Estimation of False Drops in Set-valued Object Retrieval with Signature Files

Hiroyuki Kitagawa* Yoshiaki Fukushima†§
Yoshiharu Ishikawa‡ and Nobuo Ohbo*

* Institute of Information Sciences and Electronics, University of Tsukuba
† Master's Program in Science and Engineering, University of Tsukuba
‡ Doctoral Program in Engineering, University of Tsukuba
Tennohdai, Tsukuba, Ibaraki 305, Japan
E-mail: kitagawa,ishikawa,ohbo@dblab.is.tsukuba.ac.jp

Abstract. Advanced database systems have to support complex data structures as treated in object-oriented data models and nested relational data models. In particular, efficient processing of set-valued object retrieval (simply, set retrieval) is indispensable for such systems. In the previous paper [6], we proposed the use of *signature files* as efficient set retrieval facilities and showed their potential capabilities based on a disk page access cost model. Retrieval with signature files is always accompanied by mismatches called *false drops*, and it is very important in designing signature files to properly control the false drops.

In this paper, we present an in-depth study of false drops in set retrieval with signature files. We derive formulas estimating false drops in four types of set retrieval based on the "has-subset," "is-subset," "has-intersection," and "is-equal" relationships. Then we evaluate their validity by computer simulations. Simulation study is also done to investigate false drops in practically probable more complex situations.

1 Introduction

According to the recent expansion of the database application domain, requirements for the database system to handle complex data have been increasing. To meet the requirements, advanced data models such as nested relational data models [5, 8, 11] and object-oriented data models [9, 15] are investigated. Those models manipulate more complex data structures and, in particular, directly handle set values. Therefore, efficient set retrieval methods are necessary for database systems supporting those models. Although new indexing schemes such as the nested index and the multiindex [1, 12] considering the hierarchical structures in the complex data objects have been investigated, they are not designed to fully support set manipulation in general. We have proposed the use of *signature files* as efficient set retrieval facilities and showed their potential capabilities focusing on the retrieval with the set inclusion operator ($\subseteq$) [6].

Signature files were originally designed for text retrieval, where text blocks including a certain query word are retrieved [2, 3, 4]. Although applications of signature files to non-text object retrieval such as to the traditional record retrieval [10] and to the Prolog clause retrieval [14] were discussed, study on their capabilities in

§ Current Affiliation: Open Systems Development Laboratories, NEC Corporation.

set retrieval has not been reported by other researchers to the best of our knowledge. Retrieval with signature files is always accompanied by mismatches called *false drops*. The number of false drops has a direct effect on the number of disk page accesses [6]. Therefore, it is important to estimate the false drops and to properly control them in the design of signature files.

In this paper, we present an in-depth study of false drops in set retrieval with signature files. Here, we consider the use of signature files in facilitating four types of set retrieval based on the "has-subset," "is-subset," "has-intersection," and "is-equal" relationships. In the analysis, we take distribution of cardinalities of retrieved sets into account. We derive probability-theoretic formulas estimating the false drops, and evaluate their validity by computer simulations. Moreover, we investigate false drops in practically probable more complex situations.

The paper is organized as follows: In Section 2, we give an overview of signature files and present their use in set retrieval based on the "has-subset," "is-subset," "has-intersection," and "is-equal" relationships. In Section 3, we derive formulas estimating false drops of the retrieval in two situations; (1) Uniform target cardinality, where retrieved sets have the same cardinality, and (2) Varying target cardinality, where their cardinalities follow the binomial distribution. In Section 4, we evaluate validity of those formulas by computer simulations. Simulation study is also done to investigate false drops in more complex situations. Section 5 is the summary and conclusion.

2 Set Retrieval with Signature Files

2.1 Set Retrieval Conditions

One of the typical data models supporting set values and their retrieval is the nested relational model [5, 8, 11]. Figure 1 shows a sample nested relation representing departments in a sales company. Attributes DNO and DNAME represent the department number and name, respectively. Each tuple is tagged with the tuple identifier (TID). SALES_ITEM is a *set attribute* and represents the set of sales items of each department. Tuple retrieval based on the SALES_ITEM attribute value, for instance SALES_ITEM $\supseteq$ {`pen, pencil`}, is an example of set retrieval investigated in this paper. In this case, each SALES_ITEM attribute value is called a *target set*, and {`pen, pencil`} is called a *query set*.

Although a number of nested relational algebras have been proposed, the selection operation in most of the nested relational algebras considers selection conditions including (some of) the following set comparison operators. Here, T and Q denote the target set and query set, respectively, and q denotes a simple value given in case of $T \ni q$.

1. $T \ni q$ (*has-element*): The target set has the simple value q as an element.
   Q1: Retrieve the departments that sell `pencils`.
2. $T \supseteq Q$ (*has-subset*): The target set has the query set as a subset.
   Q2: Retrieve the departments that sell both `pens` and `pencils`.
3. $T \subseteq Q$ (*is-subset*): The target set is a subset of the query set.
   Q3: Retrieve the departments that do not sell items other than `pens`, `pencils`, `erasers`, and `cutters`.

DEPT			
TID	DNO	DNAME	SALES_ITEM
			INAME
T1	314	stationery_1	pen pencil ink
T2	125	stationery_2	notebook eraser clip cutter

Figure 1 Example Nested Relation

4. $T \sqcap Q^*$ (*has-intersection*): The target set has intersection with the query set.
   Q4: Retrieve the departments that sell `erasers` or `staplers`.
5. $T \equiv Q$ (*is-equal*): The target set is equal to the query set.
   Q5: Retrieve the departments that sell only `pens`, `pencils`, and `ink`.

As $T \ni q$ is a special case of $T \supseteq Q$, we consider four types of set retrieval conditions ($T \supseteq Q$, $T \subseteq Q$, $T \sqcap Q$, $T \equiv Q$) in the remaining part of this paper.

2.2 Use of Signature Files

Signature files were originally designed for text retrieval [2, 3, 4]. They require much smaller storage space than inverted files, and they can handle updates easily. A *signature* is a bit pattern formed for each data object and stored in the signature file. A typical query processing with the signature file is as follows: When a query is given, a *query signature* is formed from the query value. Then each signature in the signature file is examined over the query signature for potential match. If the signature satisfies a pre-defined condition implied by the query condition, the corresponding data object becomes a candidate that may satisfy the query. Such a data object is called a *drop*. The last step is the *false drop resolution*, and each drop is accessed and examined whether it actually satisfies the query condition. Drops that fail the test are called *false drops*, while the qualified data objects are called *actual drops*.

In set retrieval with signature files, a *target signature* is generated for each target set as shown in Figure 2. First, each element in a target set is hashed to a binary bit pattern called an *element signature*. All element signatures have F bit length, and m bits are set to "1". Then, a target signature is obtained by bit-wise OR-ing (*superimposed coding*) element signatures of all the elements in the target set. Pairs of such a target signature and a TID of the tuple including the target set are stored in the signature file as shown in Figure 3.

