Regular Paper

Assessment and Simulation of System Performance of a Cloud-based Data-protection System

Yuichi Taguchi[1,2,a)] Tsutomu Yoshinaga[2]

Received: December 25, 2017, Accepted: May 5, 2018

Abstract: A method for assessing performance of a cloud-based remote data-protection system is proposed. This method makes it possible to determine whether the system performance achieves recovery point objective (RPO) or not. It is also necessary to resize the data-protection system for satisfying required performance with smaller expenses. Therefore, we established a formula for simulating system performance. Based on the results of the simulation, we show it is possible to determine an appropriate system size that achieves RPO as well as minimizes the amount of system resources. It is experimentally demonstrated that there are no gap between the experimentally measured and simulated recovery points at peak of system workload. So we concluded that the proposed simulation is accurate enough. Also, it is estimated that 89% of system resource expenses could be reduced in comparison to the situation in which data-protection system is designed to fit a peak workload.

Keywords: data protection, data backup, disaster recovery, cloud, RPO

1. Introduction

1.1 Significance of data protection

Today, enterprise IT systems are an essential foundation for business operations. They generate large amounts of data that has been increasing steadily [1], [2]. And many attempts to discover new value from the analytics of big data, such as digital maps, images, and enterprise-business records, have been made [3], [4], [5]. In other words, the data itself is recognized as a valuable asset [6], [7]. As a result of this trend, data loss causes enormous damage to enterprises from the viewpoint of not only loss of business opportunities but also losing customers' trust and social credibility [8]. Accordingly, data protection is one of the most-important tasks concerning enterprise IT operations, and various solutions have been deployed [9].

A variety of data-protection methods are available, and which one to use depends on the importance of the IT system implemented. To protect important data, not only local data backup or duplication [10] but also countermeasures against site-scale damage (such as terrorist attacks and natural disasters) are required. Disaster recovery has been deployed to confront these threats. It replicates data to sites that are geographically separated and implemented with an alternative IT system. Even in case of a disaster, the system keeps running at the alternative site [11], [12], [13].

1.2 Requirements for data-protection system

To design a data-protection system a variety of requirements must be satisfied. First, enterprise companies have to define their requirements in advance of system implementation. The requirements may concern performance, cost, distance from the local site to the remote site, a specified country in which to store the backup data, a service provider to manage the data, and so on. They must be defined according to the kind of business operations and the importance level of the data [14], [15], [16].

Among these requirements, two performance indices for data protection are defined as "recovery point objective (RPO)" and "recovery time objective (RTO)". RPO represents the risk of data loss. The "recovery point" is a performance parameter that is defined as the difference between the time at which a failure or disaster occurs and the time at which data can be recovered retroactively. And RPO is a preset value of performance target of the recovery point. As an example, if RPO is 60 seconds, it is required to recover all data recorded up to 60 seconds before the time of failure. On the other hand, RTO represents recovery time after the time of disaster.

A cost requirement is also important because the operation expense of an IT system is one factor in decreasing a company's profit.

1.3 System management

The system-performance logs are time-series digital data. They are discrete values recorded at preset intervals. As an example, Amazon Cloudwatch, which is a system-monitoring service provided by Amazon Web Services [*1] (AWS), produces time-series performance logs [17]. In the case of Cloudwatch, the interval is a variable, and the IT administrator can control the unit time of the interval from one second to sixty minutes. Statistics such as sum,

[1] Research and Development Group, Hitachi, Ltd., Kokubunji, Tokyo 185–8601, Japan
[2] Graduate School of Informatics and Engineering University of Electro-Communications, Chofu, Tokyo 182–8585, Japan
[a)] yuichi.taguchi.nh@hitachi.com

[*1] "Amazon Web Services", "AWS" and "Amazon CloudWatch" are registered trademarks of Amazon Technologies, Inc.

maximum, and minimum values of the data points measured during the unit time are output at intervals of unit time. For example, if the time interval is set as five minutes, maximum, and minimum values at the observation points from after 11:00 to 11:05 per second are recorded as a statistical value of 11:00.

Although the accuracy of system monitoring is better when the time interval is shorter, huge loads in terms of data recording and transfer processing are generated at application servers. These loads can degrade the application performance. From the viewpoint of IT administrators, it is important to grasp the system behavior with a high level of accuracy; however, it is more important to keep system stability. In consideration of such a trade-off, the time interval parameter should be designed appropriately.

In this research, the time count of the recovery point depends on the interval period by which logs are recorded. That is, if the time interval is set to 5 minutes, the recovery point parameter is the output in multiples of 5 minutes.

1.4 Research objective

The objective of this research is to make it possible to control the risk of data loss. Although we premise that RPO is achieved, we try to save the expense of system resource consumption by allowing the risk of data loss at the time of high workload. In order to achieve that target, we propose a method for controlling the recovery point. This method reduces system expense as much as possible within the range in which the recovery point achieves the pre-defined RPO. To address this solution, the two technologies described below are proposed.

(1) **Assessment of system performance**

An assessment method makes it possible to monitor a recovery point that is an actual performance measurement of a running system.

(2) **Simulation of system performance**

A recovery point parameter of one system that is assumed as a candidate for a new configuration can be calculated by simulation. By this method, it is possible to find a system configuration that satisfies a performance requirement as well as a lower cost.

1.5 Effectiveness of this research

For an enterprise IT system, which is required to provide reliability and stability, "capacity planning" that fits system capacity to peak load has been generally applied. However, a system configuration fitted to instantaneous peak load may require a redundant amount of resources. An example of write data generated at a real enterprise storage system is shown in **Fig. 1**. This is an example of write data workload observed in the relational database system. The vertical axis shows an amount of sequential write data monitored at external storage system connected to the database server. In this example, an instantaneous workload is generated around 6:00, but a low-workload state continued during the periods before and after that time. If the system has been designed to fit the peak in this situation, redundant system resources will be installed and are not utilized at times other than 6:00. That would result in the waste of resources. If an enterprise can accept the risk of performance degradation at the peak time only, it would be possible to save expenses.

