ELSEVIER

Journal of Systems Architecture 47 (2001) 445–458

JOURNAL OF SYSTEMS ARCHITECTURE

www.elsevier.com/locate/sysarc

Gray code clustering of wireless data for partial match queries

Ji Yeon Lee [a], Yon Dohn Chung [b], Yoon Joon Lee [b], Myoung Ho Kim [b,*]

[a] *New Technology Team, Hyundai Information Technology Co., Ltd., San 1-8, Mabuk-ri Guseong-up, Yongin-si, Gyeonggi-do 449-910, South Korea*

[b] *Department of EECS, Division of Computer Science, Korea Advanced Institute of Science and Technology (KAIST), 373-1 KuSung-Dong, YuSung-Gu, Taejon 305-701, South Korea*

Received 20 March 2000; received in revised form 25 November 2000; accepted 21 January 2001

Abstract

This paper proposes a broadcast data clustering method for partial match queries in mobile distributed systems. An effective broadcast data clustering method enables mobile clients to access the data in short latency. Our method utilizes the properties of the Gray coding scheme – Gray codewords have high locality. We describe the way the Gray code method (GCM) effectively clusters wireless data for partial match queries. And we analyze and evaluate the performance of the Gray code clustering method through comparison with other methods.

Keywords: Data clustering; Partial match query; Data broadcasting; Wireless data; Gray codes; Mobile computing

1. Introduction

There are two important parameters related to wireless data broadcasting: access time and tuning time. The access time is the amount of elapsed time from the point of a query submitted by a client to the receipt of the required data through the broadcast channel. The tuning time is the amount of time spent by a client listening to the channel.

There have been some studies on reducing the access time such as caching and non-uniform broadcasting [1,12]. Caching frequently accessed data objects in local storage enables mobile clients to directly access the data without remote access delay to the server. And, when the server broadcasts data in a non-uniform fashion, i.e., highly accessed data objects are more frequently broadcasted, mobile clients that access frequently broadcasted data can access data objects of their interests in a short access time.

Studies on reducing the tuning time such as indexing and hashing have also been made in the past [3,6]. Through efficient index or hash information on broadcasting data objects, mobile clients can access the data in an energy-efficient manner, that is, they can skip unnecessary tuning of wireless broadcasting (i.e., energy consumption).

Most of the previous studies consider that a query qualification for required data objects is based on a single attribute, i.e., the key attribute. In this paper, we focus on effective clustering of wireless data for

* Corresponding author.
E-mail address: mhkim@dbserver.kaist.ac.kr (M.H. Kim).

1383-7621/01/$ - see front matter
PII: S1383-7621(01)00005-4

Notation

n	the number of bits in a query signature
$\vec{q}$	a n-bit query signature; $\vec{q} = (b_n, b_{n-1}, \ldots, b_1)$
b_i	the value of the ith bit in $\vec{q}$; b_i is one of '0', '1', or '$*$'
l	the location of a bit specified by '0' or '1' in $\vec{q}$
un	the location of the left most bit (in $\vec{q}$) that has the value '$*$'
mlb	the location of the left most bit (in $\vec{q}$) that has the value '0' or '1'
$\mathrm{QD_B}(\vec{q})$	the query distance of $\vec{q}$ on the binary code
$\mathrm{QD_G}(\vec{q})$	the query distance of $\vec{q}$ on the Gray code

partial match queries. A partial match query retrieves data objects by specifying some non-key attributes [2,8,9]. In other words, the partial match query uses a content-based retrieval. It is known as a query type that has been widely used in various applications.

We propose a wireless data clustering method that utilizes some properties of the Gray code. The Gray code provides a sequence of codewords that has high locality. For the use of Gray codes, we represent data records as bit vectors using multi-attribute hashing. After describing the method, we analyze and evaluate the method by experiments.

The rest of the paper is organized as follows. In Section 2 we introduce the wireless data clustering and explain some properties of wireless data and the measure *query distance* (QD). We propose a clustering method for partial match queries in Section 3. In the method we use the multi-attribute hashing for the representation of data and query signatures. We analyze the proposed method in Section 4, and discuss on the ordering of attributes for signature representation in Section 5. In Section 6 we describe experimental results. In the last section we conclude this paper with future work.

2. Preliminaries

In this section, we give some background information on wireless data clustering, and introduce the QD measure, which is a wireless data clustering measure that will be used in the paper.

The clustering of wireless data is different from that of disk-resident data. The clustering of wireless broadcast data is to find a broadcast schedule (i.e., the sequence of data broadcasting) such that mobile clients can access the data on the air in short latency. The access time of a mobile query is determined by the duration from the query start time to the time of all required data records being downloaded.

Fig. 1 illustrates a wireless data broadcasting example, where five data records R_1–R_5 are broadcasted based on a broadcasting schedule $\sigma = \langle R_1, R_2, R_3, R_4, R_5 \rangle$. If a mobile client accesses the data $\{R_1, R_4\}$ at the time of R_2 in the figure, then the client completes its data retrieval at the end of R_1 in the next *bcast*, since the data record R_1 in the current *bcast* is passed over when the query starts. (In this paper, we assume there is no

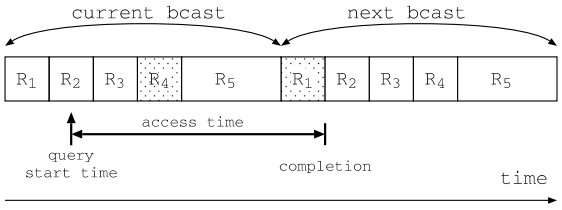

Fig. 1. Query processing on the wireless broadcast data.