Queries mentioned in Subsection 2.1 are processed with the signature files in the following way.

---

* "$T \sqcap Q$" stands for "$T \cap Q \neq \phi$."

element	element signature	
`pen`	→ 0001000000000101	
`pencil`	→ 1100100000000000	
`ink`	→ 0100001010000000	
	⇓	TID
target signature	1101101010000101 →	T1

Figure 2 Generation of a Target Signature

1101101010000101	T1
1110110111000010	T2
⋮	⋮

Figure 3 Signature File

1) A query signature is generated from the query set Q in the same way as the target signature.
2) The signature file is examined. Each target set becomes a drop if the following condition is satisfied.

   $\boldsymbol{T \supseteq Q}$: $query\ signature \wedge target\ signature = query\ signature$[†].

   $\boldsymbol{T \subseteq Q}$: $query\ signature \wedge target\ signature = target\ signature$.

   $\boldsymbol{T \sqcap Q}$: $weight(query\ signature \wedge target\ signature) \geq m$.

   The number of bits set to "1" is called the *weight*, and the function $weight()$ returns the weight.

   $\boldsymbol{T \equiv Q}$: $query\ signature = target\ signature$.

3) The drops in step 2) are retrieved and checked whether they actually satisfy the query condition (false drop resolution).

Figure 4 illustrates steps 1) and 2) in query processing of the query **Q2**. The tuple T1 becomes a drop because its satisfies the above condition for $T \supseteq Q$.

query element	element signature
`pen`	→ 0001000000000101
`pencil`	→ 1100100000000000
	⇓
query signature	1101100000000101

signature file

drop ←	1101101010000101	T1
	1110110111000010	T2
	⋮	⋮

Figure 4 Query Processing of **Q2**

query element	element signature
`scissors`	→ 1010010010000000
`compasses`	→ 0110000100000000
	⇓
query signature	1110010110000000

signature file

	1101101010000101	T1
drop ←	1110110111000010	T2
	⋮	⋮

Figure 5 False Drop

3 False Drop Probabilities

Figure 5 shows a case where false drops occur. The query with the condition $T \supseteq$ {`scissors, compasses`} is processed in this example. Although the tuple T2 does

[†] '$\wedge$' stands for bit-wise AND operation.

not satisfy this condition, it becomes a drop. Thus, it is a false drop. The false drop is due to the collision of element signatures and the superimposed coding method.

The frequency of false drop is usually measured in the *false drop probability*. The false drop probability Fd is defined as follows:

$$Fd = \frac{\text{false drops}}{N - \text{actual drops}}, \tag{1}$$

where N is the total number of target sets [3].

In set retrieval with signature files, the number of false drops is an important performance measure. Therefore, it is necessary to properly control the false drops. In this section, we derive mathematical formulas estimating false drop probabilities for the four types of queries mentioned in Section 2. As discussed in [3], we can derive those formulas assuming the case of unsuccessful search, where we have no actual drops.

3.1 Basic Considerations

Table 1 shows symbols used in our analysis. We make following assumptions related with the element signature generation.

1. The weight of an element signature is very small compared with the signature size ($m \ll F$).
2. The "1"'s are uniformly distributed in an element signature. Therefore, each bit position is set to "1" with the same probability.

From the assumption 2, each bit position in an element signature is set with the probability m/F. Therefore, the probability that a bit position b_t of a target signature is set to "1" is given by

$$p(b_t) = 1 - \left(1 - \frac{m}{F}\right)^{D_t} \approx 1 - e^{-\frac{mD_t}{F}} \quad (m \ll F).$$

Similarly, the probability that a bit position b_q of a query signature is set to "1" is given by

$$p(b_q) = 1 - \left(1 - \frac{m}{F}\right)^{D_q} \approx 1 - e^{-\frac{mD_q}{F}} \quad (m \ll F).$$

When the signature size F, the element signature weight m, the target set cardinality D_t, and the query set cardinality D_q are given, the expected values of the target signature weight $\bar{m}_t$ and the query signature weight $\bar{m}_q$ are expressed as follows.

$$\bar{m}_t = F \times p(b_t) \approx F(1 - e^{-\frac{mD_t}{F}}),$$
$$\bar{m}_q = F \times p(b_q) \approx F(1 - e^{-\frac{mD_q}{F}}).$$

The following formula giving the false drop probability for $T \ni q$ was derived in [3].

$$Fd_{\{T \ni q\}} = (p(b_t))^m \approx (1 - e^{-\frac{mD_t}{F}})^m. \tag{2}$$

When we consider this formula as a function of m, $Fd_{\{T \ni q\}}$ take the minimum value for

$$m = \frac{F \ln 2}{D_t} \quad (= m_{opt}) \tag{3}$$

[13]. We derive formulas giving false drop probabilities for $T \supseteq Q$, $T \subseteq Q$, $T \sqcap Q$, and $T \equiv Q$ taking these considerations as a starting basis.

Table 1. Symbols

symbol	definition
F	Signature size in bits
m	Element signature weight
D_t	Cardinality of a target set T
D_q	Cardinality of a query set Q
N	Total number of target sets in the database
V	Cardinality of the set element domain

3.2 Uniform Target Cardinality

In this subsection, we consider the case where all target sets have the same cardinality D_t. First, we derive equations used in later analysis. Let b_t^j $(1 \leq j \leq F)$ be the jth bit position of the target signature and b_q^j $(1 \leq j \leq F)$ be jth bit position of the query signature. For each i $(1 \leq i \leq F - m)$, the following equations hold:

$$\begin{aligned} Prob(b_t^1 = 0 \wedge \cdots \wedge b_t^i = 0) &= \left(\frac{{}_{F-i}C_m}{{}_FC_m}\right)^{D_t} \\ &= \left(\frac{(F-m)(F-m-1)\cdots(F-m-i+1)}{F(F-1)\cdots(F-i+1)}\right)^{D_t} \\ &= \prod_{k=1}^{i}\left(1 - \frac{m}{F-k+1}\right)^{D_t}. \end{aligned} \tag{4}$$

If $\frac{m}{F-k+1} \ll 1$ is satisfied for $1 \leq k \leq i$,

$$\begin{aligned} Prob(b_t^1 = 0 \wedge \cdots \wedge b_t^i = 0) &= \left(1 - \tfrac{m}{F}\right)^{D_t}\left(1 - \tfrac{m}{F-1}\right)^{D_t}\cdots\left(1 - \tfrac{m}{F-i+1}\right)^{D_t} \\ &\approx \left(1 - \tfrac{1}{F}\right)^{mD_t}\left(1 - \tfrac{1}{F-1}\right)^{mD_t}\cdots\left(1 - \tfrac{1}{F-i+1}\right)^{mD_t} \\ &= \left(1 - \tfrac{i}{F}\right)^{mD_t}. \end{aligned} \tag{5}$$

Similarly,

$$Prob(b_q^1 = 0 \wedge \cdots \wedge b_q^i = 0) = \prod_{k=1}^{i}\left(1 - \frac{m}{F-k+1}\right)^{D_q}. \tag{6}$$

If $\frac{m}{F-k+1} \ll 1$ is satisfied for $1 \leq k \leq i$,

$$Prob(b_q^1 = 0 \wedge \cdots \wedge b_q^i = 0) \approx \left(1 - \tfrac{i}{F}\right)^{mD_q}. \tag{7}$$

3.2.1 False Drop Probability Formulas

We derive formulas giving the false drop probabilities based on assumption that the weight of every target signature is equal to its expected value $\bar{m_t}$ and the weight of every query signature is equal to $\bar{m_q}$.

