \varphi_i^{avg}| > 2\sqrt{w_i \cdot \varphi_i^{avg} \cdot (1-\varphi_i^{avg})} \quad (9)$$

If the condition in (9) holds, to decide tag transition more accurately, WASWIN attempts to analyze reading patterns in the window in relation to the significant deviations. There are broadly 2 categories of reading pattern to satisfy condition in (9): biased pattern and non-biased pattern over the



(a) A heavily biased pattern towards 1st half (b) A heavily biased pattern towards 2nd half

〈Figure 5〉 Illustrative heavily biased cases of reading pattern

window. Therefore, to see the effects of size decrease due to transition, it is proper to investigate biased pattern before the decision about the transition. Figure 5 depicts two heavily biased cases of reading patterns in the window.

If we carefully inspect the real reading patterns shown in Figure 5, we can suppose that in both cases the estimated average reading probability over the window *W* have almost similar value because the real reading is similar. Therefore, in making a decision about window adaption, they lead to the same decision. Let W = (t-w_i, t] be the current window of size w_i. In the biased case of a), with the current window size w_i, many false positives (at most approaching to about w_i/2) are produced if tag transition actually has occurred immediately after time t- w_i/2. In this case, if the window size is decreased, the number of false positives is reduced (at most approaching to about 0, if window size is decreased to w_i/2 without affecting the number of false negative). As a result, it is desirable to determine that there exists a tag transition, thereby decreasing the window size (e.g., w_i/2, a new window W^ = (t-w_i/2, t]). In contrast, in case of b), if the window size is decreased to the size of w_i/2, many false negatives (at most approaching to about w_i / 2^2) are produced. That is, the possibility of false negatives may reversely increase because of the decreased window size. From the above observation, it is desirable to decide the occurrence of tag transition only in case of (a). Therefore, we come to the conclusion that it needs to differentiate the above two biased cases.

From Figure 5, we can know that there exist statistically significant changes in estimation of tag reading over sub-ranges of the current window. Therefore, we can identify reading patterns in Figure 5 by considering the contribution of sub-range (e.g., (t - w_i/2, t]) to the expectation value E[$|S_i|$]. This is possible because the higher reading probability gets to contribute to the expectation more. If reading pattern is homogeneous, the expectation value also deserves to show relatively uniform distribution over the window. Therefore, the expectation value of sub-range (e.g., (t – w_i/2, t]) deserves to be proportional to the size of the sub-range. Let W^+ = (t - w_i/2, t] be the 2nd half of the window W of size w_i. From CLT argument we get the expression shown in (10). $|S_i^+|$ is the actual number of readings in W+ (i.e., the 2nd half of the window W_i) for tag i.

$$|S_i^+| \in E[|S_i^+|] \pm 2\sqrt{Var|S_i^+|} \qquad (10)$$

Based on the above observation, WASWIN tries to identify reading patterns for enhancement to tag transition detection by determining whether the condition shown in (11) hold (***BiasedReadingPattern***($|S_i^+|$, w_i φ_i^{avg})). Satisfying the condition in (11) implies that a transition is more likely to have occurred in the 2nd half of the window in terms of statistically significant change. Hence, reading pattern of window can be classified as heavily biased pattern towards 1st half similar to Figure 5-a). As a result, we can make a decision

about transition detection more accurately.

$$||S_i^+| - \frac{1}{2} \cdot w_i \cdot \varphi_i^{avg}| > 2\sqrt{\frac{1}{2} \cdot w_i \cdot \varphi_i^{avg} \cdot (1 - \varphi_i^{avg})} \quad (11)$$

