A Study on Device Management for IoT Services with Uncoordinated Device Operating History

Megumi Shibuya[†], Teruyuki Hasegawa[†] and Hirozumi Yamaguchi[‡]

[†]KDDI R&D Laboratories, Inc.
Saitama, JAPAN
e-mail: {shibuya, teru}@kddilabs.jp

[‡]Osaka University
Osaka, JAPAN
e-mail: h-yamagu@ist.osaka-u.ac.jp

***Abstract*—The cumulative failure rate is an important reliability index for evaluating Internet of Thing (IoT) device and system quality and reliability. However, in the horizontal specialization business model, IoT service infrastructure is often operated by multiple players such as service providers and device vendors, and device management information necessary to obtain the cumulative failure rate are owned independently and uncoordinatedly by them. In this paper, we propose a method of calculating the cumulative failure rate in such environment. We design an algorithm to aggregate and organize such distributed, uncoordinated information to derive the device operating history, which is fed into the cumulative failure rate calculation formula. Through several simulation experiments, we show the effectiveness of our method in some realistic scenarios.**

Keywords - IoT; Reliability; Cumulative failure rate; Operating history; Multiple Players

I. INTRODUCTION

The concept of Internet of Thing (IoT) has been widely penetrating. According to [1], the number of devices which are available for mobile access is expected to grow to approximately 50 billion units (6.58 units per user) by 2020. As IoT-based systems are becoming more indispensable, they should be more reliable to achieve sufficient service availability. This cannot be achieved without high reliability and dependability of *IoT devices* themselves.

Cumulative failure rate is often utilized as a device reliability index [2]. It is a probability of occurrence of failure in a certain time period starting from the time when the device becomes in operation. The cumulative failure rate is usually derived using the failure rate for every unit of time, which is defined as a ratio of the number of failed devices to the number of devices being in operation in the unit of time. Here, the devices being in operation may vary at every moment not only due to device failure but also due to operational activities such as new device installation, removal and replacement. Therefore, we have to trace the operating history of each device to calculate the cumulative failure rate.

However, in order to obtain the operating history of each device, it is required to obtain the occurrence dates of device-associated events such as installation of the device, suspension and resumption of device utilization and failure. If the device manufacturers (simply called *vendors*) themselves provide services (this way of service provision is called *vertically integrated business model* [3]), such information can be obtained easily as it is managed in a single place. In contrast, in *horizontal specialization business model* [4] where service providers (simply called *providers*) purchase the devices from vendors and use them (this style is often seen in smart meter services and the Internet access services), device operating history is owned and managed partially and uncoordinatedly by multiple business operators called *players*, which will cause a significant issue in building a single, consistent view of operating history.

For example, in smart meter services, an electric company (i.e., a provider) purchases power meters in bulk from a vendor, lends them to subscribers, and stocks the rest as spare ones. When a power meter becomes out of order, the provider supports to replace it and orders a repair service to the vendor. Then, the vendor is able to manage the product-related information such as the production date and model number of the meter as well as the failure-related information such as the date and reason of failure and the repair process. However, the device operating status (e.g., operating start date) is not observed by vendor. Meanwhile, the provider has to manage the subscriber information including the asset information (e.g., the current meter location). Hence, it is not necessary to trace back the information about failure and others. Consequently, in order to obtain a consistent history of meters, it is necessary to design a method of aggregating the management information which are separately and uncoordinatedly managed by multiple players to enable calculation of the cumulative failure rates.

In this paper, we propose a method of calculating the cumulative failure rate, which is an important reliability index that represents device reliability. We assume that services are provided (i) using a large quantity of homogeneous devices and (ii) following the horizontal specialization business model where multiple players are involved and management information are owned separately and uncoordinatedly by them. Then, the method aggregates and analyzes those distributed information to derive the operating history of each IoT device to enable calculation of cumulative failure rates.

The contributions of this work are three-hold. Firstly, we deal with a significant issue of IoT device management through real case studies (based on our business experience) on how we grasp and measure the device reliability, which is mandatory for maintaining the quality of large-scale IoT service infrastructure operated by multiple players in the horizontal specialization business model. Secondly, we propose a method to obtain the operating history of each IoT device from various types of management information. We note that calculating the cumulative failure rate using complete device history is usual in device management, but it is not straightforward taking those devices which are often replaces, repaired and reused at different times and locations into account. Finally, we present the experimental result of measuring the accuracy of cumulative failure rates with realistic scenarios where a part of information is missing, a situation that often occurs in the real environment.