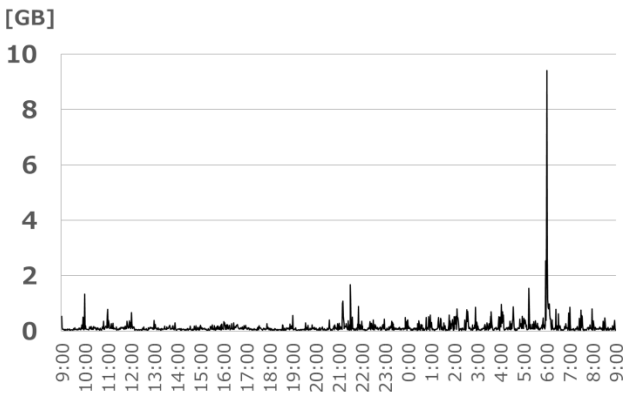

Fig. 1 An example of write workload in a storage system.

2. Cloud-based data-protection system

2.1 Cloud-storage gateway

A "cloud storage gateway" service is available for backing up data generated at the on-premises site. For example, Amazon Web Services (AWS) have released their AWS Storage Gateway service [18]. As illustrated in **Fig. 2**, this service is a gateway function installed on the on-premises side that transfers data sequentially to online cloud storage, AWS S3. Storage Gateway is a software function offered by AWS but installed at an enterprise's on-premises datacenter. Two operation modes are available: "cached volume mode" (in which primary data is placed in cloud) and "stored volume mode" (in which primary data is placed on-premises). In this research, we focus on a "cached-volume mode" architecture because it provides a smaller delay in data transfer.

In cached-volume mode, a gateway server supplies storage resources originally deployed on the cloud as a virtual device. An application server running on-premises connects to this virtual device via the iSCSI protocol and applies it for backup volume. The application data generated on the server is recorded in a local volume, and it is copied to a backup volume by functions such as

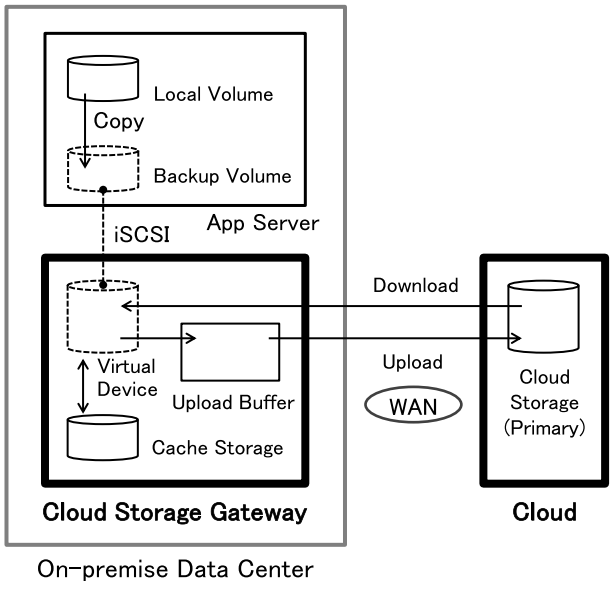

Fig. 2 A cloud-based data-protection system.

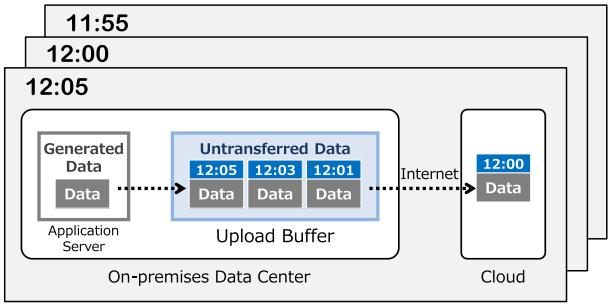

Fig. 3 An example of recovery point.

storage mirroring or backup software.

The gateway records data written on a virtual device into cache storage, and transfers them to the cloud. An upload buffer is used as a temporary storage for data to be transferred. The standby data stored in the upload buffer is transferred to the cloud via the Internet or a dedicated communication line.

2.2 Performance of data-protection system

Since the cloud gateway asynchronously transfers data temporarily stored in the upload buffer, part of the data might be lost in the event of a disaster. To guarantee a specified system performance, the recovery point must be shorter than RPO.

As illustrated in **Fig. 3** in the case data written by a server is recorded at 12:00 and reaches the cloud at 12:05 after being temporarily stored in the upload buffer or a communication delay, the recovery point at 12:05 is "5 minutes". At this point, if a failure occurs at the on-premises site, all data recorded after 12:00 is lost because it has not been transferred yet [19]. If RPO is shorter than 5 minutes, the performance target has not been achieved, so actions such as system resizing are required.

2.3 Issues concerning cloud-based data protection

To control the risk of data loss, it is required to monitor the recovery point and to check whether it has achieved RPO. If the recovery point is shorter than RPO, system performance is satisfactory for the workload. However, in general, the system-monitoring function provided by cloud services does not support a monitoring metric that corresponds to the recovery point. In the example of AWS Cloudwatch, the metric corresponding to recovery point is not supported. The time stamp as described in Fig. 3 is not implemented on the cloud storage gateway because the workload generated by the process for attaching and detaching time stamps may be an overhead. For that reason, this research tries to make it possible to calculate the recovery point by using other available metrics provided by standard monitoring services that do not require a special implementation like a time stamp.

Also, if system performance is too high or too low to guarantee RPO, system configuration should be properly reconsidered. To achieve both the performance objective and the cost minimization, the amount of system resources (such as network bandwidth) should be adjusted for appropriate size. However, a method for estimating the recovery point after the amount of system resources has been modified has not been developed. That situation is the motivation of this research to simulate the behavior of a data-protection system in order to estimate the recovery point.

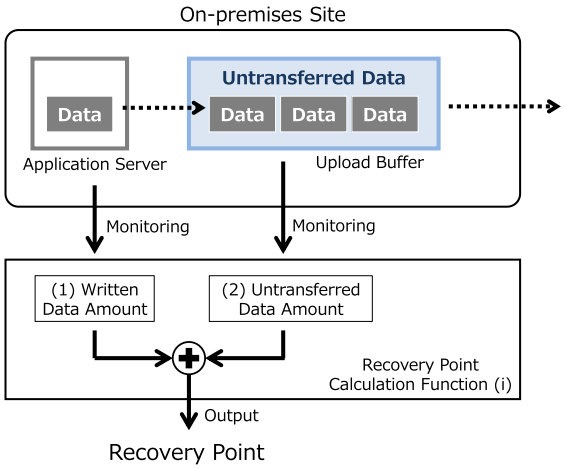

Fig. 4 An approach for performance assessment.