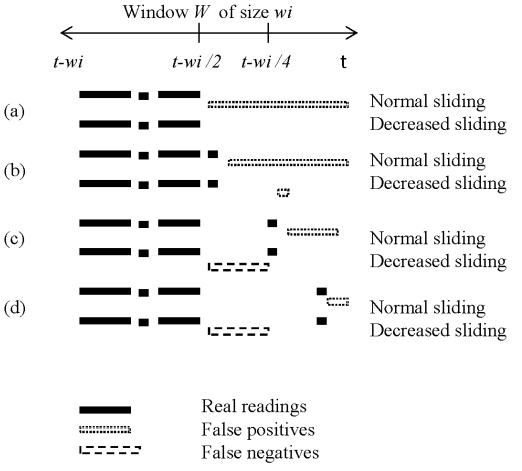


Figure 6 Example cases in 2nd half of the window

If the window size is decreased due to the pattern similar to Figure 5-a), the 2nd half of the window W needs to be investigated again to examine the effects of the decreased window size. Consider Figure 6 which shows 4 representative cases of the 2nd half. With normal sliding, in case of Figure 6-a) and 6-b) the number of false positives is increased approximately to $w_i/2$. In contrast, if the window size is decreased (e.g., $w_i/2$), we can gain the decreased false positives approaching to 0 at most. In case of Figure 6-c), the normal sliding increases the false positives approaching to $w_i/2^2$ at least, while the false negatives due to the decreased window size are also increased to approximately to $w_i/2^2$ at most. However, the situation becomes different in case of Figure 6-d). Whereas the normal sliding increases the false positives approaching to $w_i/2^2$ at most, the false negatives due to the decreased window size are also increased to approximately to $w_i/2^2$ at least. From these observations, except patterns similar to Figure 6-d), we can gain more benefits due to the decreased window size. As a result, if $|S_i^{2+}| > 0$ which means the reading patterns are similar to Figure 6-d) holds, it is undesirable to decrease the window size. $|S_i^{2+}|$ denotes the actual number of readings in W2+ = $(t - w_i/2^2, t]$ for tag i.

The following **Algorithm : WASWIN** is a pseudo-code description of WASWIN's adaptive single tag cleaning algorithm. Each tag is cleaned in its own window. Like similar previous works, WASWIN also adjusts the window size according to the common Additive-Increase/Multiplicative-Decrease paradigm. Since the main goal of WASWIN is the capture of transition, window decreasing is checked before than window increasing.

Algorithm : WASWIN

Input: T = set of all observed tag IDs
δ = required completeness confidence
Output: t = set of all identified tag IDs to be reported
$\forall i \in T,\ w_i \leftarrow 1$
While (*GetNextEpoch()*) **do**
for ($i \in T$) **do**
if ($|S_i| > 0$)

WindowProcessing(W_i) ;
$w_i^* \leftarrow$ *FindCompleteSize*(φ_i^{avg}, δ) ; //find estimated window size
if (*NegativeSlopeOfLine* **and** *SeeSinificant Deviation* ($|S_i|$, w_i, φ_i^{avg}))
if (*BiasedReadingPattern* ($|S_i^+|$, w_i, φ_i^{avg})) **and** ($|S_i^{2+}| < 0$)) //
$w_i \leftarrow \max\{\min\{{w_i}/{2}, w_i^*\}, 1\}$; //decrease size if proper conditions are satisfied
end if
else if ($w_i^* > w_i$) //increase window size if estimated size is larger than current one
$w_i \leftarrow \max\{\min\{w_i + 2, w_i^*\}, 1\}$
end if
end if
end if
end for
end while

Table 1 shows the brief comparison with primarily related works, SMURF (Jeffery et al., 2008) and WSTD (Massawe et. Al., 2012). As you can see, WASWIN has no assumption about the reading pattern by a reader, which implies that WASWIN can appropriately cope with unreliability of the reader and tag dynamics (transitions) compared to SMURF and WSTD. This is because binomial distribution cannot be proper any more in the heterogeneous reading patterns. In addition, in order to more carefully make a decision on the window size adaption, WASWIN requires to satisfy additional meaningful conditions so that it can make an accurate and efficient decision.