This paper is organized as follows. Section II summarizes related work and Section III introduces a service scenario in IoT infrastructure with multiple players. Section IV presents our method and experimental results are shown in Section V. We conclude this work in Section VI.

II. RELATED WORK

There have been various activities on evaluating product quality and device reliability [5]-[8]. Several studies on Operation And Management (OAM) issues of IoT devices in IoT service infrastructure [9]-[11] have also been conducted.

References [5][6] present evaluation methods at the design or production phase of devices, where the cumulative failure rate is estimated by modeling the occurrence of major failures at the component level of devices. Reference [5] focuses on Hot Carrier Injection (HCI) and Time Dependence Dielectric Breakdown (TDDB) of N-channel Metal Oxide Semiconductor (NMOS) as a major failure factor. Reference [6] discusses how to determine a new parameter from failure factors observed in the field, e.g., electrostatic discharge inrush current, to combine with a conventional estimation method for more accurate cumulative failure rate at the product design phase. These approaches assume that all information elements, which are necessary for calculating the cumulative failure rate, are maintained in the vendor manufacturing the devices, and it is not considered that the number of devices varies by factors other than failures.

On the other hand, [7][8] propose evaluation methods for product reliability based on the observation of each device's operation history considering device changes due to non-failure, which is not taken into account in the existing work [2][12]. Specifically, the failure rate is calculated by the number of devices and time differentiation of the cumulative number of failure devices at the given timing of elapsed time.

All the above assume a vertical integration structure in which all the information elements for calculating the cumulative failure rate (or a similar index) are maintained by only one player. In contrast, we are focusing on IoT service infrastructure in which multiple players (e.g., providers and vendors) are collaboratively involved. In such a horizontal specialization structure, the followings should be done to manage the product reliability of IoT devices for realizing dependable infrastructure; 1) coordinating the management information provided by each player, 2) extracting and deriving information elements and 3) reconstructing the device operation history from the information elements. As far as we know, this is the first activity focusing on IoT device management with multi-player issues.

Meanwhile, there have been many approaches so far toward IoT device applications [9]-[11], which basically focus on the management and configuration of remote sensor devices over the Internet. Therefore, they do not deal with the IoT device management issues.

III. SERVICE SCENARIO

In this paper, we assume IoT infrastructure with multiple players in the horizontal specialization business model. Under this assumption, we explain the device operating and management information that are separately and uncoordinatedly managed by multiple players. The scenario is based on our own experience, so everything here is likely to occur in the real world business.

As explained briefly in Section I, we target such a service provider as an electric company or a network provider that purchases the devices in bulk from an IoT device vendor and lends them to subscribers (users). Figure 1 illustrates the interactions between each player and users. We explain the service provision scenario using this figure.

(1) IoT Device Purchase and Stocking: The provider purchases IoT devices from the vendor and stocks them as spares. Lending an IoT device from the provider to a user and returning it by the user due to cancellation is conducted via the provider's warehouse. The provider records the *current* location of the purchased IoT devices in asset management information. The vendor records the product-related information such as the shipping date and model number of IoT devices in shipment management information.

(2) Service Startup: The provider creates the contract-related information for every user and manages it. The provider lends an IoT device to a user, starts the service and records the service start date in user contract information.

(3) IoT Device Failure and Replacement: When an IoT device fails at a user location, the provider sends an alternative IoT device to the user. The user sends the failed IoT device to the address of the repair service. Since the vendor repairs the failed IoT device, the user sends it back to the vendor directly to optimize the transport route. The vendor repairs the failed IoT device and records the failure-related information such as date and model number (or send-back date or receiving date as the date of failure). After the repaired IoT device is sent from the vendor to the provider, it is stocked in the warehouse. The provider updates the records related to these two devices (i.e., the current locations of failed and alternative devices).

(4) Service Cancellation: When a user cancels its contract, she/he returns her/his IoT device to the provider. The provider re-stocks it in the warehouse and updates the current location information of the IoT device. Furthermore, the service end date of this user is recorded in the contract-related information.