1) $\boldsymbol{T \supseteq Q}$: A false drop occurs when the following condition holds for every bit position j $(1 \leq j \leq F)$:

$$b_t^j = 0 \quad \Rightarrow \quad b_q^j = 0.$$

As the number of "0"'s in the target signature is $F - \bar{m}_t \approx Fe^{-\frac{mD_t}{F}}$ by the above assumption, we can tell the false drop probability for $T \supseteq Q$ from equation (6) as follows:

$$Fd_{\{T \supseteq Q\}} = Prob(b_q^1 = 0 \wedge \cdots \wedge b_q^{F-\bar{m}_t} = 0) \approx \prod_{k=1}^{Fe^{-\frac{mD_t}{F}}} \left(1 - \frac{m}{F-k+1}\right)^{D_q}.$$

If $\frac{m}{F-k+1} \ll 1$ holds for $1 \leq k \leq Fe^{-\frac{mD_t}{F}}$, we get from equation (7),

$$Fd_{\{T \supseteq Q\}} \approx (1 - e^{-\frac{mD_t}{F}})^{mD_q}. \tag{8}$$

Note that this formula coincides with formula (2) when $D_q = 1$, namely $T \ni q$. The formula (8) takes the minimum value for $m = m_{opt}$ given in (3).

2) $\boldsymbol{T \subseteq Q}$: A false drop occurs when the following condition holds for every bit position j $(1 \leq j \leq F)$:

$$b_q^j = 0 \quad \Rightarrow \quad b_t^j = 0.$$

As the number of "0"'s in the query signature is $F - \bar{m}_q \approx Fe^{-\frac{mD_q}{F}}$ by the above assumption, we can tell the false drop probability for $T \subseteq Q$ from equation (4) as follows:

$$Fd_{\{T \subseteq Q\}} = Prob(b_t^1 = 0 \wedge \cdots \wedge b_t^{F-\bar{m}_q} = 0) \approx \prod_{k=1}^{Fe^{-\frac{mD_q}{F}}} \left(1 - \frac{m}{F-k+1}\right)^{D_t}.$$

If $\frac{m}{F-k+1} \ll 1$ holds for $1 \leq k \leq Fe^{-\frac{mD_t}{F}}$, we get from equation (5),

$$Fd_{\{T \subseteq Q\}} \approx (1 - e^{-\frac{mD_q}{F}})^{mD_t}. \tag{9}$$

This formula take the minimum value for the following m:

$$m = \frac{F \ln 2}{D_q}.$$

3) $\boldsymbol{T \sqcap Q}$: A false drop occurs when there exist at least m bit positions that satisfy the following condition:

$$b_q^j = 1 \quad \wedge \quad b_t^j = 1.$$

In other words, a false drop does not occur when there exist less than m bit positions that satisfy the above condition. Here, we make an additional assumption that each bit position in the target signature is set to "1" with probability $p(b_t)$ independently of other bit positions. Then the probability that there exist k bit positions that satisfy the above condition is given as follows:

$${}_{\bar{m}_q}C_k p(b_t)^k (1 - p(b_t))^{\bar{m}_q - k}.$$

Therefore, the false drop probability for $T \sqcap Q$ is given by

$$Fd_{\{T \sqcap Q\}} = 1 - \sum_{k=0}^{m-1} {}_{\bar{m}_q}C_k p(b_t)^k (1 - p(b_t))^{\bar{m}_q - k}. \tag{10}$$

4) $\boldsymbol{T \equiv Q}$: We consider the case where $D_t = D_q$. More general cases are considered in the next subsection. In this case, $\bar{m}_t = \bar{m}_q = \bar{m}$. Then, a target signature and a query signature may take ${}_FC_{\bar{m}}$ different bit patterns. A false drop occurs when the target signature is equal to the query signature. Hence, the false drop probability for $T \equiv Q$ is given as follows:

$$Fd_{\{T \equiv Q\}} = \frac{1}{{}_FC_{\bar{m}}}. \tag{11}$$

In this case, ${}_FC_{\bar{m}}$ takes the maximum value for $\bar{m} = F/2$. Therefore, the false drop probability takes the minimum for $m = m_{opt}$.

3.2.2 Revised False Drop Probability Formulas

In Subsection 3.2.1, we assumed that the weights of the target signature and the query signature are given by $\bar{m}_t$ and $\bar{m}_q$, respectively. Actually, this assumption does not hold, and the weight of the target signature, for example, varies even if F, m, and D_t are fixed. In this subsection, we revise the false drop probability formulas taking distribution of the target signature weight and the query signature weight into account. As in case of $T \sqcap Q$ in Subsection 3.2.1, let us assume that each bit position in the target signature and the query signature is set to "1" with probabilities $p(b_t)$ and $p(b_q)$, respectively, independently of other bit positions. Then the distribution of the target signature weight follows the binomial distribution, and the probability that the target signature weight is i is given by

$$p_t(i) = {}_FC_i p(b_t)^i (1 - p(b_t))^{F-i}.$$

Similarly, the probability that the query signature weight is i is given by

$$p_q(i) = {}_FC_i p(b_q)^i (1 - p(b_q))^{F-i}.$$

In the following, we derive revised false drop probability formulas for $T \supseteq Q$, $T \subseteq Q$, $T \sqcap Q$, and $T \equiv Q$. The effect of this revision is evaluated with simulation results in Section 4.