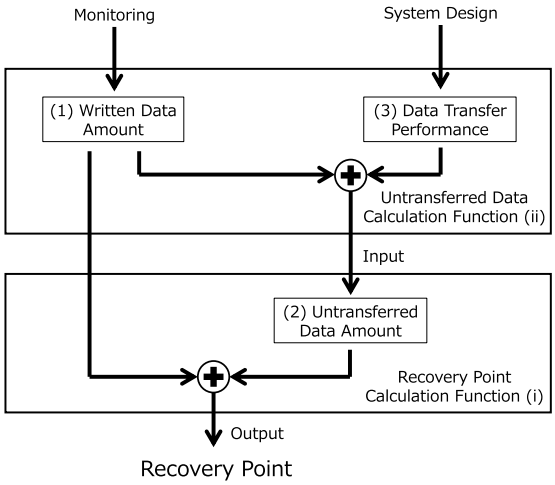

Fig. 5 An approach for performance simulation.

2.4 Approach for managing data-protection system

(1) **Performance assessment by estimating recovery point**

To evaluate system performance, a procedure for estimating the recovery point is defined as follows. As described in **Fig. 4**, since the recovery point cannot be measured directly through a monitoring service, two available performance metrics are used as the input of this estimation: written-data amount and untransferred-data amount. The recovery point output by this procedure is compared to RPO to check whether it satisfies the performance objective.

(2) **Performance Simulation for System Sizing**

To estimate an appropriate amount of system resources, a simulator that calculates recovery point is defined. As shown in **Fig. 5**, in this simulation, a formula that estimates the amount of untransferred-data is defined. As defined above (1), it is possible to estimate the recovery point by using the input composed of the untransferred-data and written-data amounts.

An example in which six patterns of system configuration with different network performance are simulated is shown in **Fig. 6**. This example shows that the recovery point gets shorter in proportion to increasing network performance, but it will also be a cause of additional cost. Therefore, among the patterns under which the

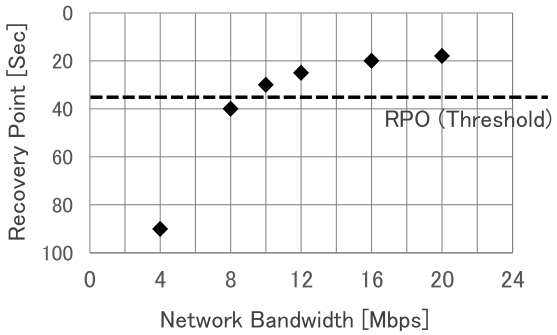

Fig. 6 An example of simulated performance.

recovery point achieves RPO, a configuration that minimizes cost is adopted as the optimum. In this case, a system designed on the basis of 10 Mbps network bandwidth that achieves RPO at minimum expense of network resource consumption should be selected as the optimum configuration.

All variable resource parameters should be considered in designing a system configuration. As well as a network resource, capacity of buffer storage and number of CPU cores for a storage gateway can be variable parameters during system configuration design.

3. System-performance assessment

3.1 System modeling

A system model for simulating the behavior of a cloud storage gateway is an abstract representation of system configuration and the process for data transfer. Although a cloud and cloud-based data protection system have various components, in this research, the components necessary for estimating the recovery point are focused on here, and a system architecture composed of those components only is described in **Fig. 7**. A cloud storage gateway is installed at the on-premises site, a cloud service provides online storage, and they are connected by a cloud-connecting network. The storage gateway provides a backup volume, cache memory, and an upload buffer. The backup volume is a storage resource that is writable from the server. The cache memory is simply used for temporarily storing data written to the backup volume. The upload buffer is also a temporary storage that is used only for data to be transferred to the cloud. These storage components can be implemented as a general medium such as volatile memory, flash memory or a hard-disk drive. The cloud service delivers online storage for the backup data pool.

The storage gateway stores data written on the backup volume (step 1) into the cache memory (step 2). Thereafter, the cached data are copied to the upload buffer in a time series (step 3) and sent to the cloud via the cloud-connecting network in write order (step 4). In the cloud, the received data are stored in the local storage (step 5). According to this model, the gap between the time at which data is written and stored in cache (step 2) and the time at which data is transferred and stored in the cloud (step 5) corresponds to the recovery point.

3.2 Estimation of recovery point

As described in Fig. 4 and Fig. 5, in this research, a method for estimating the recovery point from an input of untransferred data

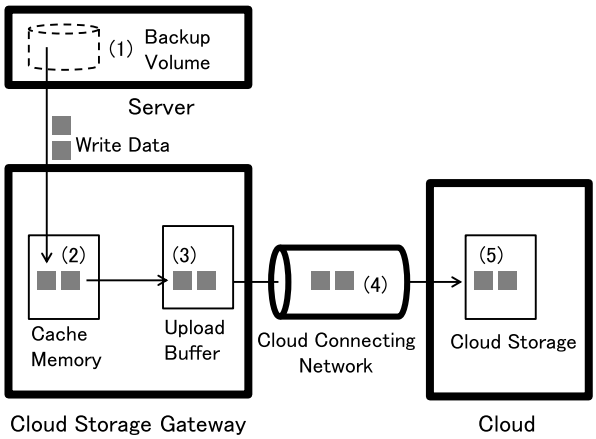

Fig. 7 A model of cloud-based data-protection system.