In summary, the provider maintains a) contract information with users and b) asset management information of IoT devices, the vendor maintains c) shipment-related information of IoT devices and d) the failure-related information. Here, b) is usually sufficient for asset management by the provider. This is because the provider does not care about whether an IoT device was installed at different locations in the past.

On the other hand, as described in Section I, it is required to obtain the occurrence dates of device-associated events such as installation of the device, suspension and resumption of device utilization and failure to calculate the cumulative information such as the service start date and suspension and resumption of the device date in this scenario, and the provider does not observe the information such as the failure date. Therefore, each player cannot collect and build complete device-associated information. This is our motivation to provide a method to build complete operating history of each IoT device from such partial, distributed operating and management information as indicated by the above a) to d).

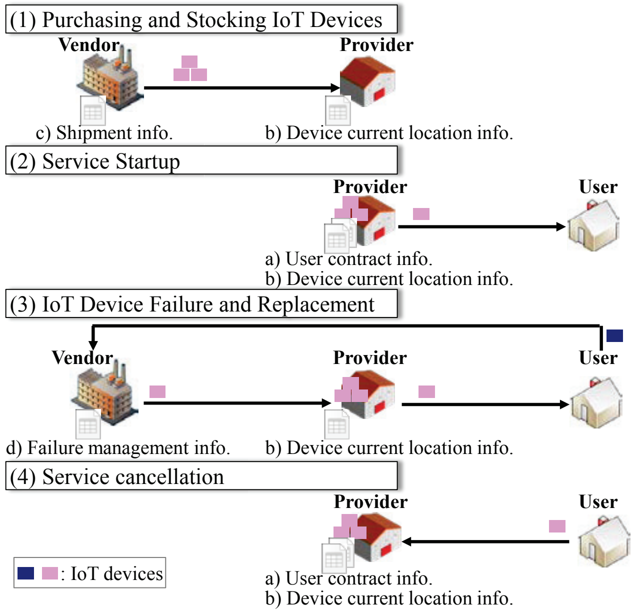

(1) Purchasing and Stocking IoT Devices
Vendor
Provider
c) Shipment info.
b) Device current location info.
(2) Service Startup
Provider
User
a) User contract info.
b) Device current location info.
(3) IoT Device Failure and Replacement
Vendor
Provider
User
d) Failure management info.
b) Device current location info.
(4) Service cancellation
Provider
User
■ ■: IoT devices
a) User contract info.
b) Device current location info.

Figure 1. Interaction among multiple players and user.

Firstly, from a) and b), we try to obtain List-A of Table I with its created date *Tc*. Each pair of the sequence number *SN* and its location *L* can be obtained from b). This *L* is matched with that in a) to associate this pair with the service start date *T1* contained in a). If this device has not been failed since the day of initial installation at a user, we can obtain the history indicating that device *SN* has been working without failure between *T1* and *Tc*, which results in the fact that the operation start date *T4* is *T1* (*T4*=*T1*). Meanwhile, if there was a failure, *T1* is set to the date when an alternative device is started working at location *L*. In this case, *T4*, the operating start date of the original device *SN,* is left *unknown*.

Secondly, we try to obtain List-B of Table I from a) and d). In this scenario, the vendor records the location where the failure occurred (this kind of information is generally useful for such vendors which need some statistics of failure occurrence patterns). Here, we should consider how we will obtain column *T5*, which is the operation start date of each failed device. To do this, we associate date *T2* of failure, the sequence number *SN* and location *L* with contract information of a). Then, we obtain *T5* and the history indicating that device *SN* installed at location *L* had been working from *T5* until *T2* and then failed at *T2*.

Moreover, a) contains the service end date *T3* and the service start date *T1*. If we have sequence number *SN* of the device that was returned from location *L,* we can obtain List-C of Table I containing *T6,* the operation start date of the returned device. We note that the provider may not be motivated to record sequence number *SN*. Similarly with the List-A case, from this List-C, we can obtain the history indicating that the returned device *SN* had been working from *T6* until *T3* without failure. Under a certain condition, *T6* is equal to *T1*.