〈Table 1〉 A brief comparison with primary related works

		SMURF (Jeffery et al., 2008)	WSTD (Massawe et. Al., 2012)	WASWIN (Ours)
Assumption (reading pattern)		Relatively homogeneous	Relatively homogeneous	None
Base		Binominal distribution	Binominal distribution	Average weighting
Minimum window size		1	3	1
Increasing window size	Conditions	$w_i^* > w_i$	$w_i^* > w_i$ AND $\lvert S_i\rvert > w_i\ \varphi_i^{avg}$	$w_i^* > w_i$
	size	2	2	2
Decreasing window size	Conditions	Negative slope AND statistically significant deviation	Negative slope AND |S2i|=0	Negative slope AND statistically significant deviation AND biased reading patterns AND positive effects of decreasing size)
			statistically significant deviation	
	size	${w_i}/{2}$	${w_i}/{2}$	${w_i}/{2}$
			-2	

$|S_{2i}|$: observed size of second half of the window Wi

5. Conclusions and Further Works

In this paper, we have proposed the adaptive RFID data cleaning scheme, WASWIN, to reduce the false readings more intelligently so that applications can ultimately provide higher quality of services with clean data delivered by RFID middleware systems. WASWIN employs the weight averaging to actively cope with non-uniform distribution of successful tag readings frequently occurred in real environments. Owing to weight averaging, we expect WASWIN can make more accurate and efficient decision on the adaption of window size dynamically even in heterogeneous distribution of successful tag readings in a window. In addition, to reach a more accurate decision, WASWIN detects the heavily biased reading pattern by analyzing the reading patterns in the window. Finally, by investigating the tradeoff between false positives and false negatives due to adaption of window size, WASWIN comes to the decision on the adaption carefully. Our improved window adaption scheme can be expected to contribute to a wide range of industrial fields introducing RFID/SN technology including manufacturing systems, distribution logistics, factory automation, supply chain management as well as embedded system applications.

The subsequent ongoing work is planned to verify the novel aspects of our RFID data cleaning scheme through experiments under various environments. This involves the effect analysis of several weight factors on the efficiency of the scheme. The next is to expand our scheme to the multi-tag data cleaning. Currently, our ongoing work includes the approach to remove redundant data when there are overlaps in the reading areas of multiple readers especially in case of tag movements.

References

Aggarwal, C. C. and J. Han, *A Survey of RFID Data Processing*, In *Managing and Mining Sensor Data*, C. C. Aggarwal (ed.), Springer, New York, 2013, 349~382.

Ahsan, K., H. Shah and P. Kingstone, "RFID Applications: An Introductory and Exploratory Study," *International Journal of Computer Science Issues*, Vol.7, No.3(2010), 1~7.

Arivarasi, S. and S. K. Anand, "A Detailed Survey on Various Tracking Methods Using RFID," *International Journal of Engineering and Technology*, Vol.5, No.2(2013), 900~904.

Bashier, A. K., S.-J. Lim, C. S. Hussain and M.-S. Park, "Energy Efficient In-network RFID Data Filtering Scheme in Wireless Sensor Networks," *Sensors*, Vol.11, No.7(2011), 7004~7021.

Chen, H., W.-S. Ku, H. Wang, and M.-T. Sun, "Leveraging Spatio-Temporal Redundancy for RFID Data Cleansing," *Proceedings of the 2010 ACM SIGMOD International Conference on Management of data*, (2010), 51~62.

Chui, D.-M. and R. Jain, "Analysis of the Increase and Decrease Algorithms for Congestion Avoidance in Computer Networks," *Computer Networks and ISDN Systems*, Vol.17, No.1 (1989), 1~14.

Derakhshan, R., M. E. Orlowska and X. Li, "RFID Data management: Challenges and Opportunities," *Proceedings of IEEE International Conference on RFID*, (2007), 175~182.

Han, J., J.-G. Lee, H. Gonzalez, and X. Li, "Mining Massive RFID, Trajectory, and Traffic Data sets(tutorial), *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, (2008).

Jeffery, S. R., M. Garofalakis, and M. J. Franklin, "Adaptive Cleaning for RFID Data Streams," *Proceedings of the 32nd international conference on Very large data bases*, (2006), 163~174.