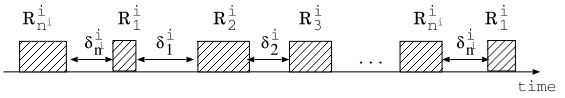

Fig. 2. Placement of the set of data of a query.

precedence relationship in accessing the data.) If the query issues before R_1 is broadcasted, then the query accesses both R_1 and R_4 in the current *bcast*.

Note that, since the clustering of wireless data determines a sequence of data broadcasting, it mainly affects the access time performance of mobile clients. The tuning time performance usually depends on index or caching methods. In this paper, we focus on the access time performance improvement through clustering of wireless data.

Since, as the figure shows, the access time of a mobile query depends on its start time, it is difficult to use the access time measure for developing clustering methods. Therefore, we use a new clustering measure named QD that was proposed in our previous work [4]. The QD of a query represents the coherence degree of data that the query accesses, and is independent of the query's start time.

Definition 1. Let q_i be a query that accesses data records $\{R_1^i, R_2^i, \ldots, R_{n^i}^i\}$ (n^i is the number of data records that q_i accesses), B be the size of one *bcast* and δ_j^i be the interval between two data records R_j^i and R_{j+1}^i in the given schedule σ. The QD of query q_i on schedule σ is (see Fig. 2)

$$\mathrm{QD}_\sigma(q_i) = B - \mathrm{MAX}(\delta_k^i), \quad k = 1, \ldots, n^i.$$

Definition 2. Given two schedules σ_1 and σ_2, if $\mathrm{QD}_{\sigma_1}(q_i)$ is equal to $QD_{\sigma_2}(q_i)$ for all queries q_i, then σ_1 is distance-equivalent to σ_2, denoted by $\sigma_1 \equiv \sigma_2$.

Property 1. Suppose that $\mathrm{QDS}(q_i)$ is the set of data that query q_i accesses, N is the total number of data records that are broadcasted in a single bcast and there are two schedules organized as follows:

$$\sigma_1 = \langle R_1, R_2, \ldots, R_{i-1}, R_i, R_{i+1}, \ldots, R_{j-1}, R_j, R_{j+1}, \ldots, R_{N-1}, R_N \rangle,$$
$$\sigma_2 = \langle R_1, R_2, \ldots, R_{i-1}, R_j, R_{i+1}, \ldots, R_{j-1}, R_i, R_{j+1}, \ldots, R_{N-1}, R_N \rangle.$$

Then, the following property holds.

$$\mathrm{QD}_{\sigma_1}(q_i) = \mathrm{QD}_{\sigma_2}(q_i) \quad \text{if } R_i, R_j \in \mathrm{QDS}(q_i).$$

Property 2. If a schedule σ_2 is the mirror image of σ_1 as in the below, then σ_1 is distance-equivalent to σ_2, i.e., $\sigma_1 \equiv \sigma_2$

$$\sigma_1 = \langle R_1, R_2, \ldots, R_{i-1}, R_i, R_{i+1}, \ldots, R_{j-1}, R_j, R_{j+1}, \ldots, R_{N-1}, R_N \rangle,$$
$$\sigma_2 = \langle R_N, R_{N-1}, \ldots, R_{j+1}, R_j, R_{j-1}, \ldots, R_{i+1}, R_i, R_{i-1}, \ldots, R_2, R_1 \rangle.$$

3. The Gray code clustering method

3.1. Gray codes

The Gray coding scheme is one of the linear mapping schemes for multi-dimensional space. In the binary reflected Gray code, numbers are coded into binary strings such that successive strings differ in

decimal	Gray code	binary code
0	000	000
1	001	001
2	011	010
3	010	011
4	110	100
5	111	101
6	101	110
7	100	111

Fig. 3. Illustration of the 3-bit binary reflected Gray code.

exactly one bit position. It is observed that the difference in only one bit position has a relationship with locality [5,7].

Fig. 3 illustrates 3-bit binary reflected Gray codes with corresponding binary codes. From now on, we use the term 'Gray code' for 'binary reflected Gray code' in this paper. The *Gray value* of a binary string is the *order* (or position) of the binary string in the Gray code. For instance, the Gray value of '110' is 4, which is equal to the *binary value* of '100'. In the figure the column 'decimal' represents the *Gray* and *binary* value of each codeword. The conversion of a Gray codeword to its order (i.e, Gray value) is described in [10].

3.2. Signature representation

In our method we represent data records and queries as data signatures and query signatures using multi-attribute hashing. We use a running example for explaining how data records and queries are represented as signatures based on multi-attribute hashing.