TABLE I. SERVICE OPERATING AND MANAGEMENT DATA

(LIST-A) CURRENT DEVICE LIST (MANAGED BY PROVIDER)

List created date (=Today) (*Tc*) : 2014/10/01			
Location (*L*)	Service start date (*T1*)	Current Device (*SN*)	Operating start date of current device at *L* (*T4*)
1	2014/01/01	a	-
3	2014/05/01	b	-
4	2014/03/01	c	-
:	:	:	:

(LIST-B) FAILED DEVICE LIST (MANAGED BY VENDOR)

Location (*L*)	Failed Date (*T2*)	Failed device (*SN*)	Operating start date of failed device at *L* (*T5*)
1	2014/02/01	a	-
3	2014/04/01	a	-
1	2014/04/01	b	-
:	:	:	:

(LIST-C) RETURN DEVICE LIST (OUT OF MANAGEMENT BY PROVIDER)

Location (*L*)	Return date (*T3*)	Return device (*SN*)	Operating start date of return device at *L* (*T6*)
3	2014/03/01	c	-
5	2014/06/01	d	-
2	2014/07/01	e	-
:	:	:	:

(▨: unknown)

In the next sections, we present how *T4*, *T5* and *T6* are obtained using List-A, B and C, and how the cumulative failure rate is calculated using the history.

IV. PROPOSED METHOD

A. Overview

In this section, we explain how to obtain the cumulative failure rate of IoT devices whose management information is maintained separately and uncoordinatedly by multiple players. Our proposed method consists of the following three Steps;

Step1: Reconstructing the operating history of each IoT device,

Step2: Counting the operating days, and

Step3: Calculating the cumulative failure rate.

Specifically, our proposed method basically uses List-A and B for reconstructing the operating history, and List-C as well as (if exists). Note that even without List-C, the method can reconstruct the history but some error may occur because *T3*, and *T6* in List-C are not plotted on the time-sequence diagram (See Figure 2 in Section IV-B). We numerically evaluate the impact of such error in Section V.

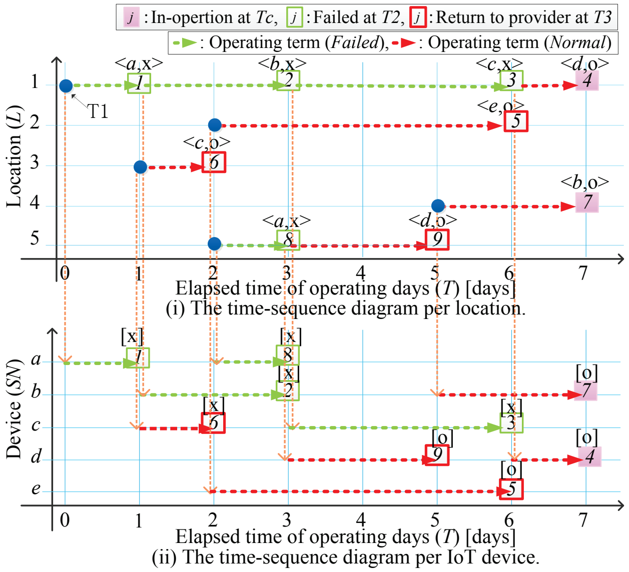
Copyright (c) IARIA, 2016. ISBN: 978-1-61208-450-3

B. Design Details

Step1: Reconstruct the operating history of IoT device.

Step1-1) Create time-sequence diagram per location.

(1) First, the time-sequence diagram per location is created as shown in Figure 2 (i) where x- and y-axes are # of days passed (denoted as *T*) from the reference date "0" and location *L* (1, 2, …), respectively. Current date *Tc*, failed date *T2* and return date *T3* in List-A, B, and C are plotted as square boxes on the diagram at (*x*,*y*)=(*Tc*/*T2*/*T3*, relevant *L*), respectively. Note that for easy understanding, in Figure 2, we assign a numeral number *j* to each plot as ID. It is denoted inside the square box corresponding to the plot.


j : In-opertion at *Tc*, *j* : Failed at *T2*, *j* : Return to provider at *T3*
: Operating term (*Failed*), : Operating term (*Normal*)
Location (*L*)
1
2
3
4
5
<*a*,x>
<*b*,x>
<*c*,x>
<*d*,o>
<*e*,o>
<*c*,o>
<*b*,o>
<*a*,x>
<*d*,o>
T1
1
2
3
4
5
6
7
8
9
0
1
2
3
4
5
6
7
Elapsed time of operating days (*T*) [days]
Device (*SN*)
a
b
c
d
e
[x]
[x]
[x]
[x]
[x]
[o]
[o]
[o]
[o]
0
1
2
3
4
5
6
7
Elapsed time of operating days (*T*) [days]

(i) The time-sequence diagram per location.