Kim, J. K. and H. K. Kim, "Agricultural and Stockbreeding Products Recommender System Using RFID Based Traceability System," *Journal of Intelligence and Information Systems*, Vol.14, No.2(2008), 207~222.

Lee, J. W. and Y.-K. Lee, "Distributed RFID Information Service Architecture for Ubiquitous Logistics," *Journal of Intelligence and Information Systems*, Vol.11, No.2(2005), 105~121.

Liao, G., J. Li, L. Chen, C. Wan, "KLEAP: An Efficient Cleaning Method to Remove Cross-Reads in RFID Data Streams," *Proceedings of the 20th ACM international conference on Information and knowledge management*, (2011), 2209~2212.

Lohr, S. L., *Sampling: Design and Analysis*, Duxbury Press, New York, 1999.

Mahdin, H. and J. Abawajy, "An Approach for Removing Redundant Data from RFID Data Streams," *Sensors*, Vol.11(2011), 9863~9877.

Massawe, L. V., H. Vernaak, and J. D. M. Kinyua, "An Adaptive Data Cleaning Scheme for Reducing False Negative Reads in RFID Data Streams," *Proceedings of IEEE International Conference on RFID*, (2012), 157~164.

Mitrokotsa, A. and C. Douligeris, *Integrated RFID and Sensor Networks: Architecture and Applications*, In *RFID and Sensor Networks*, Y. Zhang, L.T. Yang, J. Chen (eds.), CRC press, 2009, 511~535.

Shen, H. and Y. Zhang, "Improved Approximate Detection of Duplicates for Data Streams Over Sliding Windows," *Journal of Computer Science and Technology*, Vol.23, No.6(2008), 973~987.

Wang, L., L. D. Xu, Z. Bi, Y. Xu, "Data Cleaning for RFID and WSN Integration," *IEEE Transactions on Industrial Informatics*, Vol.10, No.1(2014), 408~418.

Want, R., "The Magic of RFID," *Queue*, Vol.2, No.7(2004), 40~48.

Jin, X., X. Lee, N. Kong, and B. Yan, "Efficient Complex Event Processing over RFID Data Stream," *Proceedings of the Seventh IEEE/ ACIS International Conference on Computer and Information Science*, (2008), 75~81.

국문요약

지능적인 RFID 미들웨어 시스템을 위한 적응형 윈도우 슬라이딩 기반의 유연한 데이터 정제

신동천* · 오동옥** · 류승완* · 박세권***

RFID는 유비쿼터스 환경의 다양한 응용분야에서 기본적인 기술로 사용되어 왔다. 특히, 사물 인터넷을 위한 향후 RFID 기술의 폭 넓은 활용의 장애물중의 하나는 태그 리더기에 의한 RFID 데이터의 근본적인 비 신뢰성이다. 특히, 읽기 손실과 잘못된 읽기 같은 읽기오류 문제는 RFID 시스템이 적절히 처리해야 할 필요가 있다. 왜냐하면, 미들웨어 시스템이 전달한 오류 데이터는 궁극적으로 응용 서비스의 품질을 저하시킬 수 있기 때문이다. 따라서 높은 품질의 서비스를 위해서 지능형 RFID 미들웨어 시스템은 응용에 깨끗한 데이터를 전달하기 위해 읽기오류를 상황에 따라 적절하게 처리하여야 한다. 읽기 오류를 해결하기 위한 보편적인 방법 중의 하나는 슬라이딩 윈도우 필터의 사용이다. 따라서 최적의 윈도우 크기를 결정하는 것은 특히 모바일 환경에서는 읽기 오류를 줄이기 위해 쉽지 않은 중요한 일이다. 본 논문에서는 지능형 윈도우 크기 조정을 통해 읽기 오류를 줄이기 위하여 단일 태그를 위한 RFID 데이터 정제 방안을 제안한다. 이항 샘플링을 기반으로 한 기존 연구와 달리, 본 논문에서는 가중치 평균을 사용한다. 이는 최근의 읽기가 더 정확한 현재의 태그 전이를 나타낼 수 있으므로 과거와 현재의 읽기를 차별화하는 일이 필요하다는 것에 기반을 두고 있다. 가중치 평균을 사용하므로 이질적인 읽기 패턴을 갖는 모바일 환경에서도 효율적으로 적응하여 윈도우 크기를 동적으로 조정할 수 있게 된다. 뿐만 아니라, 윈도우 내의 읽음 패턴과 감소되는 윈도우 크기의 효과를 분석함으로서 더욱 효율적이고 정확한 크기 조정 결정을 할 수 있도록 한다. 제안한 방안을 사용하면 RFID 미들웨어 시스템이 응용에 좀 더 정확하고 무결점의 데이터를 제공함으로써 본래의 응용 서비스 품질을 보장할 수 있도록 한다는 궁극적인 목적을 달성할 수 있을 것으로 기대한다.