1) $\boldsymbol{T \supseteq Q}$: If the target signature weight is i, the probability $f_{\{T \supseteq Q\}}(i)$ that the target set becomes a false drop is derived from equation (6) as follows:

$$f_{\{T \supseteq Q\}}(i) = Prob(b_q^1 = 0 \wedge \cdots \wedge b_q^{F-i} = 0) = \prod_{k=1}^{F-i} \left(1 - \frac{m}{F-k+1}\right)^{D_q}.$$

If $\frac{m}{F-k+1} \ll 1$ holds for $1 \leq k \leq F - i$, we get from equation (7)

$$f_{\{T \supseteq Q\}}(i) \approx \left(1 - \frac{F-i}{F}\right)^{mD_q} = \left(\frac{i}{F}\right)^{mD_q}.$$

As distribution of the target signature weight follows the binomial distribution, the false drop probability for $T \supseteq Q$ is given by

$$\begin{aligned} Fd_{\{T \supseteq Q\}} &= \sum_{i=1}^{F} p_t(i) f_{\{T \supseteq Q\}}(i) \\ &= \sum_{i=1}^{F} {}_FC_i (1 - e^{-\frac{mD_t}{F}})^i e^{-\frac{mD_t}{F}(F-i)} \left(\frac{i}{F}\right)^{mD_q}. \end{aligned} \tag{12}$$

2) $\boldsymbol{T \subseteq Q}$: If the query signature weight is i, the probability $f_{\{T \subseteq Q\}}(i)$ that the target set becomes a false drop is derived from equation (4) as follows:

$$f_{\{T \subseteq Q\}}(i) = Prob(b_t^1 = 0 \wedge \cdots \wedge b_t^{F-i} = 0) = \prod_{k=1}^{F-i} \left(1 - \frac{m}{F-k+1}\right)^{D_t}.$$

If $\frac{m}{F-k+1} \ll 1$ holds for $1 \leq k \leq F-i$, we get from equation (5)

$$f_{\{T \subseteq Q\}}(i) \approx \left(1 - \frac{F-i}{F}\right)^{mD_t} = \left(\frac{i}{F}\right)^{mD_t}.$$

As distribution of the query signature weight follows the binomial distribution, the false drop probability for $T \subseteq Q$ is given by

$$\begin{aligned} Fd_{\{T \subseteq Q\}} &= \sum_{i=1}^{F} p_q(i) f_{\{T \subseteq Q\}}(i) \\ &= \sum_{i=1}^{F} {}_F C_i (1 - e^{-\frac{mD_q}{F}})^i e^{-\frac{mD_q}{F}(F-i)} \left(\frac{i}{F}\right)^{mD_t}. \end{aligned} \tag{13}$$

3) $\boldsymbol{T \sqcap Q}$: If the query signature weight is i, the probability $f_{\{T \sqcap Q\}}(i)$ that the target set becomes a false drop is given by

$$f_{\{T \sqcap Q\}}(i) = 1 - \sum_{k=0}^{m-1} {}_i C_k p(b_t)^k (1 - p(b_t))^{i-k}.$$

As distribution of the query signature weight follows the binomial distribution, the false drop probability for $T \sqcap Q$ is given by

$$\begin{aligned} Fd_{\{T \sqcap Q\}} &= \sum_{i=1}^{F} p_q(i) f_{\{T \sqcap Q\}}(i) \\ &= \sum_{i=1}^{F} {}_F C_i p(b_q)^i (1 - p(b_q))^{F-i} \left(1 - \sum_{k=0}^{m-1} {}_i C_k p(b_t)^k (1 - p(b_t))^{i-k}\right). \end{aligned} \tag{14}$$

4) $\boldsymbol{T \equiv Q}$: A false drop occurs if both the target signature weight and the query signature weight take the same value i, and the target signature is equal to the query signature. Therefore, the false drop probability for $T \equiv Q$ is given by

$$\begin{aligned} Fd_{\{T \equiv Q\}} &= \sum_{i=1}^{F} p_t(i) p_q(i) \frac{1}{{}_F C_i} \\ &= \sum_{i=1}^{F} {}_F C_i (1 - e^{-\frac{mD_t}{F}})^i (1 - e^{-\frac{mD_q}{F}})^i e^{-\frac{m(D_t + D_q)}{F}(F-i)}. \end{aligned} \tag{15}$$

3.3 Varying Target Cardinality

In this subsection, we consider a case where the cardinality of the target set varies. Let a *set element domain* be a set from which each target set element is taken, and its cardinality be V. Here, we assume that each of the V elements in the set element domain appears in a target set with the same probability p. Then distribution of the target set cardinality follows the binomial distribution, and the average target set cardinality is given by $\bar{D}_t = pV$.

In this case, we can estimate the false drop probability with the following formula for each case of $T \supseteq Q$, $T \subseteq Q$, $T \sqcap Q$, and $T \equiv Q$:

$$Fd_{B,c} = \sum_{D_t=1}^{V} {}_V C_{D_t} p^{D_t} (1-p)^{V-D_t} Fd_c, \tag{16}$$

where c is a parameter indicating one of $\{T \supseteq Q\}$, $\{T \subseteq Q\}$, $\{T \sqcap Q\}$, and $\{T \equiv Q\}$, and Fd_c is the estimation formula of the false drop probability derived in Subsection 3.2 for each case. For example, the false drop probability for $T \supseteq Q$ using formula (8) is given as follows:

$$Fd_{B,\{T \supseteq Q\}} = \sum_{D_t=1}^{V} {}_V C_{D_t} p^{D_t} (1-p)^{V-D_t} (1 - e^{-\frac{mD_t}{F}})^{mD_q}$$

4 Simulation Study

In this section, we evaluate validity of estimation by the formulas derived in Section 3 by simulations. A number of physical signature file organizations have been proposed [2, 7]. In this study, we show the simulation results where parameter values are set assuming the bit-sliced signature file organization, since our previous work [6] showed advantages of the bit-sliced signature files in set retrieval. We set $M = 10000, V = 10000$, and $m = 2$. Simulation results with $D_t = 10$ and $D_t = 100$ are presented for four types of queries $T \supseteq Q$ (including $T \ni q$), $T \subseteq Q$, $T \sqcap Q$, and $T \equiv Q$. In the following discussion, we show simulation results comparing with estimates by the formulas derived in Section 3. In figures, we tag simulation results with `sim`, estimates by the formulas in Subsection 3.2.1 with `est1`, and estimates by the formulas in Subsection 3.2.2 with `est2`.

4.1 Uniform Target Cardinality

To check validity of the formulas derived in Subsection 3.2, we consider the case where the target sets have the same cardinality D_t.

1) $\boldsymbol{T \supseteq Q}$: Figure 6 shows the simulation results, estimates by formula (8), and estimates by formula (12) for $D_t = 10$. Figure 7 shows the case of $D_t = 100$. We can see that (a) there is almost no difference between false drop probabilities given by formulas (8) and (12), and that (b) both formulas quite correctly estimate the actual false drop probabilities. In the following discussion, we use the simpler one, formula (8), to estimate the false drop probability for $T \supseteq Q$.

2) $\boldsymbol{T \subseteq Q}$: Figure 8 shows the simulation results, estimates by formula (9), and estimates by formula (13) for $D_t = 10$. Figure 9 shows the case of $D_t = 100$. We can see that (a) the false drop probability given by formula (9) tends to be smaller than that given by formula (13), and that (b) the false drop probability given by (13) coincides well with the simulation results. Therefore, we can see that formula (13) gives more correct estimates of the false drop probability[‡].