(ii) The time-sequence diagram per IoT device.

Figure 2. Time-sequence diagrams for reconstructing the operating history of each IoT device.

(2) For each plot *j*, device *SN* and device operating status x="*Failed*" or o="*Normal*" at the relevant date are associated as its attribute. For instance in Figure 2 (i), plot *j*=8 with <*a*,x> (at *T*=3 and *L*=5) means that the device *a* was "*Failed*" at location *L*=5 (then it was sent back to the vendor for repair). Plot *j*=7 with <*b*,o> means that the device *b* was "*Normal*" at *L*=4 (therefore it is in-operation now (*T*=7)). Plot *j*=9 with <*d*,o> means that the device *d* was *"Normal"* at *L*=5 (because it was returned to the provider without failure due to user cancellation at *T*=5).

(3) Service start date *T1* at location *L* in List-A is plotted on the diagram.

(4) We assume that the failed device is replaced to another device on the same day for non-stop service. Along the time-sequence of each location *L*, the plots on it are traced back from the current date *Tc* (*T*=7) to the reference date (*T*=0) in order to determine the start date (referred to as S_j) of each plot *j* at the location. The date of *j*'s previous plot is regarded as the start date of *j*. For example, the start date of device *b* at plot *j*=2 ((*x*,*y*)=(3,1)) is determined as S_2=1 because its previous plot (*j*=1) is at *T*=1 ((*x*,*y*)=(1,1)).

(5) The operating term for each plot *j* can be extracted as the term from S_j to *T* of plot *j*. For example, device *b* at plot *j*=2 is operated from S_2=1 to *T*=3 so the term is 2 days. As another example, device *b* at *j*=7 ((*x*,*y*)=(7,4)) is operated from S_7=5 to *T*=7 (now in-operation) so the term is also 2 days.

Step1-2) Transform time-sequence diagram per location to per IoT device.

(1) The time-sequence diagram per IoT device (see Figure 2 (ii)) is transformed from Figure 2 (i). At first, each plot *j* in Figure 2 (i) is re-plotted on Figure 2 (ii) according to the device *SN* in its attribute. Note that the device operating status x/o in its attribute is inherited. For instance, plot *j*=2 with <*b*,x> ((*x*,*y*)=(3,1)) in Figure 2 (i) is re-plotted to *j*=2 with [x] ((*x*,*y*)=(3,*b*)) in Figure 2 (ii).

(2) For each plot *j*, the relevant operating term from S_j to *T* of the plot *j* is drawn on Figure 2 (ii). It is easy to obtain each IoT device's operating history by collecting operating terms per device from Figure 2 (ii). For instance, the operating history of device *b* includes two operating terms, i.e., S_2=1 to *T*=3 (*Failed*) and S_7=5 to *T*=7 (*Normal*).

Step2: Counting the operating days.

Two types of operating days are counted per IoT device from the operation histories. The first type is referred as "*Failed days*" (*P*) which is ended with a plot derived from *T2*, i.e., failed date. The second type is referred to as "*Normal days*" (*Q*) which is ended with a plot derived from *T3* or *Tc*, i.e., return or current date without failure. For counting the operating days of each IoT device, an operation term of the device is selected in chronological order and checked whether the term is ended by *T3* or not. If so, the term should be concatenated to the next operating term (if any) as a single piece of operating days. For example in Figure 2 (ii), the device *b* is set *P*=2 and *Q*=2, while the device *c* is set *P*=4(=1+3), and the device *d* is set *Q*=3(=2+1). *P* and *Q* of each device are shown in Table II.

TABLE II. THE OPERATING DAYS FOR EACH *SN* IN FIGURE 2 (ii)

SN	# of operating days (# of terms)	Operating days [days]	
		1st	2nd
a	2 (2)	*P*=1	*P=1*
b	2 (2)	*P*=2	*Q=2*
c	1 (2)	*P*=4(=1+3)	-
d	1 (2)	*Q*=3(=2+1)	-
e	1 (1)	*Q*=4	-

Step3: Calculating the cumulative failure rate.