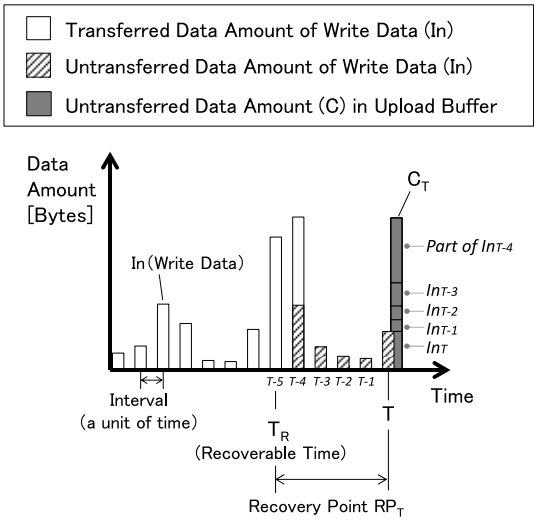

Fig. 8 Method for counting recovery points.

pooled in the upload buffer is proposed. Further, as illustrated in Fig. 3, it is supposed that the time of the oldest write data in the upload buffer approximates the time of latest data recorded on the cloud storage. Among the untransferred data, the oldest data itself is lost; however, the data written up to the preceding time of the oldest data in the buffer can be recovered because they have already been transferred to the cloud. Therefore, to identify the recovery point, the written time of the oldest data among the untransferred data stored in the upload buffer is determined.

A formula for finding the oldest data is defined as follows. Two types of time-series data are used as the inputs of this formula: untransferred data (C) and amount of written data (In). The write time of the oldest data coincides with the time at which the value obtained by cumulating In reaches the untransferred data amount, C_T, by tracing back from time T. That is, if the time at which the cumulative value of the hatched portions in **Fig. 8** reaches C_T at *T-4*, the immediately preceding time *T-5* is recoverable time T_R. Recovery point RP_T is the difference between T and T_R. These procedures are formulated as follows:

$$RP_T = T - T_R = \begin{cases} 0, & C_{\mathrm{T}} = 0 \\ (n+1) \times Interval, & C_{\mathrm{T}} > 0 \end{cases}$$

where n is the smallest integer satisfying

$$C_T \leq \sum_{i=0}^{n} In_{T-i}$$

The static parameter *Interval* of this formula is a unit of time, which corresponds to the sampling interval of a time-series system log parameter such as *In* and *C*. And *n* is the number of times *In* has been accumulated backward from time *T*, and the immediately preceding time of T is the recoverable time T_R.

3.3 Assessment of performance

With recovery point RP_T estimated according to the above procedure, it is possible to judge whether RPO is achieved or not. If the maximum value of the recovery point during the time range of system log is longer than RPO, the performance target has not been reached, and it is desirable to achieve the performance objective by the expansion of system resources. On the contrary, if it is significantly shorter than RPO, it is expected to lower the system operation cost by allowing some resources to be discarded.

4. System sizing

4.1 Simulation of system performance

As depicted in Fig. 6, a method for estimating the recovery point is defined under certain assumptions concerning system configuration. Also, as illustrated in Fig. 5, the amount of untransferred data *C* is used as the input of the recovery point estimation. Therefore, a formula to simulate the amount of untransferred data *C* is defined as follows.

$$C_T = C_{T-1} + In_T - Out_T$$

Where the amount of data written to backup volume at time *T* is denoted as In_T, and the amount of data transferred to the cloud is denoted as Out_T.

As described in **Fig. 9**. This equation means that the untransferred data amount C_T increases if the amount of transferred data Out_T, which is dependent on the performance capability of the cloud-storage gateway and cloud-connecting network, is insufficient with respect to written data amount In_T. Here, C_{T-1} represents the untransferred data amount at time *T-1*, which is one time unit before the time *T*.

According to this formula, the amount of untransferred data varies according to the performance of data transfer. Among the data to transfer at time *T*, the amount of data that can be transferred is limited by the performance. The rest of the data remains in the buffer under untransferred status at time *T*.

In the system model defined in Fig. 7, it is assumed that the data written to the storage gateway is stored in the upload buffer for a certain period of time and then sent to the cloud at once. If this standby period is defined as static value *D*, the amount of data to be transferred at time *T* corresponds to the untransferred data amount at time $T-D$. On the other hand, the transfer performance matches the lowest IO performance of upload buffer, P_{Buffer_IO}, and network performance for cloud-data transfer, $P_{Network}$. Under the above conditions, transfer-data amount Out_T is formulated as

$$Out_T = \min\left\{C_{T-D}, P_{Buffer\_IO}, P_{Network}\right\}$$

$P_{Network}$ should be considered with variables that impacts the transmission performance degradation. For example, if a cloud connecting network of Fig. 7 is implemented by public Internet, the actual communication speed is usually less than the catalogue specification of network bandwidth. The communication performance degradation is caused by other users' traffic because it is a "best effort" service. So $P_{Network}$ is defined with the transmission efficiency variable as follows:

$$P_{Network} = Bandwidth \times Transmission\_Efficiency$$

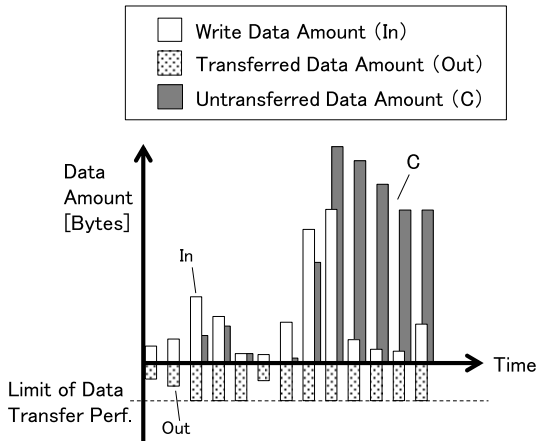

Fig. 9 Example of accumulation of untransferred data.

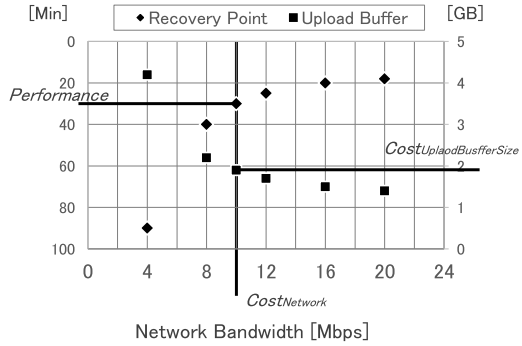

Fig. 10 Relationship between cost and performance.

