Efficient Monitoring Method Considering User Utility for Networked Information Sources.

Shinji Sugawara  Kengo Honda  Yutaka Ishibashi

This paper proposes a monitoring method for information retrieval from the sources in large-scale networks, which tries to achieve the maximum gain of user utility with the minimum source observation cost. Generally, information accumulated in a network is being updated every second and the contents which have been downloaded by network users become obsolete as time passes. User utility, which is obtained by the users when they get their target information successfully, declines with the elapsed time from the instant of placing a retrieval request or the content’s renewal. Accordingly, the proposed monitoring method adjusts its observation intervals according to the information sources’ update intervals considering user utility decrease and monitoring cost caused by the observation of sources. The usefulness of the proposal is confirmed by computer simulations, which illustrate that it is effective especially in the condition that the expense of the observations is neither extremely emphasized nor neglected. In addition, a prototype of the monitoring system which implemented the proposal is developed and some monitoring experiments in the real Internet environment are conducted. The results show that the proposed method seems to be available in the case where the conditions are close to those of the simulations.

1 Introduction

With the popularization of the Internet, huge amount of information has been accumulated in the large-scale network, and the network users have come to retrieve valuable contents from there, recently. Information exchanged in the network has a wide variety, and freshness is important for some contents such as headline news and stock prices [6] [8] [5]. Because the contents accumulated in the network are occasionally being updated and the content items which have been retrieved by the users become obsolete as time passes, monitoring of information sources is needed for the users in order to keep freshness and accuracy of their retrieved and possessing content items [3] [4] [18].

Although a content item published by a source continues to be updated anytime it needs in general, if all sources’ update intervals are unknown, monitoring the sources with a regular interval is shown to be effective [9] [10] [1]. However, if monitoring interval is shorter than the actual update intervals, even granting that the information freshness can be kept, futile monitoring costs accrue. On the other hand, if monitoring interval is longer than the actual update intervals, even supposing that the futile monitoring costs decrease, possessing information tends to become obsolete. Therefore, monitoring the appropriate source at appropriate time is important, however this kind of discussion has not done yet sufficiently.

In addition, the importance of each content item is generally different for different users. If a user repeats retrieving an interesting content item continuously from its source in quest of its high freshness and accuracy, the user may have enough satisfaction; however, if the user retrieves an uninteresting content item from its source at the same expense, his/her satisfaction level may be relatively not so high. Hence, it is needed for the users to execute prioritized monitoring according to their preference level of content items in order to get their satisfac-
In this paper, we propose an efficient monitoring method for information sources, preliminarily intended to be monitored in a large-scale network; the concept of this method is to try to maximize user’s satisfaction for information retrieval by reducing futile monitoring costs and prioritizing searches of sources according to their content’s importance. In this context, we assume that user’s satisfaction varies according to the acquisition situation of the content item, however, the content’s importance for the user is constant or hard to change, and unique to him/her. The effectiveness of this method is evaluated by computer simulations. In addition, we implement the method on a prototype monitoring system actually working on the Internet, and evaluate the effectiveness of the method in a real environment.

The rest of this paper is organized as follows. In Section 2, a basic concept of information monitoring is explained briefly, user utility is outlined, and then, the assumptions for discussion in this paper are described. An efficient monitoring method is proposed in Section 3, and its evaluation by using computer simulations is illustrated in Section 4. In addition, the implemented monitoring system working in the Internet environment and its evaluation are shown in Section 5. Finally, conclusion and future works are presented in Section 6.

This paper was written based on [15], as a complete version of this study.

2 Basic concept of this study

2.1 Monitoring of information sources

If we monitor some information sources in a large-scale network, we need a continual access to the intended sources in order to get updated content items in case of their renewals. However, if a user should do all this task for a lot of sources by him/herself, it is too heavy to be reasonable, and keeping appropriate monitoring schedule is difficult. Then in this paper, we assume to use an agent which monitors the intended sources automatically. A basic concept of the information monitoring is shown in Fig. 1.