Suppose there is a wireless information system that broadcasts stock price information in real time. Each data record has four attributes: A_1 (company), A_2 (amount of sale), A_3 (amount of purchase) and A_4 (price: \$). For each attribute, we assume the following simple hash functions:

$$h_{\text{company}}(x) = \begin{cases} 00 & \text{if } x \text{ is a financial company,} \\ 01 & \text{if } x \text{ is a manufacturing company,} \\ 10 & \text{if } x \text{ is a computer and communication company,} \\ 11 & \text{otherwise,} \end{cases}$$

$$h_{\text{amount of sale}}(x) = \begin{cases} 00 & \text{if } x < 10\,000, \\ 01 & \text{if } 10\,000 \leqslant x < 20\,000 \\ 10 & \text{if } 20\,000 \leqslant x < 30\,000 \\ 11 & \text{if } 30\,000 \leqslant x, \end{cases}$$

$$h_{\text{amount of purchase}}(x) = \begin{cases} 00 & \text{if } x < 10\,000, \\ 01 & \text{if } 10\,000 \leqslant x < 20\,000 \\ 10 & \text{if } 20\,000 \leqslant x < 30\,000 \\ 11 & \text{if } 30\,000 \leqslant x, \end{cases}$$

$$h_{\text{price}}(x) = \begin{cases} 000 & \text{if } x < 5, \\ 001 & \text{if } 5 \leqslant x < 10, \\ 010 & \text{if } 10 \leqslant x < 20, \\ 011 & \text{if } 20 \leqslant x < 30, \\ 100 & \text{if } 30 \leqslant x < 40, \\ 101 & \text{if } 40 \leqslant x < 50, \\ 110 & \text{if } 50 \leqslant x < 60, \\ 111 & \text{if } 60 \leqslant x. \end{cases}$$

A data signature is generated by concatenating the hashed l_i-bit vector of each attribute i. Therefore the signature becomes l-bit binary bit vector, where $l = \sum_{i=1}^{k} l_i$ (k is the number of attributes). Here, we assume that the attributes are arranged $\langle A_1, A_2, A_3A_4 \rangle$ in this order, and $l_1, l_2, l_3 = 2$ and $l_4 = 3$. We will discuss on the arrangement of attributes and the bit allocation in Section 5.

Based on the above hashing scheme, we can represent a data record {company = 'New York Bank', amount of sale = '13 000', amount of purchase = '2000', price = '22'} as '010100011'. Similarly, we can represent a partial match query {amount of sale $\geqslant$ '30 000', '10' $\leqslant$ price $<$ '20'} as '$**11**010$', where '$*$' denotes a don't-care condition.

3.3. Clustering method

The proposed method consists of two steps: (1) representing data records as data signatures based on the given hash functions, (2) sorting data signatures based on their Gray values.

Suppose that there are seven data records R_1–R_7 (in the following table), and attributes and hash functions are the same as in Section 3.2. Then, the clustering method works as follows.

Id	Company	Amount of sale	Amount of purchase	Price ($)	Record signature	Gray value	Binary value
R_1	New York Bank	5000	35 000	69	000011111	21	31
R_2	LA Broadcasting	15 000	25 000	49	110110101	294	437
R_3	San Jose Computer	25 000	15 000	39	101001100	392	332
R_4	Dallas Machine	35 000	5000	29	011100011	190	227
R_5	Texas Electronics	500	45 000	19	010011010	236	154
R_6	Miami Industry	8000	31 000	4	010011000	239	152
R_7	Hawaii Comm.	24 000	19 000	32	101001100	392	332

In the first step of the proposed method, we represent the data record as record signature, which is described in the 'record signature' column of the table. In the second step, we sort the data based on their Gray values. As a result, we get the following schedules. σ_{Gray} is the result of the Gray coding method (GCM) and σ_{binary} is the result of the binary coding method (BCM). The BCM is the method which sorts the data signatures based on binary values.

$$\sigma_{\text{Gray}} = \langle R_1, R_4, R_5, R_6, R_2, R_3, R_7 \rangle,$$

$$\sigma_{\text{binary}} = \langle R_1, R_6, R_5, R_4, R_3, R_7, R_2 \rangle.$$

In the sorting step, the data records of the same Gray value or binary value are mutually interchangeable based on Property 1. (In this example R_3 and R_7 have the same Gray value.) Also, because of Property 2, it does not matter whether the set of data is sorted in an increasing order or decreasing one. In this paper, we sort the data in a non-decreasing order.

4. Analysis

First, we briefly show the clustering effects of the Gray coding scheme. Fig. 4 illustrates the 3-bit Gray and binary codewords, where the data records satisfied by partial match query '$*1*$'; are marked with '$\surd$' symbols. We can find that the Gray code clusters the data more effectively than the binary one, i.e., $\text{QD}_{\text{Gray}}(\text{`}*1*\text{'}) = 4$ and $\text{QD}_{\text{binary}}(\text{`}*1*\text{'}) = 6$.

Note that clustering of wireless broadcast data just determines the broadcasting sequence of data in the server. Thus, it does not incur additional overhead such as additional bandwidth or energy consumption.

Now, we analyze the QD of a partial match query when Gray and binary clustering methods are used. For convenience, we assume that each record signature corresponds to exactly one data record and vice versa. That is, if the number of bits is n, then it is assumed that there are 2^n data records each of which has a unique signature representation.