4.2 Appropriate system size

The above-described simulation process makes it possible to calculate the amount of untransferred data for variously assumed system configurations and to predict the recovery point. The optimum system size among them achieves RPO at the lowest system resource expense. In this study, the bandwidth of the cloud-connecting network and the capacity of the upload-buffer storage were chosen as variables for configuring a cloud-based data protection system. The amount of untransferred data increases or decreases according to data-transfer network bandwidth. Therefore, storage capacity of the upload buffer should also be considered as a factor determining system cost. As shown in **Fig. 10**, the network-bandwidth parameter and recovery point are proportional, and the bandwidth and required size of buffer capacity are inversely proportional. For example, in the case that network bandwidth is set to 10 Mbps, the recovery point is estimated as 30 minutes, and the maximum untransferred data amount is about 2 GB. In this case, it is required to install 2 GB storage in the upload buffer. Thus, $Cost_{UploadBufferSize}$, which is one of cost vari-

ables, is also an output of performance simulation.

The total cost of system resource is a sum of the cost of the components that comprise the entire system. In this study, a variable part of the system cost is defined as the following formula:

$$Cost_{Variable} = Cost_{Network} + Cost_{UploadBufferSize}$$

An appropriate system size is one that achieves RPO and minimizes the system cost calculated by this equation.

5. Evaluation

5.1 Experiment on assessment of system performance

(1) **Experimental conditions**

The workload on an actual enterprise IT system was experimentally reproduced. The original IT system runs an enterprise web service application and installed with a transaction database that records transaction histories. The I/O performance-benchmark tool "fio" was used for generating write workload. It did not reproduce the contents of write data; instead, it reproduced the amount of data. As described in Fig. 8, a proposed method focuses not on the number of write commands but on the amount of write data, so in this experiment, not random write records but sequential write records are reproduced. A time-series transition of the amount of written data measured at a sampling rate of five-minute intervals is shown in **Fig. 11**. According to the time counting method of Fig. 8, the recovery point is estimated in a multiple of time interval; thus, in this experiment, it is calculated in a unit of five minutes. The data-production system implemented at the on-premises site was connected to the internet. Amazon Web Services was used for the cloud that is the target of data transfer. AWS Storage Gateway was installed and set to cached-volume mode. The storage gateway was installed on a virtual machine, and upload buffer was implemented with hard-disk drives. Also, to simulate the upper limit of network bandwidth, the output performance parameter of the storage gateway was tuned to 10 Mbps.

The relationships between the metrics in Cloudwatch and the parameters defined in this research are listed in **Table 1**.

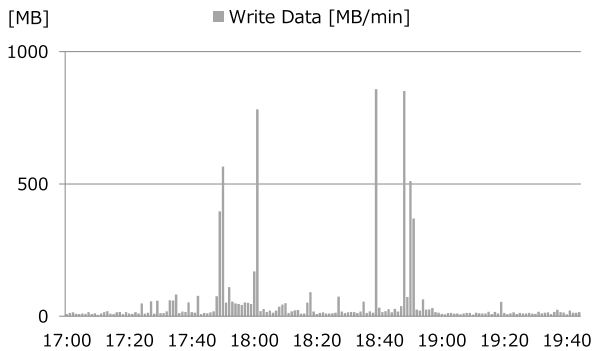

Fig. 11 Amount of write data.

Table 1 Relationships between parameters.

#	Performance metric	This research	AWS Cloudwatch
1	Write-data amount	*In*	*WriteBytes*
2	Transferred-data amount	*Out*	*CloudBytesUploaded*
3	Untransferred-data amount	*C*	*QueuedWrites*

(2) **Results**

The experimental results are shown in **Fig. 12**. The *Writebytes* is a measured value of the write workload of Fig. 11. The *CloudBytesUploaded* parameter (corresponding to transfer-data amount *Out*) was monitored as less than 352 MB per 5 minutes. This means that the upper limit of data-transfer performance of the storage gateway was 94% of 10 Mbps, namely,

$$10\,\text{Mbps} \times 0.94 \div 8\,\text{bit} \times 60\,\text{sec} \times 5\,\text{min} = 352.5\,\text{MB}/5\,\text{min}$$

It is considered that *Transmission_Efficiency* is 0.94 so that the actual limit of data transfer performance is 6% lower than the bandwidth parameter setting of the cloud gateway. The measured *QueuedWrites* parameter (which corresponds to untransferred-data amount *C*) shows how untransferred data is accumulated in the upload buffer due to write-data amount *WriteBytes* exceeding the upper limit of transfer performance. In this experiment, it was found that amount of untransferred data exceeded 2 GB at the peak at 18:50, and all of the 2 GB data will be lost if trouble occurs at on-premises site.

Calculation results for the recovery point are shown in **Fig. 13**. Measured *QueuedWrites* in Fig. 12 was used as the input of the calculation. According to the calculation result, the maximum recovery point within the inspection time range is "35 minutes". Thus, if RPO is longer than 35 minutes, it is concluded that required system performance was achieved. However, if it is less than 35 minutes, the performance requirement was not achieved, so it is necessary to extend some system resources such as network bandwidth to enhance system performance.

5.2 Consideration of system sizing

(1) **Simulation of the amount of buffered data**

In this section, we verify the accuracy of performance simu-

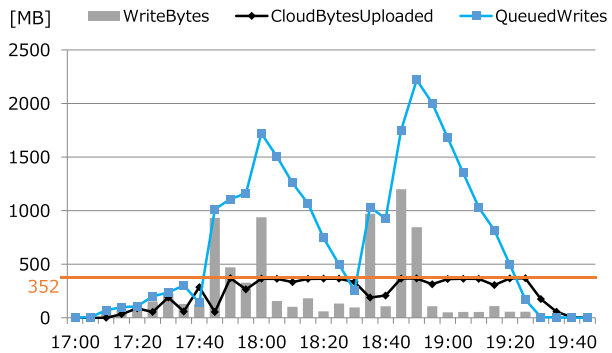

Fig. 12 Results for performance of storage gateway.