From both *Failure days* and *Normal days* in Step2, the cumulative failure rate is calculated. Let *f*(*x*) denote the failure density function, the failure occurrence probability until time *i* has passed, i.e., the cumulative failure ratio *F*(*i*), is expressed in Eq. (1) [2].

$$F(i) = \int_0^i f(x)dx \qquad (1)$$

We can approximately obtain the following difference equation by differentiating Eq. (1) and substituting infinitesimal *di* to the unit time (a day).

$$F(i) - F(i-1) = f(i) \qquad (2)$$

ISBN: 978-1-61208-450-3

Here, let $\lambda(i)$ denote the failure rate per unit time. Since $f(i) = \left(1 - F(i-1)\right) \cdot \lambda(i)$, E.q. (2) is expressed as;

$$F(i) = F(i-1) + \left(1 - F(i-1)\right) \cdot \lambda(i) \qquad (3)$$

where

$$F(0) = 0, \qquad \lambda(i) = \frac{n(i)}{N(i) + n(i)} \quad (i = 1, 2, \ldots) \qquad (4)$$

and $n(i)$ is the number of failed devices (P = i-1) at day i, $N(i)$ is the number of in-operation devices at the end of day i. From the above discussions, the cumulative failure rate can be calculated from the operating history.

In addition, the number of returned devices, which is suspended at day i, is given by $N(i\text{-}1)$ – ($N(i)$ + $n(i)$). Assuming that $\lambda(i)$ is the same regardless of the devices, the cumulative failure rate is not affected even if suspended devices exist.

V. Experimental Evaluation and Discussion

We verify that the proposed method expressed in Section IV can reconstruct operating histories and calculate the cumulative failure rate. In addition, we evaluate the accuracy of the cumulative failure rate when some information elements are missing.

A. Experimental Setup

In order to validate the effectiveness of our proposal, we develop the simulator implementing the proposed method. The input data set for this simulator consists of List-A, B, and C (if any) without *T4, T5*, and *T6*. From these input data set, the simulator complements the unknown fields (*T4, T5*, and *T6*), then reconstructs operating histories and calculates the cumulative failure rate. This simulator is a Ruby program with approximately 17,000 lines, executing on a PC with the following specifications; CPU: E5-2650L v2@1.70GHz, memory: 126GBytes, OS: CentOS 6.6, Ruby 1.9.3.

Evaluation data sets as shown in Table III are arranged with various return rates *R* and failure rates *U*, both of which follow uniform distribution irrespective of *T*. Simulation days *T* is 1,826 days (= 5 years), the maximum number of devices is 70,000 [units], the maximum number of locations is 10,000. We assume no IoT devices are in-operation at *T* = 0, and the replacement of failed device is finished on the same day as the failure occurs. For accurate evaluation results, we arranged 180 data sets in total, because 18 *R*/*U* pairs are specified and 10 random data sets are generated per *R*/*U* pair. Note that these data sets are given as List-A, B and C. At first, a data set consists of all the information elements in List-A, B and C is generated (we call it "reference data set"). Then, an evaluation data set is created from it by omitting unknown fields, i.e., *T4*, *T5* and *T6* or *T4*, *T5* and whole List-C, according to the case in Table IV.

TABLE III. Parameters of Creating Evaluation Data

Parameters	Values
Failed rate *U* [%/day]	0.2, 0.5, 0.8
Return rate *R* [%/day]	0, 0.2, 0.4, 0.6, 0.8, 1.0
The number of simulation days *T* [days]	1,826(= 5 years)
The number of devices [units]	15,000～70,000
The number of locations [locations]	100～10,000

TABLE IV. Verification Cases

Case #	List-C management / non-management	unknown data
Case1	management (=use List-C)	*T4*, *T5*, *T6*
Case2	Non-management (=not use List-C)	*T4*, *T5*, List-C*

**T6*: unknown because of List-C unmanaged

B. Verification of proposed method

We verify that the proposed method can complement the unknown fields in List-A, B, and C in Case1 and Case2 by comparing with the reference data sets. We also reconstruct operating histories and calculate cumulative failure rates from all the evaluation data sets.