In the BCM, the QD of a query which is denoted by n-bit query signature $\vec{q}$ is as follows:

$$\text{QD}_{\text{B}}(\vec{q}) = 2^n - \sum_{\text{specfied bit locations } l} 2^{l-1}.$$

In the GCM, the QD of a query which is denoted by n-bit query signature $\vec{q}$ is as follows:

$$\text{QD}_{\text{G}}(\vec{q}) = 2^n - \sum_{\text{specfied bit locations } l} \phi(l)$$

decimal	Gray code		binary code	
0	000		000	
1	001		001	
2	011	$\surd$	010	$\surd$
3	010	$\surd$	011	$\surd$
4	110	$\surd$	100	
5	111	$\surd$	101	
6	101		110	$\surd$
7	100		111	$\surd$

Fig. 4. QD of partial match query '$*1*$'.

$$\phi(l) = \begin{cases} 2^{n-1} & \text{if } l = n, \quad \text{(a)} \\ 2^{l} & \text{if } l = \text{mlb and } l \neq n, \quad \text{(b)} \\ 2^{l-1} & \text{if } l \neq \text{mlb and } l > \text{un}, \quad \text{(c)} \\ 0 & \text{if } l \neq \text{mlb},\ l < \text{un},\ b_l = 0 \text{ and } b_{l+1} = \text{`}*\text{'}, \quad \text{(d)} \\ 2^{l} & \text{if } l \neq \text{mlb},\ l < \text{un},\ b_l = 1 \text{ and } b_{l+1} = \text{`}*\text{'}, \quad \text{(e)} \\ 2^{l} & \text{if } l \neq \text{mlb},\ l < \text{un},\ b_l = 0,\ l+1 = \text{mlb and } l+1 \neq n, \quad \text{(f)} \\ 0 & \text{if } l \neq \text{mlb},\ l < \text{un},\ b_l = 1,\ l+1 = \text{mlb and } l+1 \neq n, \quad \text{(g)} \\ 0 & \text{if } l \neq \text{mlb},\ l < \text{un},\ b_l = 0,\ l+1 \neq \text{mlb and } \phi(l+1) = 0, \quad \text{(h)} \\ 2^{l} & \text{if } l \neq \text{mlb},\ l < \text{un},\ b_l = 0,\ l+1 \neq \text{mlb and } \phi(l+1) = 2^{l+1}, \quad \text{(i)} \\ 2^{l} & \text{if } l \neq \text{mlb},\ l < \text{un},\ b_l = 1,\ l+1 \neq \text{mlb and } \phi(l+1) = 0, \quad \text{(j)} \\ 0 & \text{if } l \neq \text{mlb},\ l < \text{un},\ b_l = 1,\ l+1 \neq \text{mlb and } \phi(l+1) = 2^{l+1}. \quad \text{(k)} \end{cases} \tag{1}$$

Example 1. Suppose that the number of bits (n) is 5 and a partial match query has the form of '$*0*10$'. Then the $\mathrm{QD_B}$ and $\mathrm{QD_G}$ of the query are as follows:

$$\mathrm{QD_B}(\vec{q} = (*, 0, *, 1, 0)) = 2^5 - \sum_{l=1,2,4} 2^{l-1} = 32 - (8 + 2 + 1) = 21,$$

$$\mathrm{QD_G}(\vec{q} = (*, 0, *, 1, 0)) = 2^5 - \sum_{l=1,2,4} \phi(l) = 32 - (16 + 4 + 2) = 10. \quad \square$$

By using the formulas for $\mathrm{QD_B}$ and $\mathrm{QD_G}$, we analyze the average QD gain of the GCM over the BCM ($\mathrm{AQD_{gain}}(n)$) as follows: (For details, see Appendix A.)

$$\mathrm{AQD_{gain}}(n) = \frac{2(6^{n-1} - 1)}{5(3^n - 2^n)}.$$

5. The optimal attribute ordering

In the previous sections, attributes are assumed to be referenced with equal probabilities. However, if the reference probability of each attribute is not the same, we have to consider the following two issues in the signature representation of the data record:

1. The number of bits for each attribute.
2. The order of attributes in the record signature.

The first issue was studied in [2], and hence we can use its result without much modification. However, the second issue has not been studied in the past. Although [11] studied the ordering of attributes in the Attribute Gray (AG) encoding scheme, the work is for the disk-based environment. Thus, in this section, we propose a method that effectively finds the ordering of attributes for the wireless environment.