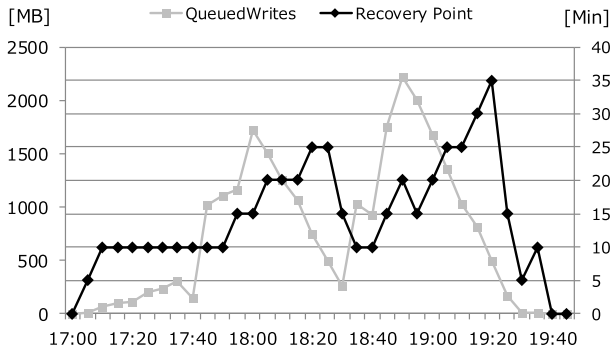

Fig. 13 Result of counting recovery points.

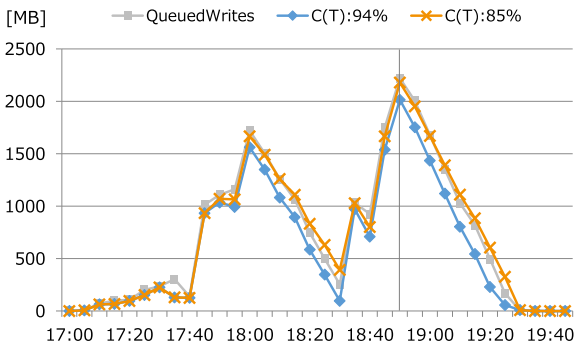

Fig. 14 Results of simulation of untransferred data.

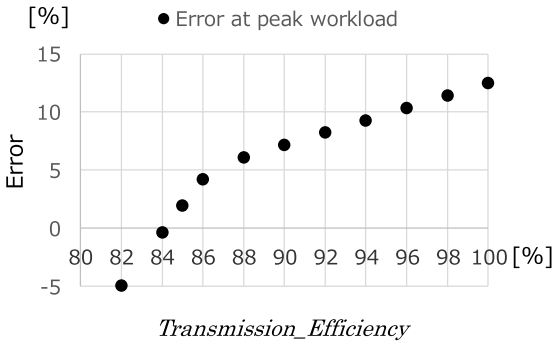

Fig. 15 An adjustment of *Transmission_Efficiency*.

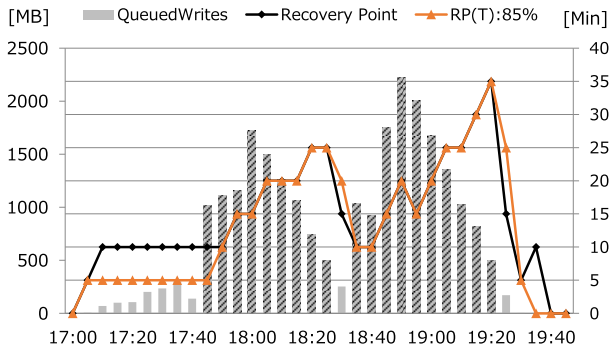

Fig. 16 Results of simulation of recovery point.

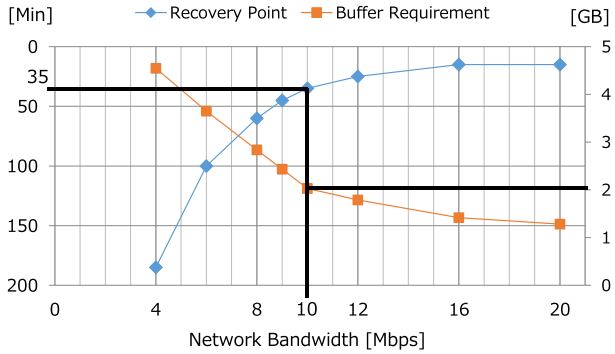

Fig. 17 Estimation performance of system configurations.

lation. The verification is conducted by a comparison between a simulation result and experimentally monitored value of untransffered data. First, the amount of untransferred data was calculated under the condition that the system performance is assumed to be the same as the experimentally measured performance. In this condition, network performance, $P_{Network}$, was set to 94% of 10 Mbps as same as the actual system performance observed in experiment above. Standby time D (i.e., a parameter that depends on the specification of the storage gateway) at which data stay in the upload buffer was set to 5 minutes. The simulation results for untransferred-data amount are shown in **Fig. 14**. According to this result, the gap between the actual measured data amount and the simulated amount at peak time 18:50 was less than 9.3%.

Furthermore, in the case that the transmission efficiency was adjusted to 85% instead of 94%, the gap at peak time could be minimized to 1.9%. **Figure 15** shows a relationship between *Transmission_Efficiency* and a gap between monitored value and simulation result. At this time, a calculation process to estimate *Transmission_Effiency* parameter that achieves a smallest gap is not established. However it is estimated that the accuracy of the proposed simulation process can be improved by setting this parameter in accurate.

The recovery point calculated under communication efficiency set to 85% is plotted in **Fig. 16**. The gap between the measurement and calculation results is small enough, and the maximum recovery point could be reproducible by the simulation. There are no recovery point errors from 17:45 to 19:20, except for 17:45, at which untransferred data more than 300 MB is accumulated in the upload buffer, as described with shaded line. It is thus concluded that this simulation result is sufficiency accurate.

On the other hand, errors from five minutes to ten minutes occurred at all times during the section in which the amount of untransferred data is less than 300 MB. Although it is possible that the storage gateway suppressed data transfer when the amount of write data were small, that possibility was unverified in this research.

(2) **System sizing**

To verify the effectiveness of the system sizing, some simulation were conducted. Eight patterns of network bandwidth, from 4 to 20 Mbps, were assumed as shown in **Fig. 17**. Since the amount of untransferred data remaining in the upload buffer increases as network bandwidth decreases, smaller bandwidth makes the recovery point longer. It is thus required to prepare a larger storage capacity for the upload buffer. In this study, recovery points are calculated under the condition of buffer residence of five minutes and transmission efficiency of 85% (as same as assumed in Section 5.2 (1)).

For example, as plotted in Fig. 17, for a configuration in which the bandwidth is 10 Mbps, the recovery point was calculated as 35 minutes, and the required storage capacity for the upload buffer was 2.02 GB. In this case, the variable system cost is the sum of the communication cost for 10 Mbps and the storage cost for 2.02 GB. It is possible to detect appropriate system size by choosing the smallest cost from the simulated system configurations that achieve RPO.