In Fig. 1, each source has a content item which is open to any of network users and updates it independently. When the content is updated, old (not newest) one is discarded. In this environment, a network user retrieves content items from the sources with the use of an agent. If the item brought by the agent involves something novel for the user, he/she keeps it in the storage as a part of his/her preserved current knowledge. In this situation, the agent needs to repeat accesses to each of the intended sources continuously while the user needs to keep the content item up to date. The accesses for checking if a content item is updated are called observations, and a monitoring of an information source consists of routine observations and content retrievals in case of the updates, in this paper.

2.2 User utility

If we assume that a user will have a certain satisfaction when he/she retrieves a new or renewed content item from a source, and also assume that initially expected satisfaction decreases according to the required cost of retrieval such as elapsed time, this is very similar to the concept of user utility. User utility means the user’s satisfaction with a service in general, which decreases monotonically with the increase of consumed time or cost [20][19][16]. In this paper, we consider the service is to get target content item from sources in a large-scale network, and user utility is a user’s satisfaction which decreases according to the elapsed time from the instant when the content item becomes open to any of the network users.

The relationship between service time and user utility is said to be expressed with power functions or exponential functions [13], and these functions are often used in practice to estimate user utility from user’s waiting time.

When using a power function, user utility $g(t)$ is expressed as follows.

$$g(t) = Dt^{-k}$$

(1)
Here, $k (0 < k < 1)$ is a degree how far user utility depends on the waiting time, $D$ is a constant, and $t$ denotes the waiting time. If $k$ is close to 1, user’s satisfaction with the service depends strongly on the waiting time; on the contrary, when $k$ is close to 0, the satisfaction is almost constant regardless of the waiting time.

If using an exponential function instead of a power function, user utility $h(t)$ is expressed as follows.

$$h(t) = De^{-tk} \quad (2)$$

Information becomes obsolete as time passes. Accordingly, user’s satisfaction in the content retrieval decreases as time passes, too. In this paper, we consider user utility to be the function of the same sort, and we recognize user utility means user’s satisfaction when the user gets a content item he/she wants. That is, user utility is a function of the elapsed time from the update of the content item, and accordingly, the utility function monotonically decreases along with time.

Just in order to maximize the user utility, access or polling all the intended information sources as frequently as possible is workable. However, because update frequency differs from one source to another and access to the un-renewed sources causes wasted expense, it is ideal that access frequency to each source is kept to be necessary and sufficient value. Supposing that all accesses to the sources are done by an agent, the user’s waiting time for the each access itself must be very short. On the other hand, in the case of accesses by the agent at a constant frequency, time difference between an update of a source and the agent’s discovery of the update accrues according to the length of the access interval and the actual time of the update. It is difficult to avoid the user utility decrease caused by the time difference, and therefore, we try to minimize the difference in this paper.

2.3 Assumptions

As the requirements of the information sources monitoring in a large-scale network, we assume the following conditions.

- The number of information sources intended for this monitoring stays constant for simplicity of the following discussions.
- Each source has a single content item which is open to any of the network users.
- Content on each source is updated freely and voluntarily by the owner of the source, and the updates are independent from the intention of the user who monitors the source.
- Each content item published in its information source $i$ has a degree of importance to the user, $v_i$.
- The agent can get content items from plural sources at the same time in one unit time of monitoring.
- When retrieving a content item, the agent can know its last update time.
- When the agent obtains a new or renewed target content item successfully, the user gets user utility.
- User utility of a content item for the user becomes its maximum value when the item is just updated, and then, decreases as time passes.
- Each content item gives a different user utility function depending on the user.
- The average of the maximum user utility concerning each of the content items can be estimated by the user.