[‡] In the cost analysis in [6], we used formula (9) to estimate the false drop probability for $T \subseteq Q$. However, the analysis is still valid, since the difference between estimates by (9) and (13) does not have a significant effect on the total cost.

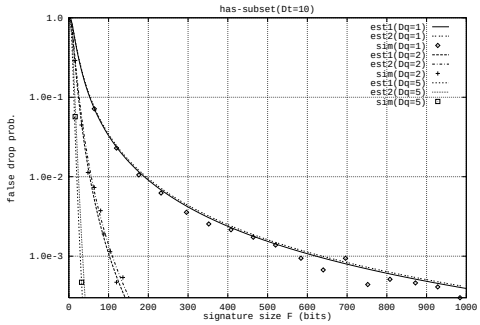

Figure 6 $T \supseteq Q$
($D_t = 10$, including $T \ni q$)
`est1`: formula (8), `est2`: formula (12)

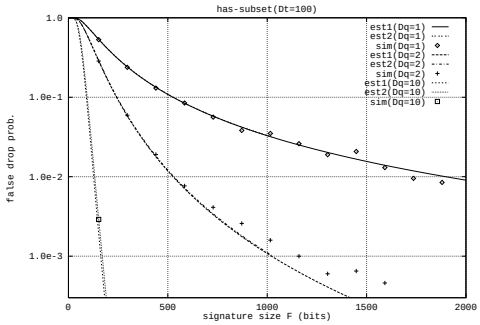

Figure 7 $T \supseteq Q$
($D_t = 100$, including $T \ni q$)
`est1`: formula (8), `est2`: formula (12)

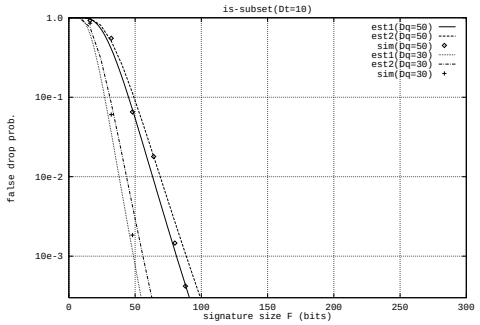

Figure 8 $T \subseteq Q$ ($D_t = 10$)
`est1`: formula (9), `est2`: formula (13)

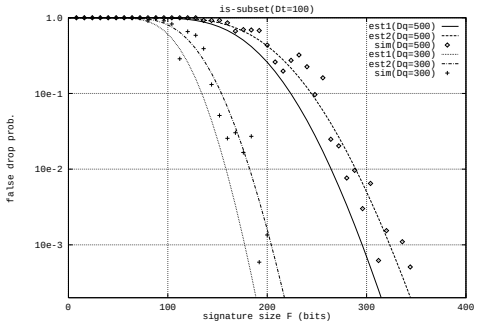

Figure 9 $T \subseteq Q$ ($D_t = 100$)
`est1`: formula (9), `est2`: formula (13)

3) $\boldsymbol{T \sqcap Q}$: Figure 10 shows the simulation results, estimates by formula (10), and estimates by formula (14) for $D_t = 10$. Figure 11 shows the case of $D_t = 100$. We can see that (a) the false drop probability given by (10) coincides well with the simulation results, and that (b) the false drop probability given by (14) does not fit the simulation results when D_q is too small. The latter phenomenon can be explained as follows. When D_q is very small, the query signature weight becomes mD_q in most cases. Therefore, the assumption for (14) that the distribution of the query signature weight follows the binomial distribution is not reasonable in such cases.

4) $\boldsymbol{T \equiv Q}$: Figure 12 shows the simulation results, estimates by formula (11), and estimates by formula (15) for $D_t = 10$. Note that formula (11) can be used only when $D_t = D_q$. Figure 13 shows the case of $D_t = 100$. We can see that (a) the false drop probability given by (11) does not fit the simulation results for large D_t values, and that (b) the false drop probability given by (15) coincides well with the simulation results. Another simulation result indicates that (15) is not so reliable

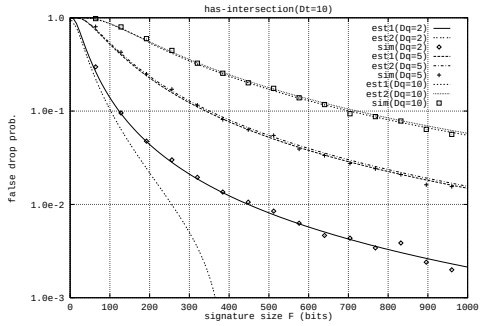

Figure 10 $T \sqcap Q$ ($D_t = 10$)
`est1`: formula (10), `est2`: formula (14)

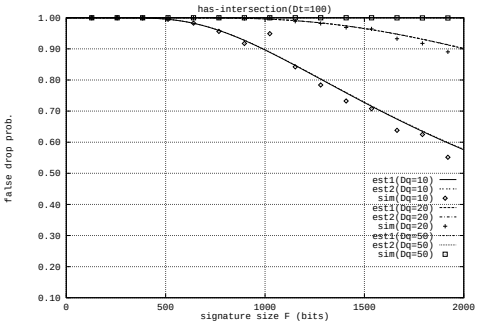

Figure 11 $T \sqcap Q$ ($D_t = 100$)
`est1`: formula (10), `est2`: formula (14)

when D_t and/or D_q is too small. A remark similar to that for $T \sqcap Q$ will apply to such cases. However, we can use (15) for most probable D_t and D_q.

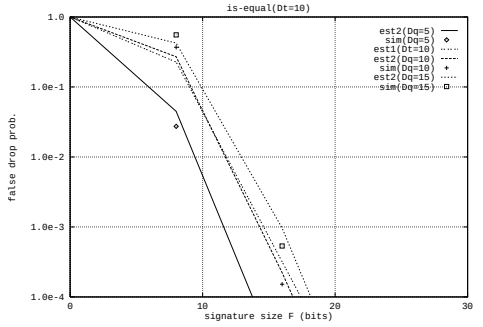

Figure 12 $T \equiv Q$ ($D_t = 10$)
`est1`: formula (11), `est2`: formula (15)

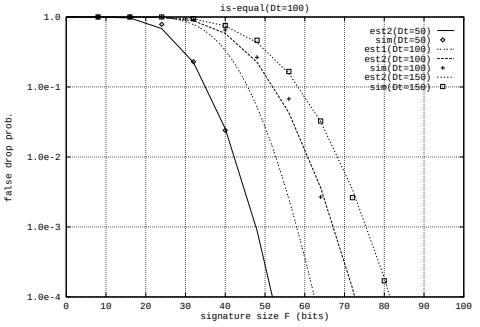

Figure 13 $T \equiv Q$ ($D_t = 100$)
`est1`: formula (11), `est2`: formula (15)