Next, we consider the cost reduction effect at the assumption in which RPO is set to 30 minutes. As described in Fig. 17, in order to achieve this RPO, it is required to install 12 Mbps of bandwidth (recovery point 25 minutes). On the other hand, a peak workload of write data amount in Fig. 11 was 858 MB/min. It is equivalent to 111.7 Mbps. As shown in **Fig. 18**, a system that is designed to fit the peak workload without applying the proposed method will be equipped with 111.7 Mbps of bandwidth, and it

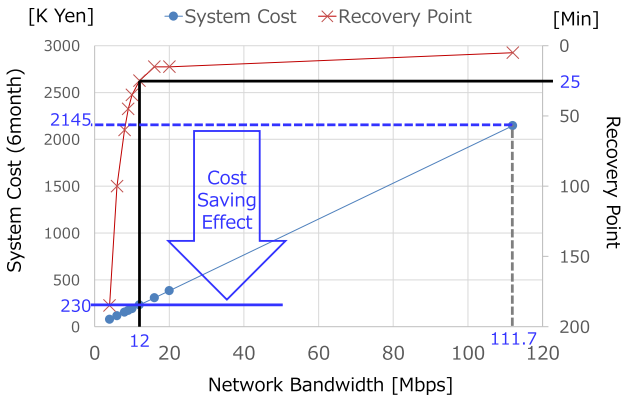

Fig. 18 Benefit of cost reduction.

Table 2 Summary of other experiments.

	# of workload pattern	1	2	3	4
a)	Feature of workload pattern				
b)	Max write data [MB/min]	858	631	140	64
c)	Standard deviation	335	112	46	11
d)	System design to fit peak load (b)				
e)	Bandwidth [Mbps]	111.7	82.3	18.3	8.4
f)	System cost [K Yen/6month]	2145	1580	351	161
g)	System design to achieve RPO 30min				
h)	Bandwidth [Mbps]	12	8	8	6
i)	System cost [K Yen/6month]	230	154	154	115
j)	Cost saving rate [%] {(f)-(i)}/(f)*100	89	90	56	28

costs 2,145,000 yen for 6months. On the contrary, a system with 12 Mbps bandwidth that is designed by a system sizing method is expected to reduce the cost by 89% in 6 months. In this calculation, we assumed 320,000 yen per 100 Mbps network, and 72 yen per 1 GB of SSD for upload buffer capacity.

Table 2 is a summary of the system sizing experiments for the workload of Fig. 11 and other three-patterns. The larger the load fluctuation (the maximum value and the deviation of write data amount) is, the greater cost reduction effect is obtained. Also, a proposed method is applicable to various situations for general purpose.

6. Next steps

The accuracy to calculating Out_T can be improved by considering the effect of data compression such as deduction of white space [20], [21]. If the data stored in the upload buffer is overwritten on the same address, it is possible to increase transfer efficiency by sending only the latest data instead of all data. Under the assumption that the compression rate of data to be transferred at time T is represented as Z_T, the data amount before compression that can be transferred coincides with the value obtained by dividing the performance limit by Z_T. The verification of the data compression effect is one of the future tasks.

The proposed methods for estimating the recovery point and calculating untransferred-data amount can be applied to various applications other than data backup. For example, with the spread of the IoT, Internet of Things, it is expected that a huge number of devices will be connected to networks [22], [23]. Many types of IoT devices, such as factory machines, cars, and monitoring cameras, have become available. A common feature of IoT devices is that they generate and transfer data to a datacenter. In some IoT applications, a delay in data transfer can cause a problem such as errors in an abnormal detection or a real-time failure-notification. Therefore, it is required to assess performance and optimize system size so as to achieve the required data-transfer performance. Since this research estimates performance based on the amount of written data and system-resource size, it is an advantage that can be applied to general-purpose data-transfer systems without special implementation such as an additional time stamp in the network protocol. Thus, one of our future tasks is to expand the system performance assessment and sizing into the IoT field.

In addition, although this research defined a system model by a deep understanding of system architecture and behavior, it may be possible that a model can be defined by an inductive approach such as machine learning if a sufficient amount of system-activity data is available.

The limitation of this research is that the measurement and assessment of the recovery point is processed in the unit of system-monitoring interval. Usually, the interval is set in minutes or seconds. However, that time axis is very different from that of computing processes executed in microseconds or nanoseconds. This is a common issue concerning system management not only for data protection but also for general enterprise IT systems with monitoring intervals in the order of minutes or seconds.

7. Related research

There are some earlier works on data protection system planning. A reference [12] proposed a method to choose an appropriate disaster recovery solution that is based on the consideration of a risk of data loss and system cost. It focused on a very high level view of disaster recovery planning on the nature of replication (e.g., synchronous and asynchronous) and storage system architecture. On the other hand, reference [15], [16] proposed a method to determine a more specific system design at initial planning phase. In this approach, a process to select a data-copy method from the server virtualization, database and storage controller was shown. This approach requires some structured knowledge that is specific to each implementation in advance of system design.

A proposed method in this paper does not require any special knowledge because it is independent from products and services. This general purpose method requires a few standard monitoring metrics only, so it can be applied to various type of data protection system. Also, the estimation of network bandwidth proposed in this research is out of scope of earlier researches.

In a previous research, we proposed a method for system assessment and sizing during disaster recovery that replicates data among storage systems [24], [25], [26]. As for that system, it was not necessary to consider “a stand-by period D” in this research that is caused by the implementation of data transfer on the sender side, because the remote copy function of the storage controller attempts to transmit the written data as soon as possible [19], [20], [21]. Therefore, for the procedure for calculating untransferred data amount proposed in this research, we considered the behavior of buffering written data for a certain period of

time at the cloud storage gateway. With the consideration of D, it became possible to apply the performance simulation to the cloud based system.

8. Conclusion

A method for assessing system performance of data-protection targeting the cloud and a method for resizing system resources were proposed. For this assessment, a model of data protection system architecture and data transfer processes was defined. This model was used to estimate the recovery point to verify whether the system configuration to be simulated achieved RPO or not. Furthermore, a method for simulating an appropriate system configuration that optimizes system performance and operating cost was proposed.