The reference probability of attribute A_i is denoted by $\mathrm{ref}(A_i)$. We assume an attribute referenced when the value of the attribute is specified '0' or '1'. Thus, the probability that the value of attribute A_i is '$*$' is $1 - \mathrm{ref}(A_i)$. We use the the weighted sum of QD (WSQD) for the criteria of the good attribute ordering. The WSQD is defined as $\sum_{\forall \vec{q}} \mathrm{prob}(\vec{q}) \times \mathrm{QD}(\vec{q})$. Here, $\mathrm{prob}(\vec{q})$ is defined as $\prod_i \tau(A_i, \vec{q})$, where $\tau(A_i, \vec{q})$ is $1/2 \cdot \mathrm{ref}(A_i)$ if the value of A_i in $\vec{q}$ is specified '0' or '1', and otherwise $1 - \mathrm{ref}(A_i)$. In the definition of

Ordering	WSQD in the GCM	WSQD in the BCM
$< A_1, A_2, A_3 >$	4.668	5.644
$< A_1, A_3, A_2 >$	4.128	5.344
$< A_2, A_1, A_3 >$	4.668	5.244
$< A_2, A_3, A_1 >$	3.868	4.744
$< A_3, A_1, A_2 >$	4.128	4.344
$< A_3, A_2, A_1 >$	3.868	4.144

Fig. 5. WSQD of different attribute orderings: An example.

n	GCM	BCM	RCM	OPT
2	2.40	2.80	2.72	2.40
3	3.58	4.31	4.19	3.53
4	5.91	7.23	7.97	-
5	10.45	12.90	15.74	-
6	19.34	24.02	33.40	-
7	36.90	45.97	70.16	-
8	71.69	89.45	151.43	-
9	140.84	175.88	323.78	-
10	278.55	348.02	686.15	-
11	553.18	691.31	1446.32	-
12	1101.42	1376.60	3032.40	-
13	2196.48	2745.44	6317.40	-

Fig. 6. AQD of all partial match queries.

WSQD, $QD(\vec{q})$ is the QD on the given schedule, i.e., it is $QD_G(\vec{q})$ in the GCM, $QD_B(\vec{q})$ in the BCM, and so on. For generality, we assume that each attribute is represented by one bit in the signature, i.e., an attribute A_i is represented as a bit b_i in the signature. We also assume that attributes are independently specified.

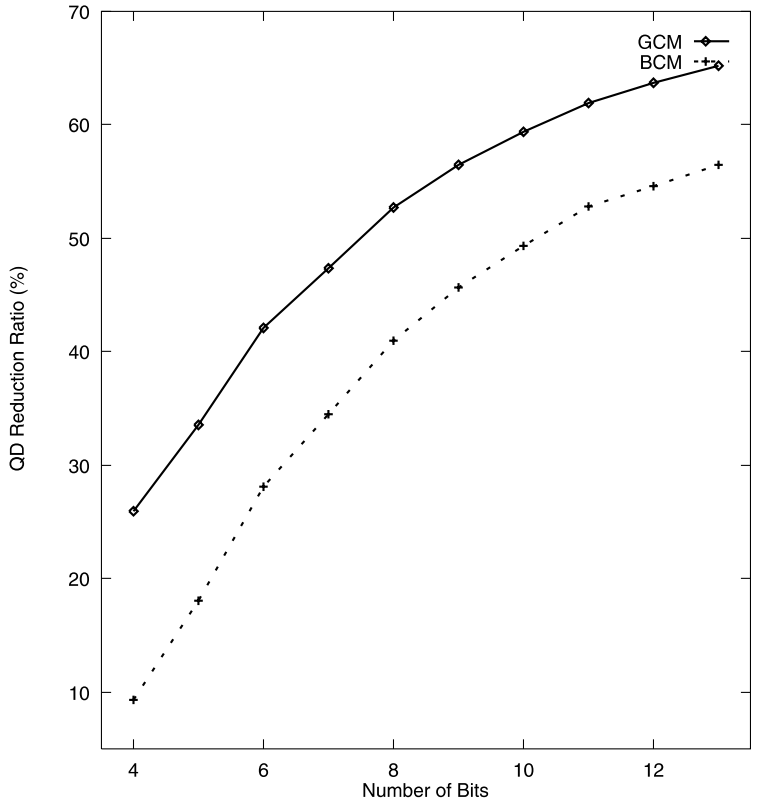

Fig. 7. QD reduction ratios of GCM and BCM.

We show the effects of attribute ordering with a simple example. Suppose that the number of data records is 8 and each data record has three attributes A_1, A_2 and A_3. The reference probability of each attribute is as follows: $\mathrm{ref}(A_1) = 0.2$, $\mathrm{ref}(A_2) = 0.4$ and $\mathrm{ref}(A_3) = 0.7$. So, in representing the data signature, 6 (i.e., 3!) orderings are possible. Fig. 5 shows the WSQD of 6 orderings in the GCM and BCM.

Now we describe the optimal arrangement (i.e., ordering) of attributes when using the GCM and BCM. An attribute ordering is optimal if its WSQD is minimum among all possible orderings.

Theorem 1. *Let $\langle A_{[n]}, A_{[n-1]}, \ldots, A_{[2]}, A_{[1]} \rangle$ be an arrangement of attributes for n-bit signature representation. In the BCM and GCM, an arrangement of attributes is optimal if the following condition holds:*

$$\mathrm{ref}(A_{[i+1]}) \geqslant \mathrm{ref}(A_{[i]}), \quad 1 \leqslant i \leqslant n-1.$$

Proof. See Appendix B. □

The theorem says that, when we represent data records and queries into signatures, we should place attributes with high reference probabilities at significant bits in the signature. For example, in Fig. 5, since the reference probabilities of attributes A_1, A_2 and A_3 are 0.2, 0.4 and 0.7, respectively, the attribute arrangement $\langle A_3, A_2, A_1 \rangle$ in this order gives the best performance.