주제어 : 데이터 정제, RFID 미들웨어 시스템, 윈도우 타임 슬라이드, 태그 전이

* 중앙대학교 경영경제대학 경영학부 교수
** 중앙대학교 대학원 문화예술경영학과 박사과정
*** 교신저자 : 박세권
중앙대학교 경영경제대학 경영학부 교수
Tel: +82-31-670-3058, Fax: +82-31-675-1384, E-mail: psk3193@cau.ac.kr

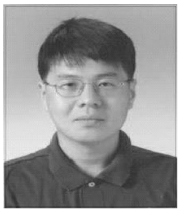
저 자 소 개

신 동 천

서울대학교 컴퓨터공학과를 졸업한 후 한국과학기술원 전산학과에서 석사와 박사학위를 각각 취득하였다. 1991년부터 한국전산원에서 선임연구원으로 근무하였으며 1993년부터 중앙대학교 경영경제대학 경영학부에서 교수로 재직 중이다. 최근의 주요 관심분야는 IT 서비스 모델링, 기업정보시스템, 비즈니스 인텔리젼스, e-biz 등이다.

오 동 옥

중앙대학교 정보시스템학과에서 2008년과 2011년에 각각 학사와 석사학위를 취득하였다. 2014년부터 중앙대학교 문화예술경영학과에서 박사과정에 재학 중이다. 최근의 주요 관심분야는 서비스 모델링, 디지털 콘텐츠 생태계 연구 등이다.



류 승 완

고려대학교 산업공학과에서 1988년과 1991년에 각각 공학사와 공학석사를 취득하였으며, 뉴욕주립대(SUNY at Buffalo) 산업공학과에서 2003년에 공학박사를 취득하였다. 1991년부터 1993년까지 LG전자 영상미디어연구소에서 주임연구원으로 근무하였으며, 1993년부터 2004년까지 한국전자통신연구원 이동통신연구단에서 선임연구원으로 근무하였다. 2004년부터 중앙대학교 경영경제대학 경영학부에서 교수로 재직 중 이며, 주요 연구 관심 분야는 이동통신시스템설계 및 성능분석, 무선 MAC 프로토콜, 차세대 이동통신 서비스 및 비즈니스 모델 개발, 문화 및 디지털 컨텐츠 생태계 연구 등이다.

박 세 권

서울대학교 공과대학과 대학원 산업공학과에서 1978년과 1981년에 공학사(BS)와 공학석사(MS)를 취득하였으며, Texas A&M 대학교에서 1985년에 산업공학 박사(Ph.D.)를 취득하였다. 1978년부터 1980년까지 한국전자통신연구소에서 연구원으로 근무하였으며, 1985년부터 1987년까지 한국전자통신연구원 통신망계획부에서 선임연구원으로, 1987년부터 1990년까지 농촌경제연구원에서 수석연구원으로 근무하였다. 1990년부터 현재까지 중앙대학교 경영경제대학 경영학부에 재직 중이며 연구관심 분야는 시스템공학 등이다.