Accuracy and effectiveness of the proposed methods was experimentally evaluated. In the experiment, the recovery point could be estimated by inputting other metrics such as untransferred data. In the performance simulation, it was demonstrated that the gap between simulation and actual measurement of untransferred data was only 1.9% which is within the acceptable error range. Also, it is estimated that 89% of system resource expense can be reduced in compared with the situation in which data-protection system is designed to fit a peak workload.

Acknowledgments We appreciate anonymous reviewers who gave us many useful comments.

References

[1] Gantz, J. and Reinsel, D.: The Digital Universe Decade – Are You Ready?, IDC – IVIEW (2010).
[2] Data Age 2025: The Evolution of Data to Life-Critical, IDC (2017).
[3] Gupta, C., Farahat, A., Hiruta, T., et al.: Collaborative Creation with Customers for Predictive Maintenance Solutions on Hitachi IoT Platform, *Hitachi Review*, Vol.65, No.9, pp.403–409 (2016).
[4] Takeda, N., Kita, Y., Nakagawa, H., et al.: Using Operation Information in Reliability Design and Maintenance: Analytics for the IoT Era, *Hitachi Review*, Vol.65, No.9, pp.450–455 (2016).
[5] Vennelakanti, R., Sahu, A. and Dayal, U.: Winning in Oil and Gas with Big Data Analytics, Hitachi Review, Vol.65, No.2, pp.884–888 (2016).
[6] LaValle, S., Lesser, E., Shockley, R., Hopkins, M.S. and Kruschwitz, N.: Big Data, Analytics and the Path From Insights to Value, MIT Sloan Management Review (2011).
[7] Manyika, J., Chui, M., Brown, B., Bughin, J., Dobbs, R., Roxburgh, C. and Byers, A.H.: Big data: The next frontier for innovation, competition, and productivity, McKinsey Global Institute (2011).
[8] Patterson, D.A.: A Simple Way to Estimate the Cost of Downtime, *Proc. USENIX 16th System Administrators Conference* (*LISA '02*), pp.185–188 (2002).
[9] Rudolph, C.G.: Business continuation planning/disaster recovery, *IEEE Communications Magazine*, Vol.28, No.6, pp.25–28 (1990).
[10] Patterson, D.A., Gibson, G. and Katz, R.H.: A case for redundant arrays of inexpensive disks (RAID), *SIGMOD '88*: *Proc. 1988 ACM SIGMOD International Conference on Management of Data*, pp.109–116, ACM Press (1988).
[11] Toigo, J.W.: Disaster Recovery Planning, Principle Hall (2003).
[12] Keeton, K., Santos, C., Beyer, D., Chase, J. and Wilkes, J.: Designing for Disasters, *Proc. 3rd USENIX Conference on File and Storage Technologies*, pp.59–62 (2004).
[13] Yamato, J.: Storage Based Data Protection for Disaster Recovery, The Journal of the Institute of Electronics, *Information and Communication Engineers*, Vol.89, No.9, pp.801–805 (2006).
[14] Kakurai, K. and Koichio, O.: The evaluation of disaster recovery level on a renewal project, *IPSJ Technical Report*, Vol.2004, No.106, pp.1–6 (2004).
[15] Gopisetty, S., Butler, E., Jaquet, M., et al.: Automated planners for storage provisioning and disaster recovery, *IBM Journal of Research and Development*, Vol.52, No.4/5, pp.353–366 (2008).
[16] Nayak, T., Routray, R. and Singh, A.: End-to-end Disaster Recovery Planning: From Art to Science, Network Operations and Management Symposium, IEEE (2010).
[17] Amazon Web Services: Amazon CloudWatch User Guide (2017).
[18] Amazon Web Services: AWS Storage Gateway User Guide (2013).
[19] Emaru, H., Takai, Y. and Hara, J.: Monitoring Recovery Point for the Asynchronous Remote Copy in Disaster Recovery, *IPSJ SIG Technical Report*, Vol.2010-EVA-31, No.1 (2010).
[20] Hugo, P., Stephen, M., Mike, F., et al.: SnapMirror: File System Based Asynchronous Mirroring for Disaster Recovery, *Proc. USENIX Conference on File and Storage Technologies* (2002).
[21] Philip, S., Mark, H., Grant, W., et al.: WAN Optimized Replication of Backup Datasets Using Stream-Informed Delta Compression, *Proc. USENIX Conference on File and Storage Technologies* (2012).
[22] Ministry of Internal Affairs and Communications, Japan: Information and Communications in Japan White Paper 2017 (2017).
[23] Hanaoka, S., Taguchi, Y., Nakamura, T., et al.: IoT Platform that Expands the Social Innovation Business, Hitachi Review Innovative R&D Report 2016 (2016).
[24] Maruyama, N., Taguchi, Y. and Yamamoto, M.: Proposal of the storage remote copy configuration model for Disaster Recovery System, *Proc. 70th National Convention of IPSJ*, pp.4-539–4-540 (2008).
[25] Taguchi, Y., Ichikawa, N. and Yamamoto, M.: Asynchronous Remote Copy System Resource Sizing for Disaster Recovery, *Proc. 25th Computer System Symposium* (*ComSys2013*) (2013).
[26] Taguchi, Y. and Yamamoto, M.: Asynchronous Remote Copy System Resource Sizing for Disaster Recovery, *IPSJ Trans. Advanced Computing Systems*, Vol.7 (2014).

Yuichi Taguchi received his B.E., and M.E. degrees from Waseda University in 1995 and 1997. Since 1997, he has been engaged in the research and development of IT platform and services in R&D Group, Hitachi, Ltd. He is currently with Ph.D program in the Graduate School of Informatics and Engineering University of Electro-Communications. He is a member of IPSJ.

Tsutomu Yoshinaga received his B.E., M.E., and D.E. degrees from Utsunomiya University in 1986, 1988, and 1997, respectively. From 1988 to 2000, he was a Research Associate with the Faculty of Engineering, Utsunomiya University. He was also a Visiting Researcher with the Electro-Technical Laboratory from 1997 to 1998. Since 2000, he has been with the Graduate School of Information Systems, The University of Electro-Communications, where he is currently a Professor. His research interests include computer architecture, interconnection networks, and network computing. He is a fellow of IEICE, and a member of ACM and IEEE.