4.2 Alternative Processing Scheme for T ⊓ Q

As shown in Figures 10 and 11, the false drop probability for $T \sqcap Q$ is rather high. The reason is that the query signature generated with the superimposed coding has much chance of having more than m bit intersection with the target signature. To solve this problem, we can use another scheme to process queries of type $T \sqcap Q$. In the scheme, the steps 1) and 2) explained in Subsection 2.2 are modified as follows:

1) An element signature is generated for each element in the query set Q.
2) The signature file is examined. Each target set becomes a drop if any element signature generated in step 1) satisfies the following condition.

$$element\ signature \wedge target\ signature = element\ signature.$$

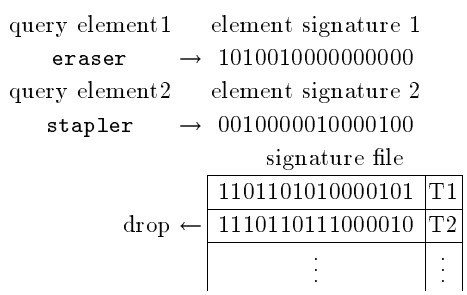

Figure 14 Query Processing of the Query **Q4**

Figure 14 illustrates query processing of the query **Q4** under this scheme.

If we ignore the probability of the hash collision in element signature generation, the false drop probability for $T \sqcap Q$ under this scheme is given by

$$Fd_{\{T \sqcap Q\}} = \sum_{i=1}^{D_q} Fd_{\{T \ni q\}} * (1 - Fd_{\{T \ni q\}})^{i-1} = 1 - (1 - Fd_{\{T \ni q\}})^{D_q}. \qquad (17)$$

By differentiating expression (17) with $Fd_{\{T \ni q\}}$, we get

$$\frac{\Delta Fd_{\{T \sqcap Q\}}}{\Delta Fd_{\{T \ni q\}}} = D_q(1 - Fd_{\{T \ni q\}})^{D_q - 1}.$$

Since $0 < Fd_{\{T \ni q\}} \leq 1$,

$$\frac{\Delta Fd_{\{T \sqcap Q\}}}{\Delta Fd_{\{T \ni q\}}} > 0.$$

Therefore, when $Fd_{\{T \ni q\}}$ takes the minimum value, namely $m = m_{opt}$, $Fd_{\{T \sqcap Q\}}$ also takes the minimum value.

Figure 15 shows the simulation results under this improved scheme, and estimates by formula (17) for $D_t = 10$. Note that formula (2) is used for $Fd_{\{T \ni q\}}$. Figure 16 shows the case of $D_t = 100$. We can see that estimates by (17) coincide well with the simulation results. In the following discussion, we suppose $T \sqcap Q$ is processed in this scheme.

4.3 Varying Target Cardinality

4.3.1 Uniform Element Occurrence Frequency

To check validity of estimation by the formulas derived in Subsection 3.3, we consider the case where each element in the set element domain appears in a target set with the same probability p. As discussed in Subsection 3.3, distribution of the target set cardinality follows the binomial distribution with the average $\bar{D}_t = pV$. In the following discussion, we tag estimates given by the formulas in Subsection 3.3 with `est3` in the figures.

1) $\boldsymbol{T \supseteq Q}$: Figure 17 shows the simulation results and estimates by formula (16) (formula (8) is used for $Fd_{\{T \supseteq Q\}}$ in (16)) for $\bar{D}_t = 10$. Figure 18 shows the case of $\bar{D}_t = 100$. For comparison, we also show the estimates by (8) with $D_t = \bar{D}_t$ in the figures. Estimates by (16) coincide with the simulation results. In $T \supseteq Q$, the false drop probability increases drastically as D_t becomes large. Therefore, formula (16) gives little larger estimates than formula (8), influenced by target sets with $D_t > \bar{D}_t$. However, there is no significant difference between estimates given by (8) and (16).

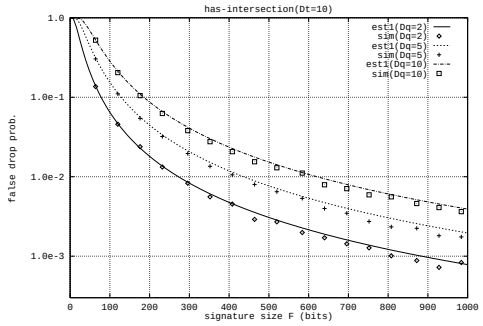

Figure 15 $T \sqcap Q$ ($D_t = 10$)
`est1`: formula (17)

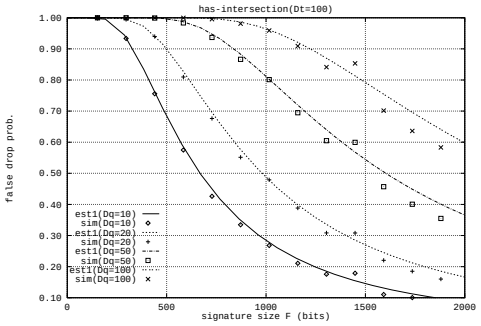

Figure 16 $T \sqcap Q$ ($D_t = 100$)
`est1`: formula (17)

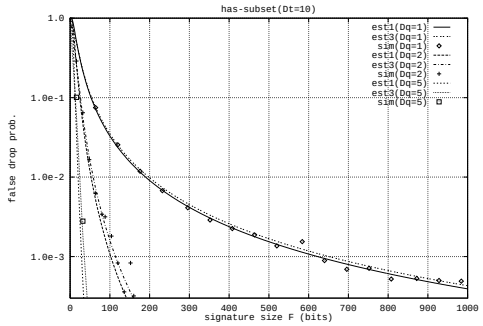

Figure 17 $T \supseteq Q$
($\bar{D}_t = 10$, including $T \ni q$)
`est1`: formula (8), `est3`: formula (16)

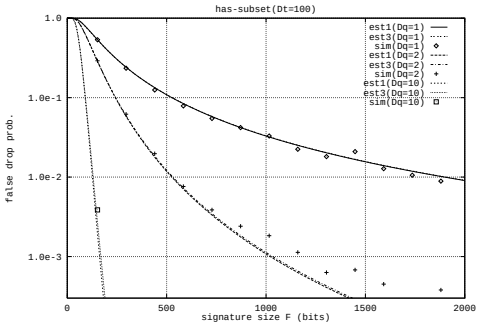

Figure 18 $T \supseteq Q$
($\bar{D}_t = 100$, including $T \ni q$)
`est1`: formula (8), `est3`: formula (16)

2) $\boldsymbol{T \subseteq Q}$: Figure 19 shows the simulation results and estimates by formula (16) (formula (13) is used for $Fd_{\{T \subseteq Q\}}$ in (16)) for $\bar{D}_t = 10$. Figure 20 shows the case of $\bar{D}_t = 100$. For comparison, we also show the estimates by (13) with $D_t = \bar{D}_t$. Estimates by (16) coincide well with the simulation results. In $T \subseteq Q$, the false drop probability increases drastically as D_t becomes small. Therefore, formula (16) gives larger estimates than formula (13), influenced by target sets with $D_t < \bar{D}_t$. In case of $\bar{D}_t = 10$, distribution of the target set cardinality gives more influence than in case of $\bar{D}_t = 100$.