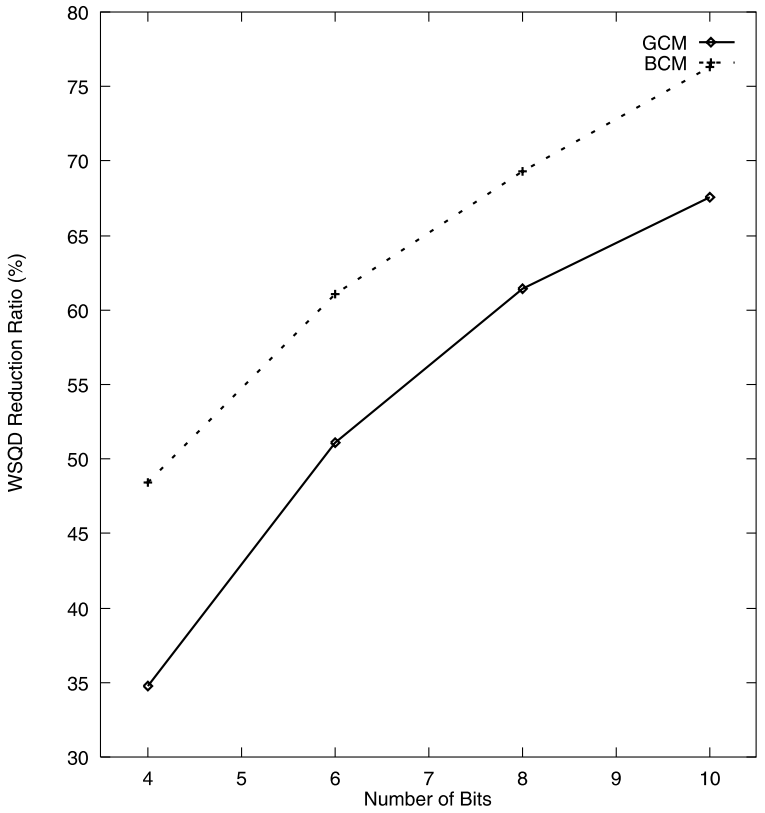

Fig. 8. Comparison of the optimal and worst attribute ordering.

6. Experimental results

In this section we describe our two experiments and their results. The first experiment verifies the correctness of formulas in Section 4, and the second experiment compares the optimal attribute ordering (in Section 5) with other orderings.

Fig. 6 shows the result of our first experiment. In the experiment we compare four clustering methods: the GCM, the BCM, a random clustering method (RCM) and the exhaustive search (i.e., optimal) method (OPT). By the result we can verify our analysis formulas described previously. (For instance, when the number of bits n is 5, $\mathrm{AQD}_{\mathrm{gain}}(5) = \mathrm{QD}^{\mathrm{avg.}}_{\mathrm{binary}}(5) - \mathrm{QD}^{\mathrm{avg.}}_{\mathrm{Gray}}(5) = 12.90 - 10.45 = 2.45$.) In addition, although our experiment for the exhaustive search method was done only for small numbers of bits due to the high computational complexity, [1] we find the performance of the proposed method is close to that of the optimal method.

Fig. 7 shows the QD reduction ratios of the GCM and BCM compared with the RCM. With the increase of the number of bits, the proposed method significantly reduces the QD compared with the RCM.

[1] The complexity of n-bit experiment is $\mathrm{O}(\mathrm{NlogN})$, where $\mathrm{N} = (2^{\mathrm{n}})!$

In the second experiment, we evaluate the effect of attribute ordering in the signature representation. For the reference probability of each attribute, we assume a normal distribution. Fig. 8 shows the result of our experiment. The measurement is the WSQD reduction ratio of the optimal attribute arrangement compared with the worst attribute arrangement i.e., the reverse ordering of Theorem 1.

7. Conclusion

Most of the previous work on wireless data broadcasting considers the data retrieval based on a single attribute, i.e., primary key, such as the primary key indexing and the data scheduling for the primary key. However, our work considers the multi-key data retrieval (i.e, content-based retrieval) environment, especially the data clustering for partial match queries.

In this paper, we proposed a wireless data clustering method for partial match queries by using the Gray code concept. In the method we represent the data record (and the query) as the record (and the query) signature through multi-attribute hashing. The method clusters the data based on their Gray values. We showed the clustering characteristics of Gray codes for wireless data and evaluated the performance in analytic and experimental ways. We also proposed an optimal attribute ordering strategy for signature representation.

The wireless communication is less reliable compared with the wireline one. Thus, in the future, we will extend the proposed clustering method considering unreliable communication environments. We will also investigate the clustering method for other query types such as range queries.

Acknowledgements

This work was supported by grant No. 1999-1-303-007-3 from the Interdisciplinary Research Program of the KOSEF.