As a result, we confirm that the simulator successfully completes the above processes for any data sets in any cases. In Case1, all *T4*, *T5* and *T6* of event start date are completely matched with those in the reference data sets. In contrast, in Case2, *T4* and *T5*, which are operating start dates of failed and current devices respectively, are different from those in the reference data sets. This is because a part of these start dates are changed due to the lack of List-C. As one of the examples at *R*=1.0, 18.1%, 23.6% and 24.9% of these start dates, i.e., *T4* and *T5*, are changed in average when *U*=0.2, 0.5 and 0.8, respectively.

On the other hand, the example computation time to obtain the operating history and the cumulative failure rate is approximately 3 [min] and 1 [min], respectively, in the case of *U*=0.8 and *R*=1.0 with 70,000 IoT devices.

C. The Effect of cumulative failure rate by return rate R

Figure 3 shows the cumulative failure rate *F*(*i*) in each failure rate *U* where the return rate *R* varies from 0.0% to 1.0% at 0.2% intervals. Each plot indicates the average of *F*(*i*) values individually calculated from 10 random data sets arranged per *R*/*U* pair.

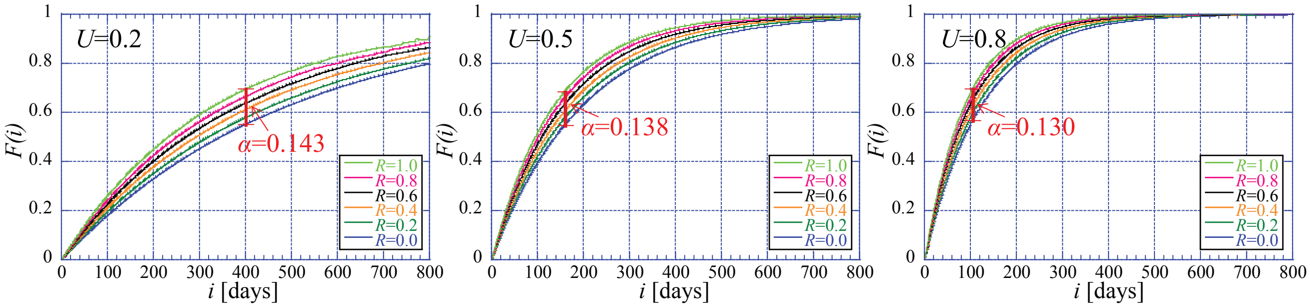


Figure 3. Cumulative failure rate *F*(*i*) vs. operating days *i* varied return rate *R* of each.

It is clear in Figure 3 that $F(i)$ increases in proportion to R irrespective of U and that the larger R becomes, the more rapidly the cumulative failure rate $F(i)$ increases. As for the errors of $F(i)$ (referred to as α) between R=0.0 and 1.0, α =0.143 at i=400, α=0.138 at i=159, and α=0.130 at i=105, respectively. So the error α decreases with increase of U.

Figure 3 also indicates that our method conservatively underestimates the cumulative failure rate. Hence from the provider's perspective, the calculated rate can be useful when the provider discloses it to the vendor for encouraging more improvement on the product quality and reliability of IoT devices. However, in the reverse direction from the vendor to the provider, such under estimation may mislead what each player should do next. So, the calculated rate should be interpreted carefully according to the player's role.

D. On addition of information elements after service start

In this scenario, we assume that each player is dedicated to playing his role and does not have any incentive to maintain extra management information beyond his role. However, in the real world, it is probable that either player may add some management information due to emerging new operational requirements after the service start, e.g., similar service infrastructures and their providers are merged into one.

Here, we briefly and qualitatively discuss on how to handle such management information change, especially in the case that some useful information elements can be obtained after a certain date. For example in Sec. III, it is considerable that the service was started with a very small number of users, it was not so important for the provider to improve product reliability of IoT devices at first. However, as increasing of IoT devices after time goes by, the provider want to improve the product reliability of IoT devices more so that he start maintaining List-C from a certain date ($T=Y$)

In such a case, return date *T3* in List-C is started from Y and there are no previous records before it, i.e., from $T=0$ to $Y-1$. For calculating the cumulative failure rate, we can choose one of the following three options.