3) $\boldsymbol{T \sqcap Q}$: Figure 21 shows the simulation results and estimates by formula (16) (formula (17) is used for $Fd_{\{T \sqcap Q\}}$ in (16)) for $\bar{D}_t = 10$. Figure 22 shows the case of $\bar{D}_t = 100$. For comparison, we also show estimates by (17) with $D_t = \bar{D}_t$. Estimates by (16) coincide well with the simulation results. In this case, there is no significant difference between estimates by (17) and (16).

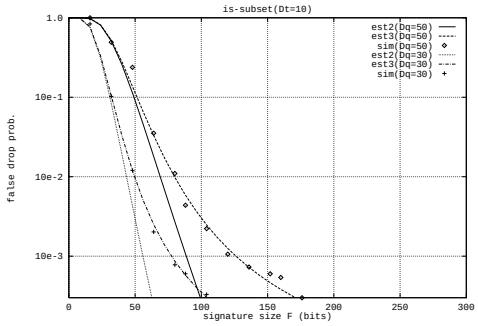

Figure 19 $T \subseteq Q$ ($\bar{D}_t = 10$)
`est2`: formula (13), `est3`: formula (16)

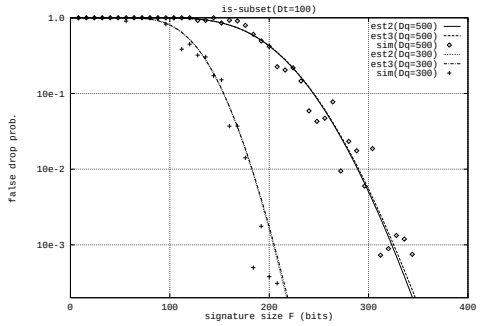

Figure 20 $T \subseteq Q$ ($\bar{D}_t = 100$)
`est2`: formula (13), `est3`: formula (16)

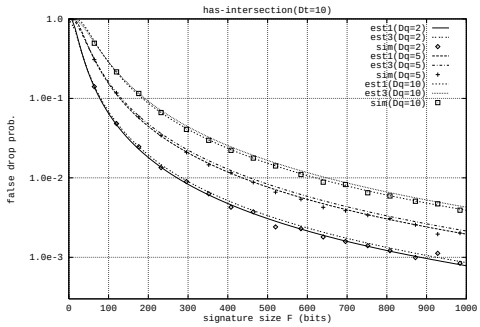

Figure 21 $T \sqcap Q$ ($\bar{D}_t = 10$)
`est1`: formula (17), `est3`: formula (16)

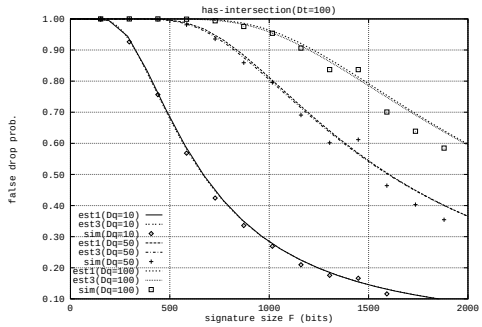

Figure 22 $T \sqcap Q$ ($\bar{D}_t = 100$)
`est1`: formula (17), `est3`: formula (16)

4) $\boldsymbol{T \equiv Q}$: Figure 23 shows the simulation results and estimates by formula (16) (formula (15) is used for $Fd_{\{T \equiv Q\}}$ in (16)) for $\bar{D}_t = 10$. Figure 24 shows the case of $\bar{D}_t = 100$. For comparison, we also show estimates by (15) with $D_t = \bar{D}_t$. Estimates by (16) coincide well with the simulation results and there is no significant difference between estimates by (15) and (16).

4.3.2 Skewed Element Occurrence Frequency

In this subsection, we consider a more probable case in practice where elements in the set element domain do not have the same probability of occurrence in a target set. We assume that some elements in the set element domain are "hot spots," and have higher occurrence frequency than the others. The cardinality V of the set element domain is assumed to be 10000 here. We examine the following three cases.

Case 1. 100 elements in the set element domain have 100 times larger occurrence probability than the others.

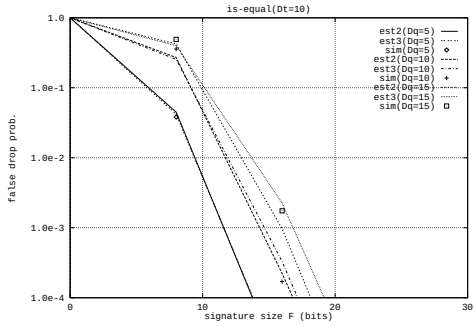

Figure 23 $T \equiv Q$ ($\bar{D}_t = 10$)
`est2`: formula (15), `est3`: formula (16)

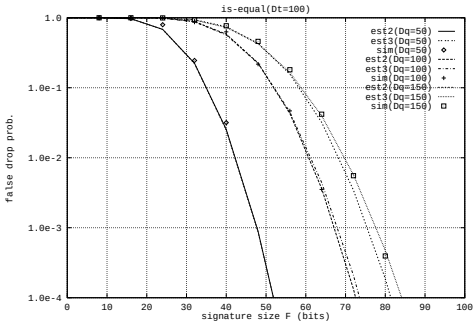

Figure 24 $T \equiv Q$ ($\bar{D}_t = 100$)
`est2`: formula (15), `est3`: formula (16)

Case 2. 1000 elements in the set element domain have 100 times larger occurrence probability than the others.

Case 3. 3000 elements in the set element domain have 100 times larger occurrence probability than the others.

The average target set cardinality is given by

$$\bar{D}_t = p_h h + p(V - h),$$

where h is the number of elements with larger occurrence probability, p is the occurrence probability of regular elements in the set element domain, and $p_h = 100p$. For each of the above three cases, the p value is determined by the above equation with $\bar{D}_t = 10$. In Case 1, the experimental target sets generated for the simulation with $p = 5.0 \times 10^{-4}$ have the average cardinality $\bar{D}_t = 10.2$. Similarly, the average cardinalities in Cases 2 and 3 are $\bar{D}_t = 10.9$ ($p = 9.2 \times 10^{-5}$) and $\bar{D}_t = 12.2$ ($p = 3.3 \times 10^{-5}$), respectively.