With these assumptions above, we propose and evaluate a monitoring method which tries to achieve largest user’s profit totally. In this paper, we define user’s profit $P_i$ as follows, which is the sum of gained user utility subtracted the monitoring cost which is proportional to the times of observations (proportional constant or weight: $\alpha$) during the period of 100 unit times, and we use this value as the measure for evaluation.

$$P_i = \left( \sum_{n=1}^{O_i} u_i(n) - \alpha O_i \right) \times 100 \quad (3)$$

$u_i(n)$: user utility obtained at $n$-th observation on source $i$

$O_i$: total number of observation on source $i$

$T$: total elapsed unit time of monitoring

$\alpha$: proportional constant or weight

---

$^1$ $k$ can be also set to a value which is equal to or more than 1 theoretically. However, it is assumed to be the value between 0 and 1 actually in this paper for simplicity, because of avoiding the extreme utility function values.
After the calculation of $P_i$ for all intended sources \((0 \leq i \leq N, N: \text{number of sources to be monitored})\), we evaluate and discuss the effectiveness of the proposed monitoring method by $P$, which is the average value of $P_i$. Because each $P_i$ is an evaluation result derived from a short period of monitoring, and varies widely according to conditions.

### 3 Proposed method

In this paper, we propose a monitoring method which tries to fit appropriate observation interval dynamically for each source in order to get large user’s profit. The main procedure of the method is illustrated as follows, and its flow diagram is shown in Fig. 2. (Note that this method is applied in each source in parallel for the monitoring of its possessing content item.)

<Main procedure>

1. Run the Default observation interval setting process. (described later)
2. Conduct $M$ times observations.
3. In the case where the current number of observations is smaller than $M'$ times, go to 2. Otherwise, calculate $m_i$ and $m_i'$.

   \[
   m_i : \text{number of updates detected in the last } M \text{ times observations in source } i \\
   m_i' : \text{number of updates detected in the last } M' \text{ times observations in source } i \\
   M, M' : \text{constant } (M < M')
   \]

4. In the case where $I_i \neq I_{\text{min}}$ and $m_i' = M'$, or in the case of $m_i' = 0$, go to 1.

   \[
   I_i : \text{observation interval of source } i \\
   I_{\text{min}}: \text{minimum observation interval (constant)}
   \]

5. In the case of $m_i = M$, run the Interval shortening process. Otherwise, run the Interval extension process.
6. Go to 2.

<Main procedure ends>

We show details of three processes in the above main procedure ((a) Default observation interval setting process, (b) Interval shortening process and (c) Interval extension process) as follows.

![Flow diagram of proposed method.](image)

(a) Default observation interval setting process

1. Set the observation interval $I_i$ for source $i$, $t_{\text{bef}}$ and $c_i$ by (4), (5) and (6).

   \[
   I_i \leftarrow I_{\text{min}} \quad (4) \\
   t_{\text{bef}} \leftarrow t_{\text{cur}} \quad (5) \\
   c_i \leftarrow 0 \quad (6)
   \]

   \[t_{\text{bef}} : \text{recorded time}\]

   \[t_{\text{cur}} : \text{current time (At the beginning of a monitoring, } t_{\text{cur}} \text{ is initialized to be zero. Every time on observation is finished, } t_{\text{cur}} \text{ increases by } I_i).\]

   \[c_i : \text{number of observations at source } i\]

2. Change the observation interval according to the following two operations alternatively, and conduct one observation.

   (i) In the case where update is detected:

   \[
   I_i \leftarrow I_{\text{min}} \quad (7)
   \]
\[ c_i \leftarrow c_i + 1 \quad (8) \]

(ii) Otherwise:
\[ I_i \leftarrow 2 \cdot I_i \quad (9) \]

3. In the case of \( c_i = M_{\text{first}} \), \( I_i \) is revised by (10) and end this process. Otherwise, go to 2.
\[ I_i \leftarrow \frac{t_{\text{cur}} - t_{\text{bef}}}{M_{\text{first}}} \quad (10) \]

\( M_{\text{first}} \): constant integer

\( (\text{end}) \)

(b) Interval shortening process

1. The elapsed time for \( M \) times observations with the interval \( I_i \) is phrased as base-time for source \( i \).
2. Calculate the number of observations \( M'' \) in the case of using the observation interval \((I_i - \Delta I)\) during the base-time by (11).
\[ M'' \leftarrow \left\lfloor \frac{I_i}{I_i - \Delta M} \right\rfloor \quad (11) \]
\( \Delta I \): observation interval change \((\Delta I > 0)\)