Appendix A. Derivation of $AQD_{gain}(n)$

We describe some notations for deriving $AQD_{gain}(n)$. $TSQD_B(n)$ is the total sum of the QD of all n-bit partial match queries on the binary code. $TSQD_G(n)$ is the total sum of the QD of all n-bit partial match queries on the Gray code. And, $TSQD_{gain}(n)$ is $TSQD_B(n) - TSQD_G(n)$.

In the formulas for $\phi(l)$ (in Section 4), the Eqs. (1d) and (1e), (1f) and (1g), (1h) and (1i), and (1j) and (1k) are formed in a pairwise way. Thus, we can compute the difference between $TSQD_B(n)$ and $TSQD_G(n)$ by summing up the cases of Eq. (1b). In consequence, the $TSQD_{gain}(n)$ is computed as follows:

$$TSQD_{gain}(n) = TSQD_B(n) - TSQD_G(n) = \sum_{\forall \vec{q}} (QD_B(\vec{q}) - QD_G(\vec{q})) = \sum_{i=1}^{n-1} ((2 \times 3^{i-1}) \times 2^{i-1})$$

$$= \sum_{i=1}^{n-1} 2^i \times 3^{i-1}.$$

The $AQD_{gain}(n)$ is $TSQD_{gain}(n)$ divided by the number of all n-bit partial match queries, and there are $3^n - 2^n$ partial match queries in the n-bit signature. Thus, $AQD_{gain}(n)$ is computed as follows:

$$AQD_{gain}(n) = \frac{TSQD_{gain}(n)}{3^n - 2^n} = \frac{\sum_{i=1}^{n-1} 2^i \times 3^{i-1}}{3^n - 2^n} = \frac{2(6^{n-1} - 1)}{5(3^n - 2^n)}.$$

Appendix B. Proof of Theorem 1

B.1. In the BCM

The QD of a n-bit query $\vec{q}$ is $2^n - \sum_l 2^{l-1}$, where l the locations of specified bits. Because the QD is determined only by specified bit values (i.e., '0' and '1'), we can compute the $\mathrm{TSQD_B}$ as follows:

$$\mathrm{TSQD_B}(n) = (3^n - 2^n) \times 2^n - 2(3^{n-1} - 2^{n-1}) \times \sum_{i=1}^{i=n} 2^{i-1}.$$

By applying the reference probability of each attribute, we can compute the $\mathrm{WSQD_B}$ as follows:

$$\mathrm{WSQD_B}(n) = (3^n - 2^n) \times 2^n - (3^{n-1} - 2^{n-1}) \times \sum_{i=1}^{i=n} 2^{i-1} \times \mathrm{ref}(A_{[i]}).$$

Since the terms '$(3^n - 2^n) \times 2^n$' and '$(3^{n-1} - 2^{n-1})$' are positive constants, we consider on maximizing the term '$\sum_{i=1}^{i=n} 2^{i-1} \times \mathrm{ref}(A_{[i]})$', which is described as $2^{n-1} \times \mathrm{ref}(A_{[n]}) + 2^{n-2} \times \mathrm{ref}(A_{[n-1]}) + \cdots + 2^1 \times \mathrm{ref}(A_{[2]}) + \mathrm{ref}(A_{[1]})$.

Therefore, based on the greedy philosophy, we can see the above formula is maximized when $\mathrm{ref}(A_{[i+1]}) \geqslant \mathrm{ref}(A_{[i]})$ for all i, $1 \leqslant i \leqslant n-1$.

B.2. In the GCM

Since the formula $\mathrm{QD_G}\vec{q}$ (in Section 4) is complex, we compute the $\mathrm{TSQD_G}(n)$ as follows:

$$\mathrm{TSQD_G}(n) = \mathrm{TSQD_B}(n) - \mathrm{TSQD_{gain}}(n) = (3^n - 2^n) \times 2^n - 2(3^{n-1} - 2^{n-1}) \times \sum_{i=1}^{i=n} 2^{i-1} - \sum_{i=1}^{i=n-1} 2^i \cdot 3^{i-1}.$$

Thus, by adding the probability factor in the above formula, we compute the $\mathrm{WSQD}_G(n)$ as follows:

$$\mathrm{WSQD_G}(n) = (3^n - 2^n) \times 2^n - ((3^{n-1} - 2^{n-1}) \times \sum_{i=1}^{i=n} 2^{i-1} \cdot \mathrm{ref}(A_{[i]}) - \frac{1}{2} \sum_{i=1}^{i=n-1} 2^i \cdot 3^{i-1} \cdot \mathrm{ref}(A_{[i]}).$$