1) The calculation is conducted using recorded *T3* ($T3 \geq Y$) only, assuming that no return event, i.e., service cancelation, occurs at $T<Y$.
2) The calculation is conducted after complementing *T3* at $T<Y$ based on R calculated from recorded *T3* ($T3 \geq Y$).
3) The calculation is conducted without List-C.

Among above three methods, Option 3 is equivalent to the result of $R=0.0$ in each U in Figure 3. According to the results in Figure 3, the cumulative failure rate is qualitatively expected Option 3 > Option 1 > Option 2. The quantitative evaluation of Options 1 and 2 is left as our future work.

VI. CONCLUSION

In this paper, we have proposed a method of calculating the cumulative failure rate in IoT service infrastructure operated by multiple players such as service providers and device vendors in the horizontal specialization business model. The method aggregates and analyzes distributed information to derive the operating history of each IoT device to enable calculation of cumulative failure rates. We have verified that the proposed method can derive operating histories and calculate the cumulative failure rate. In addition, we have evaluated the accuracy of the derived cumulative failure rates when some information about device operation are missing. Furthermore, we have discussed cases where the missing information are added after the service is started to see the performance of our method in real world cases.

We are now planning to apply our method to more different cases. Based on our business experience, there are a variety of scenarios where management information are distributed and unmanaged. We believe that we have shown the applicability of our method by introducing well-seen, representative cases in this paper, but examination of our approach in a variety of scenarios is part of our future work.

REFERENCES

[1] Cisco Systems Inc., "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2013-2018," http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white_paper_c11-520862.html, accessed Jan. 8, 2016.

[2] W. Zheng, W. Zengquan, and W. A-na, "Failure Rate Calculating Method of Components Based on the Load-strength Interference Model," Proc IEEE Industrial Engineering and Engineering Management IEEM 2010, pp.783-787, Dec. 2010.

[3] OPS rules, "Vertical vs. Horizontal Integration: Which is a better Operations Strategy?," Sep. 2012, http://www.opsrules.com/supply-chain-optimization-blog/bid/241648/Vertical-vs-Horizontal-Integration-Which-is-a-Better-Operations-Strategy, accessed Jan. 8, 2016.

[4] Z. Yu, "IT, Production Specialization, and Divison of Labor: A Smith-Ricardo Model of International Trade," Carleton Economic Paper, Jun. 2003, http://carleton.ca/economics/wp-content/uploads/cep03-06.pdf, accessed Jan. 8, 2016.

[5] Z. Zhou, X. Liu, Q. Shi, Y. En, and X. Wang, "Failure Rate Calculation for NMOS Devices under Multiple Failure Mechanisms," Proc International Symposium on the Physical and Failure Analysis of Integrated Circuits (IPFA), pp.362-365, Jul. 2013.

[6] T. Tekcan, G. Kahramanoglu, and M. Giinduzalp, "Determining Reliability by Failure Rate Estimation via a New Parameter," Proc Reliability and Maintainability Symposium (RAMS), pp.1-7, Jan. 2012.

[7] H. Funakoshi and T. Matsukawa, "A Failure Rate Estimation Considering the Change in the Number of Equipments," IEICE NS2009-17, pp.1-6, 2009 (in JAPANESE).

[8] H. Funakoshi and T. Matsukawa, "A failure Rate Estimation Considering the Change in the Number of Equipments," IEICE Trans. B Vol. J93-B No.4, pp.681-692, 2010 (in JAPANESE).

[9] Z. Sheng, H. Wang, C. Yin, X. Hu, S. Yang, and V.C.M. Leung, "Lightweight Management of Resource-Constrained Sensor Devices in Internet of Things," in IEEE Internet of Things Journal, vol.2, no.5, pp.402-411, Oct. 2015.

[10] C. Zhou and X. Zhang, "Toward the Internet of Things application and management: A practical approach," in IEEE WoWMoM 2014, pp.1-6, 2014.

[11] S.N. Han, S. Park, G.M. Lee, and N. Crespi, "Extending the Devices Profile for Web Services Standard Using a REST Proxy," in IEEE Internet Computing, vol.19, no.1, pp.10-17, Jan.-Feb. 2015.

[12] M. Xie, Y. Tang, and T. N. Goh, "A modified Weibull extension with bathtub-shaped failure rate function," Reliability Engineering and System Safety, 2002, 76(3): 279-285.