3. Calculate user’s profit \( P_i \) by user utility derived from the observation result of last \( M \) times by (12).
\[ P_i \leftarrow \sum_{j=0}^{M-1} u_i(O_i' - j) - \alpha M \quad (12) \]
\( O_i' \): total number of observations on source \( i \) so far \((O_i' \geq M)\)

4. Calculate linearly approximated user utility at \( n \)-th observation on source \( i \), \( U^\text{app}_i(n) \) by (13).
\[ U^\text{app}_i(n) \leftarrow \left( \frac{\sum_{j=0}^{M-1} u_i(O_i' - j) - t_i^\text{ave}}{\sum_{j=0}^{M-1} F(t(O_i' - j))} \right) t(n) + u_i^\text{ave} \quad (13) \]
\( t(n) \): elapsed time from the last information update at the time of the \( n \)-th observation
\( F(x) \): a function of elapsed time \( x \)
\( u_i^\text{ave} \): estimated average of maximum user utility on source \( i \)

Note that it is assumed to be impossible to retrieve a content item from a source at just the same time of the item’s update, thus \( \sum_{j=0}^{M-1} F(t(O_i' - j)) \) in (13) cannot be zero.

5. Estimate user utility \( u'_i(n) \) according to the following conditions by using \( U^\text{app}_i(n) \) derived in the case where \( M'' \) times observations are executed with the interval of \((I_i - \Delta I)\).

(i) If the retrieved content item has not preserved yet (new or updated content item).
\[ u'_i(n) \leftarrow U^\text{app}_i(n) \quad (14) \]

(ii) Otherwise
\[ u'_i(n) \leftarrow 0 \quad (15) \]
\( u'_i(n) \) estimated user utility obtained at \( n \)-th observation on source \( i \)

6. Estimate user’s profit \( P'_i \) by (16) which is derived in the case where \( M'' \) times observations are executed with the interval of \((I_i - \Delta I)\) by using user utility estimated above.
\[ P'_i \leftarrow \sum_{j=0}^{M''-1} u'_i(O'_i - j) - \alpha M'' \quad (16) \]

7. When the inequality (17) is fulfilled, shorten the observation interval as shown in (18).
\[ P_i < w(v_i) \cdot P'_i \quad (17) \]
\[ I_i \leftarrow I_i - \Delta I \quad (18) \]
\( w(v_i) \): weight which is positively correlated with source importance \( v_i \)

(c) Interval extension process

1. Definition of the base-time for source \( i \) is just the same with that of the Interval shortening process.
2. Calculate the number of observations \( M'' \) in the case of using the observation interval \((I_i + \Delta I)\) during the base-time by (19).
\[ M'' \leftarrow \left\lfloor \frac{I_i}{I_i + \Delta M} \right\rfloor \quad (19) \]

3. Calculate user’s profit \( P_i \) by (12), in which sum of user utility is derived from the observation results of last \( M \) times.
4. Calculate linearly approximated user utility at the \( n \)-th observation on source \( i \), \( U^\text{app}_i(n) \) by (13).
5. Estimate user utility \( u'_i(n) \) which is derived in the case where \( M'' \) times observations are executed with the interval of \((I_i + \Delta I)\) by the same conditions at the Interval shortening
6. Estimate user’s profit $P'_i$ by (16) which is derived in the case where $M''$ times observations are executed with the interval of $(I_i + \Delta I)$ by using user utility estimated above.

7. When the inequality (20) is fulfilled, extend the observation interval as shown in (21).

\[ w(v_i) \cdot P_i < P'_i \]  \hspace{1cm} (20)

\[ I_i \leftarrow I_i + \Delta I \]  \hspace{1cm} (21)

\[ \text{(end)} \]

4 Evaluation

In order to evaluate the proposed method mentioned above, we conduct some computer simulations, and then, discuss its effectiveness based on the results shown below.

4.1 Simulation conditions

Specific parameters for the computer simulations are shown below in Table 1. One simulation unit time is assumed to be corresponding to one minute, and 525,600 simulation unit times corresponds to one year. The other conditions are assumed as follows.

1. The worthiness of each source $i$ from the viewpoint of the user is denoted by $v_i$, and its value is given from the set of $\{1, 2, 3, 4, 5\}$ at random.