Since '$(3^n - 2^n) \times 2^n$' is a positive constant, we consider on maximizing the sum of the other two terms:

$$\begin{aligned}
&(3^{n-1} - 2^{n-1}) \times \sum_{i=1}^{i=n} 2^{i-1} \cdot \mathrm{ref}(A_{[i]}) + \frac{1}{2} \sum_{i=1}^{i=n-1} 2^i \cdot 3^{i-1} \cdot \mathrm{ref}(A_{[i]}) \\
&\quad = (3^{n-1} - 2^{n-1}) \times \sum_{i=1}^{i=n} 2^{i-1} \cdot \mathrm{ref}(A_{[i]}) + \sum_{i=1}^{i=n-1} 6^{i-1} \cdot \mathrm{ref}(A_{[i]}) \\
&\quad = ((3^{n-1} - 2^{n-1}) \cdot 2^0 + 6^0)\mathrm{ref}(A_{[1]}) + ((3^{n-1} - 2^{n-1}) \cdot 2^1 + 6^1)\mathrm{ref}(A_{[2]}) + ((3^{n-1} - 2^{n-1}) \cdot 2^2 \\
&\qquad + 6^2)\mathrm{ref}(A_{[3]}) + \cdots + ((3^{n-1} - 2^{n-1}) \cdot 2^{n-2} + 6^{n-2}) \cdot \mathrm{ref}(A_{[n-1]}) + (3^{n-1} - 2^{n-1}) \cdot 2^{n-1} \cdot \mathrm{ref}(A_{[n]}).
\end{aligned}$$

Since $((3^{n-1} - 2^{n-1}) \cdot 2^{n-1})$ is greater than $((3^{n-1} - 2^{n-1}) \cdot 2^{n-2} + 6^{n-2})$, the above equation is maximized when the attributes are arranged as $\mathrm{ref}(A_{[i]}) \geqslant \mathrm{ref}(A_{[i-1]})$ for i, $2 \leqslant i \leqslant n$. □

References

[1] S. Acharya, R. Alonso, M. Franklin, S. Zdonik, Broadcast disks: data management for asymmetric communication environments, in: Proceedings of ACM SIGMOD Conference, 1995, pp. 199–210.
[2] A.V. Aho, J.D. Ullman, Optimal partial match retrieval when fields are independently specified, ACM Trans. Database Systems 4 (2) (1979) 168–179.
[3] Y.D. Chung, M.H. Kim, An index replication scheme for wireless data broadcasting, J. Systems Software 51 (2000) 191–199.
[4] Y.D. Chung, M.H. Kim, Effective data placement for wireless broadcast, Distributed and Parallel Databases (to appear).
[5] C. Faloutsos, Multiattribute hashing using Gray codes, in: Proceedings of ACM SIGMOD Conference, 1986, pp. 227–238.
[6] T. Imielinski, S. Viswanathan, B.R. Badrinath, Data on air: organization and access, IEEE Trans. Knowledge Data Eng. 9 (3) (1997).
[7] H.V. Jagadish, Linear clustering of objects with multiple attributes, in: Proceedings of ACM SIGMOD Conference, 1990, pp. 332–342.
[8] M.H. Kim, S. Pramanik, Optimal file distribution for partial match retrieval, in: Proceedings of ACM SIGMOD Conference, 1988, pp. 173–182.
[9] S. Moran, On the complexity of designing optimal partial-match retrieval systems, ACM Trans. Database Systems 8 (4) (1983) 543–551.
[10] E.M. Reingold, J. Nievergelt, N. Deo, Combinatorial Algorithms: Theory and Practice, Prentice-Hall, Englewood Cliffs, NJ, 1977.
[11] D. Rotem, Clustered multiattribute hash files, in: Proceedings of ACM PODS Conference, 1989, pp. 225–234.
[12] C. Su, L. Tassiulas, V.J. Tsotras, Broadcast scheduling for information distribution, Wireless Networks (1998).

Ji Yeon Lee received her B.S. degree in Computer Engineering from Dong Kuk University, Seoul, Korea, in 1993, and her M.S. and Ph.D. degrees in Computer Science from Korea Advanced Institute of Science and Technology (KAIST), Teajon, Korea, in 1996 and 2001, respectively. She is currently a researcher at Hyundai Information Technology Inc. Seoul, Korea. Her research interests include mobile computing, wireless information systems and spoken language engineering.

Yon Dohn Chung received his B.S. degree in Computer Science from Korea University, Seoul, Korea, in 1994, and his M.S. and Ph.D. degrees in Computer Science from Korea Advanced Institute of Science and Technology (KAIST), Teajon, Korea, in 1996 and 2000, respectively. He is currently a post-doctoral fellow in the Department of Computer Science at KAIST. His research interests include mobile computing, OLAP, data warehouse, and semi-structured data management.

Yoon Joon Lee received his B.S degree in Computer Science and Statistics from Seoul National University, Seoul, Korea, in 1977, his M.S. degree in Computer Science from Korea Advanced Institute of Science and Technology (KAIST), Taejon, Korea, in 1979, and his Ph.D. degree in Computer Science from INPG-ENSIMAG, France, in 1983. In 1984, he joined the faculty of the Department of Computer Science at KAIST, Taejon, Korea, where currently he is a professor. His research interests include OLAP, data warehouse, multimedia information systems, and database systems. He is a member of the ACM and the IEEE Computer Society.

Myoung Ho Kim received his B.S. and M.S. degrees in Computer Engineering from Seoul National University, Seoul, Korea, in 1982 and 1984, respectively, and his Ph.D. degree in Computer Science from Michigan State University, East Lansing, MI, in 1989. In 1989, he joined the faculty of the Department of Computer Science at KAIST, Taejon, Korea, where currently he is a professor. His research interests include database systems, data mining, information retrieval, OLAP, mobile computing and distributed processing. He is a member of the ACM and the IEEE Computer Society.