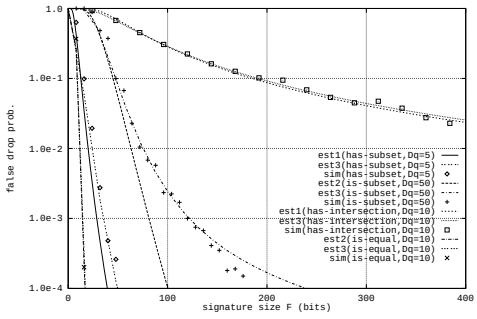

Figure 25 $h = 100, p_h = 100p$

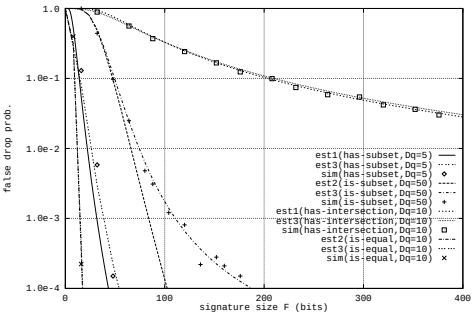

Figure 26 $h = 1000, p_h = 100p$

Figure 25 shows the simulation results, estimates by formulas in Subsection 3.2, and estimates by (16) for four types of queries in Case 1. Figures 26 and 27 show

the false drop probabilities in Cases 2 and 3, respectively. These figures show that estimates by (16) coincide well with the simulation results. Formula (16) was derived assuming that each element in the set element domain appears in a target set with the same probability. Therefore, the results here indicate that existence of hot spots does not have much effect on the false drop probability.

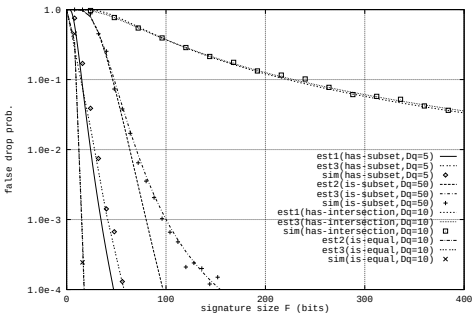

Figure 27 $h = 3000, p_h = 100p$

5 Summary and Conclusion

In this paper, we derived probability-theoretic formulas estimating false drop probabilities in set retrieval with signature files. We considered four types of retrieval conditions $T \supseteq Q, T \subseteq Q, T \sqcap Q$, and $T \equiv Q$. We evaluated their validity by computer simulations. The simulation results can be summarized as follows:

A) In case of uniform target set cardinality, formulas derived in Subsections 3.2 (and Subsection 4.2) well estimate the actual false drop probabilities. We can use formulas (8) for $T \supseteq Q$, (13) for $T \subseteq Q$, (17) for $T \sqcap Q$, and (15) for $T \equiv Q$. Note that we use the scheme in Subsection 4.2 for $T \sqcap Q$.

B) In case distribution of the target set cardinality follows the binomial distribution, formulas derived in Subsection 3.3 (and Subsection 4.2) well estimate the actual false drop probabilities. We can use formula (16) (with (8) for $T \supseteq Q$, (13) for $T \subseteq Q$, (17) for $T \sqcap Q$, and (15) for $T \equiv Q$). The simulation results show that we can get good estimates using just (8) for $T \supseteq Q$, (17) for $T \sqcap Q$, and (15) for $T \equiv Q$ even in this case.

C) We have not derived false drop probability formulas for more probable cases where the set element domain has some hot spots. The simulation results indicate that we do not make big mistakes in estimation substituting the set of formulas derived for the above case B) even though some hot spots may appear in the set element domain.

In this study, we have shown simulation results with parameters set assuming the bit-sliced signature file organization. Simulation study has been done for some other parameter sets, and we have checked that the formulas derived in this paper are also

valid even in the context of the sequential signature file organization. We can use the formulas in many situations in the design of signature files to facilitate set retrieval in advanced database systems.

Acknowledgements

The authors are grateful to Prof. Yuzuru Fujiwara and Prof. Isao Suzuki for their encouragement to this research. They also thank many members of the database laboratory, University of Tsukuba, for fruitful discussion and collaboration.

References

1. E. Bertino and W. Kim, *"Index Techniques for Queries on Nested Objects,"* IEEE Trans. Knowledge and Data Engineering, Vol. 1, No. 2, June 1989, pp. 196-214.
2. W. W. Chang and H. J. Scheck, *"A Signature Access Method for the Starburst Database System,"* Proc. 15th VLDB, Amsterdam, The Netherlands, August 1989, pp. 145-153.
3. C. Faloutsos and S. Christdoulakis, *"An Access Method for Documents and Its Analytical Performance Evaluation,"* ACM Trans. Office Information Systems, Vol. 2, No. 4, October 1984, pp. 267-288.
4. C. Faloutsos and S. Christdoulakis, *"Description and Performance Analysis of Signature File Methods for Office Filing,"* ACM Trans. Office Information Systems, Vol. 5, No. 3, July 1987, pp. 237-257.
5. C. Fischer and S. Thomas, *"Operators for Non-First-Normal-Form Relations,"* Proc. IEEE COMPSAC 83, Chicago, November 1983, pp. 464-475.
6. Y. Ishikawa, H. Kitagawa, and N. Ohbo, *"Evaluation of Signature Files as Set Access Facilities in OODBs,"* Proc. ACM SIGMOD 1993, Washington, D.C., May 1993, pp.247-256.
7. A. Kent, R. Sachs-Davis, and K. Ramamohanarao, *"A Superimposed Coding Schema Based on Multiple Block Descriptor Files for Indexing Very Large Data Bases,"* Proc. 14th VLDB, Los Angeles, August 1988, pp. 351-359.
8. H. Kitagawa and T. L. Kunii, *"The Unnormalized Relational Data Model - For Office Form Processor Design,"* Springer-Verlag, 1989.
9. W. Kim, *"Introduction to Object-Oriented Databases,"* Computer Systems Series, The MIT Press, 1990.
10. R. Sacks-Davis and A. Kent, *"Multikey Access Methods Based on Superimposed Coding Techniques,"* ACM Trans. Database Systems, Vol12, No. 4, December 1987, pp. 655-698.
11. H. Schek and M. Scholl, *"The Relational Model with Relation-Valued Attributes,"* Information Systems, Vol. 11, No. 2, 1986, pp. 137-147.
12. J. Stein and D. Maier *"Associative Access Support in GemStone,"* in K. R. Dittrich, V. Dayal, and A. P. Buchmann (Eds.), On Object-Oriented Database Systems, Springer-Verlag, 1991, pp. 323-339.
13. S. Stiassny, *"Mathematical Analysis of Various Superimposed Coding Methods,"* American Documentation, Vol. 11, February 1960, pp. 155-169.
14. K. F. Wong and M. H. Williams, *"A Superimposed Codeword Indexing Schema for Handling Sets in Prolog Databases,"* Proc. 2nd International Symposium on Database Systems for Advanced Applications, Tokyo, Japan, April 1991, pp. 468-476.
15. S. Zdonik and D. Maier, (Eds.), *"Readings in Object-Oriented Database Systems,"* Morgan Kaufmann Publishers, 1990.