2. User utility decreases according to the power function (22) or exponential function (23) below.

\[ u_i(n) = f(v_i)t(n)^{-k} \]  \hspace{1cm} (22)

\[ u_i(n) = f(v_i)e^{-t(n)k} \]  \hspace{1cm} (23)

$t(n)$: elapsed time from the last content update in the $n$-th observation

$k$: degree of how far user utility depends on the elapsed time

$f(v_i)$: a constant uniquely defined by a combination of a user and a content item, which reflects initial worthiness of the item for the user

3. The average of $f(v_i)$ is set according to $v_i$ in the following way. First of all, the average of $f(v_i)$ is assigned by using a value of $v_i$ according to Table 2. The average value assignment is prepared to be done in two ways, type 1 and type 2, which are typically different in relation between $v_i$ and average $f(v_i)$. Consequently, $f(v_i)$ is given according to a probability given by the normal distribution with the assigned average and variance; the variance is set to one tenth of the average. (To be precise, normal distribution is approximated by binomial distribution because every $f(v_i)$ is a discrete value.)

4. The degree how far user utility depends on the elapsed time, $k$ is decided at random within the range of $0 < k < 1$.

5. In every observation, the user obtains some utility when he/she gets the target content item. However, in the case where the obtained item is just the same as the content which has been already retrieved before, acquired utility for the observation is zero.

6. The information update on each source occurs according to a Poisson distribution. The mean number of update occurrences $\lambda$ is chosen from the set of $\{0.000099, 0.00069, 0.0028, 0.0056, 0.017, 0.03, 0.1\}$. These parameters correspond to 1 week, 1 day, 6 hours, 3 hours, 1 hour, 30 minutes and 10 minutes up-

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total simulation time (Number of simulation unit times)</td>
<td>525,600</td>
</tr>
<tr>
<td>Number of information sources</td>
<td>20</td>
</tr>
<tr>
<td>$M_{first}$ (Constant)</td>
<td>5</td>
</tr>
<tr>
<td>$M$ (Constant)</td>
<td>5</td>
</tr>
<tr>
<td>$M'$ (Constant)</td>
<td>15</td>
</tr>
<tr>
<td>Value of information source: $v_i$</td>
<td>1, 2, 3, 4, 5</td>
</tr>
<tr>
<td>Weight: $w(v_i)$</td>
<td>0.5, 0.75, 1, 1.25, 1.5</td>
</tr>
<tr>
<td>Minimum observation interval: $I_{min}$</td>
<td>1</td>
</tr>
<tr>
<td>Observation interval change: $\Delta I$</td>
<td>1</td>
</tr>
<tr>
<td>Simulation runs</td>
<td>50</td>
</tr>
</tbody>
</table>

| Table 2 Assignment of average of $f(v_i)$. |
| $v_i$ | 1 | 2 | 3 | 4 | 5 |
| Type 1 | 500 | 750 | 1,000 | 1,250 | 1,500 |
| Type 2 | 1,000 | 1,000 | 1,000 | 1,000 | 1,000 |
update intervals respectively. Frequency distributions of the number of sources with respective values of $\lambda$ are defined in three types: "rarely-updated", "moderately-updated" and "frequently-updated". These distributions’ mode values are 0.00069, 0.0056 and 0.03 respectively. Adopted frequency distributions for actual simulations are shown in Fig. 3.

7. We also assume a situation in which update frequency on each source varies periodically during a long-period simulation. In this scenario, "periodically-changing update frequency,” the distribution has two phases, "moderately-updated" and keeping $\lambda$ to be 0.03 in all sources. One of these phases appears alternately in every 86,400 unit times which corresponds to 2 months\textsuperscript{12}.

Under these conditions above, we conduct computer simulations with 7 scenarios, shown in Table 3. These scenarios are constructed by the combinations of the types of user utility, types of average $f(v_i)$ and the content update frequency. They encompass the basic simulation conditions which we assume as minimum required ones for the evaluation from those three points of view.

\begin{table}[h!]
\centering
\begin{tabular}{|c|c|c|c|}
\hline
\# & User utility function & Average of $f(v_i)$ & $\lambda$ frequency distribution \\
\hline
1 & power & type 1 & moderately \\
2 & exponential & type 1 & moderately \\
3 & power & type 2 & moderately \\
4 & exponential & type 2 & moderately \\
5 & power & type 1 & frequently \\
6 & power & type 1 & rarely \\
7 & power & type 1 & periodically-changing \\
\hline
\end{tabular}
\caption{Simulation conditions.}
\end{table}

\textsuperscript{12} In reality, each monitored site’s periodical update-frequency-change brakes out separately with different period and frequency in general. However, because the primary aim of this simulation scenario is simply showing the robustness of the proposal against the update-frequency churn of the sites, and the synchronization of all the sites’ update-frequency-change periods has no effect on the user’s profit in this simulation, this condition is adopted for simplicity.

4.2 Methods for comparison
In order to verify the efficiency of the proposed method, we adopt the following five monitoring methods for comparison.

- 10 minutes monitoring
  Observation is conducted at every 10 simulation unit times on a constant basis for all sources.

- 1 hour monitoring
  Observation is conducted at every 60 simulation unit times on a constant basis for all sources.

- 1 day monitoring
  Observation is conducted at every 1,440 simulation unit times on a constant basis for all sources.

- Random frequency monitoring
  Observation is conducted at intervals of $I_i$ for each source $i$, and $I_i$ is set at random between $I_{min}$ and 1,440, and changes every time of observations.

- Optimal solution
  Observation is always carried out immediately just after the information update on all the intended sources. In the case where the weight for the observation cost $\alpha$ is larger than the gettable user utility at the time, the observation is omitted in order to avoid a user’s profit decrease. Actually, it is impossible to carry out this monitoring on the real Internet environment, and the result is calculated on all simulation scenarios just for comparison.

4.3 Simulation results and discussion
Simulation results are shown in Fig. 4 through
10 which illustrate the relationship between weight $\alpha$ and the user’s profit $P$. First, we explain the trend of all the simulation results. The 10 minutes monitoring tends to get large user’s profit when $\alpha$ is small, because user utility is the main factor of the user’s profit in this case, and the method which observes sources frequently and gets large user utility must be efficient. On the other hand, when $\alpha$ is large, the 1 day monitoring tends to get large user’s profit because monitoring cost becomes relatively large compared to obtained user utility. In this case, the method which hardly conducts observations and constrains the monitoring cost achieves a good result. The user’s profit of the optimal solution achieves more than threefold score of the proposed method’s at $\alpha = 0$ in all simulations. In addition, the user’s profit’s decrease gradients of the optimal solution and the proposed method seem to be close. This reason is that the observation constraining tendencies of the two methods along with the transition of $\alpha$ are similar.

Next, we discuss the achievements of the proposed method comparing to those of the other monitoring methods except for the optimal solution. The proposed method accomplishes the best monitoring when $\alpha = 100$ in all the simulation results. The proposed method includes the control to alter the changeability of observation interval according to the importance of source by using judgment with inequalities (17) and (20) in Section 3.

The effectiveness of this control is shown in Fig. 9. In this figure, user’s profit values of two cases are compared: the case of using the proposed method directly with the changeability control of observation interval according to the importance of source (controlled) and the case of using the same method except for the control (no-control). The user’s profit gained by the controlled scheme is slightly larger than the no-control scheme when $\alpha < 200$. The reason why the controlled scheme achieves higher user’s profit when $\alpha$ is small is as follows. When the controlled scheme is used, the important sources holding promises for large user utilities tend to be shorten their observation intervals, and conversely, unimportant sources tend to be extended their observation intervals. If $\alpha$ is small, because observation cost is relatively small, the effect of active observations on the sources for earning large utility is larger than the drawback of rough judgments without worrying about futile observations. On the other hand, when $\alpha$ is large, the no-control scheme achieves larger user’s profit than the controlled scheme, because the controlled scheme’s futile observations influence user’s profit more seriously while the observations are expensive. Totally, the effectiveness of this control is not very large, and there is some room of improvement for it.

As shown in Figs. 4 and 5 which illustrate the results of scenarios 1 and 2, proposed method’s useful range of $\alpha$ in the case of using power functions for user utility (power situation) is wider than that in the case of using exponential functions (exponential situation). The reason is that, because the user utility expressed by power function detracts its value more slowly than that expressed by exponential function as time passes, it is easier to obtain averagely large user utility per observation in power situation than exponential situation.

Simulation results on scenarios 1 and 3, and those on 2 and 4 are very similar. The difference of the conditions between scenarios 1 and 3 or 2 and 4 is the way of giving the average of maximum user utility $f(v_i)$. These results mean that user’s profit is not seriously affected mainly by the bias of the average of maximum user utility, only if its mean value is the same. Specific results on scenarios 3 and 4 are illustrated in the appendix.

Fig. 6 is the result of the simulation on scenario 5. In this case, update is done more frequently than the case of scenario 1 shown in Fig. 4. The 10 minutes monitoring tends to achieve much better result than any other comparison methods and also somewhat better than the proposed method while $\alpha$ is small. The reason why the monitoring cost becomes relatively small, a lot of updated content items are retrieved by frequent observations, and the total user utility increases further than the case of scenario 1. The proposed method is better than the others while $\alpha$ is from 100 to 300.

Fig. 7 is the result of the simulation on scenario 6. In this case, update is done less frequently than the case of scenario 1 shown in Fig. 4. Because of no need of frequent observations, the 1 day monitoring
tends to achieve rather preferable result than the other methods in the wide range of $\alpha$. However, since the content items are rarely updated and their user utility is kept low, obtained total user utility is generally degraded further than that of scenario 1 even if using any method. On this condition, proposed method does not achieve an extraordinary result. If exponential functions are used for user utility in this case, user’s profit achieved by the proposed method does not become the largest throughout the whole range of $\alpha$. The main reason of this result is that the decrease of user utility per unit elapsed time when using exponential functions is sharper than that when using power functions, and then, user utility gain becomes smaller, and consequently, weight of the observation cost becomes relatively larger.

Fig. 8 is the result of the simulation on scenario 7. The proposed method achieves the slightly largest user’s profit when $\alpha$ is from 100 to 300. This reason is as follows. When the update frequency is changed, the proposed method invokes “Default Observation Interval Setting process (DOIS process),” which works effectively. For the verification of DOIS process’s effect, comparison of simulation results on scenario 7 with and without the process is shown in Fig. 10. In this figure, we can see that user’s profit gained by using the method without DOIS process is smaller than that with the process because the method without the process cannot adjust its observation interval rapidly and appropriately.

5 Experiments in the Internet environment

In order to show the effectiveness of the proposed method not only in simulations, we implement the method into a prototype of the monitoring system and carry out some experiments in the Internet environment from October in 2009 to January in 2010.
5. 1 Architecture of the monitoring system

The monitoring system consists of four major modules, i.e., User interface, Observation time setting, Page obtaining and Update check, and two databases, i.e., Registration information and Pages’ information. The User interface module has two processes, Registry and Output. Data flow among these modules is shown in Fig. 11.

This system monitors web pages on the Internet environment with the operation explained below.

First, a user inputs the target pages’ titles, URLs and their sites’ importance (i.e., content items’ importance) through the Registry process in the User interface. The Registry process sets an ID to each page for identification. The system manages the pages’ content items by the IDs, which is kept in the Registration information.

After the input of all pages the user intends to monitor, the system starts monitoring. The Observation time setting decides each page’s next observation time by the proposed method which uses the sites’ importance memorized in the Registration information. After the decision of the next observation time, the system waits until the time and then conducts the observation with Page obtaining module. This module accesses and gets target content pages at the scheduled time. After getting pages, the Update check compares the newly-acquired page with the old one saved in the Pages’ information. If there is any difference between them, the module decides that the page was updated and records it to the Pages’ information. Also, the update log is recorded in the same database at the same time. After the update checking, the operation goes back to Observation time setting module and repeatedly conducts the series of processes stated above.

The user can browse the updated pages through the Output process at any time on the browser provided by User interface module. This process also has a function to inform the user about the pages which have been newly updated since they were browsed last.

5. 2 Experimental conditions

As the pages which are intended to be moni-
tored, popular pages with a lot of links to many other pages are selected: a variety of 50 web pages such as news, universities’, governments’, companies’ and blog pages which have various update frequency [11]. The frequency distribution of the target information sources categorized by their average update time is shown in Fig. 12. The minimum time interval of accesses to the sources is 5 minutes, which is decided considering the load imposed on the web servers [12] [17]. The update discovered by the system means that there is some change of pages simply [14] [2] [7]. Therefore, the updates include some pages which are changed automatically by the scripts in respective information sources and do not have any virtual renewal; this kind of updates account for 15% of the total number of them.

The function of the user utility and the average of \( f(v_i) \) are given in the same way of scenario 1.

5.3 Experimental results

User’s profit in the experiment is shown in Fig. 13. It illustrates that the trend of gaining profit in each monitoring method is similar to that derived by the simulation results. A reason of this result is that the distribution of target pages’ update intervals shown in Fig. 12 is similar to that of the simulation’s. From this result, the proposed method also seems to be available in the real Internet environment while the observation cost is neither extremely emphasized nor neglected.

6 Conclusion

In this paper, we proposed an efficient monitoring method working for information sources in a large-scale network, which tries to increase user utility without futile observation cost. This method adjusts its observation intervals according to the information sources’ update intervals considering user utility decrease as time passes and the observation cost. The proposed method was shown to be useful while the observation cost was neither extremely emphasized nor neglected by using computer simulations. In addition, we developed a prototype of the monitoring system implementing the proposed method, and conducted some monitoring experiments. The result showed that the proposed method seemed to be available in the actual Internet environment, in the case where the conditions are close to those of the simulations.

As future works, we need to examine the effectiveness of the proposed method further by computer simulations with much variety of assumptions. Additionally, it is necessary to improve the proposed method so that the importance of information sources is used more effectively for making a monitoring schedule.

References

A Appendix: Specific simulation results and discussion on scenarios 3 and 4

Simulation results of scenarios 3 and 4 are illustrated in figs. 14 and 15. The reason why the proposed method is not affected by the difference of the ways of giving the averages of $f(v_i)$s is that the method does not use the average of maximum user utility in order to estimate user’s profit when changing the observation intervals.

Fig. 14 Relationship between user’s profit and weight of observation cost in scenario 3.

Fig. 15 Relationship between user’s profit and weight of observation cost in scenario 4.
Shinji Sugawara
Shinji Sugawara received the B.Eng., M.Eng., and Dr.Eng. degrees from Tokyo Institute of Technology, Japan, in 1994, 1996, and 1999, respectively. In 1999, he joined the University of Electro-Communications, Tokyo, Japan as an assistant professor, and in 2005, joined Nagoya Institute of Technology, Japan as an associate professor. From 2006 to 2007, he was a visiting researcher in University of California, Irvine. Since April 2008, he has been an associate professor at the Department of Scientific and Engineering Simulation, Graduate School of Engineering, Nagoya Institute of Technology. He is interested in computer communication network and distributed database. He is a member of the IEICE, IPSJ, IEEE and ACM.

Kengo Honda
Kengo Honda received the B.Eng., and M.Eng degree from Nagoya Institute of Technology, Nagoya, Japan, in 2008 and 2010, respectively. He is interested in computer communication network and information retrieval.

Yutaka Ishibashi
Yutaka Ishibashi received the B.E., M.E., and Dr.E. degrees from Nagoya Institute of Technology, Nagoya, Japan, in 1981, 1983, and 1990, respectively. In 1983, he joined the Musashino Electrical Communication Laboratory of NTT. From 1993 to 2001, he served as an Associate Professor of Department of Electrical and Computer Engineering, Faculty of Engineering, Nagoya Institute of Technology. Currently, he is a Professor of Department of Scientific and Engineering Simulation, Graduate School of Engineering, Nagoya Institute of Technology. His research interests include networked multimedia, QoS (Quality of Service) control, and media synchronization. He is a fellow of IEICE and a member of IEEE, ACM, IPSJ, ITE, and VRSJ.