Faculty Working Papers

ON PROBLEMS OF DATA BASE SEGMENTATION

Hirohide Hinomoto

#289

College of Commerce and Business Administration

University of Illinois at Urbana-Champaign

FACULTY WORKING PAPERS

College of Commerce and Business Administration

University of Illinois at Urbana-Champaign

November 19, 1975

ON PROBLEMS OF DATA BASE SEGMENTATION

Hirohide Hinomoto

#289

ON PROBLEMS OF DATA BASE SEGMENTATION

Hirohide Hinomoto

Department of Business Administration
University of Illinois
Urbana-Champaign, Illinois 61801

Abstract

The manner in which the data base is segmented affects the probability of intent conflict and the time required for retrieving necessary fields. This paper examines a basic segmentation problem in which two fields in a data base are required in different manners by three messages entering into different terminals. The problem is interpreted as a finite state Markov chain and formulated into two alternative methods of segmentation.

Introduction

A "segment" may be a field or a group of fields that can be retrieved from a data base as a unit of data by a data base management system. With systems such as IMS Version 2, the concurrent updating of segments belonging to the same type by more than one message creates an intent conflict in which only one message can have access to the segment and the remaining messages have to wait in queue for their turns. With those systems, the intent conflict usually represents a major cause of deterioration in the response time to a terminal message during a peak load period. Obviously, the manner of segmenting a data base will importantly affect the probability of intent conflict as well as the time required for retrieving necessary fields. This study examines a basic segmentation problem in which two fields are required in different manners by three messages from different terminals. The problem is interpreted as a finite state Markov chain and formulated under two alternative methods of segmentation.

Alternative Cases

Consider a situation in which three terminal x, y, and z that send in messages concerning two fields a and b in a data base. Specifically, terminal x's messages need only field a, terminal y's messages only field b, and terminal z's messages both fields a and b. There are two ways to segment the data base with fields a and b, as is illustrated by Cases I and II in Figure 1; in Case I, fields a and b are assigned to two different segments whereas in Case II both fields are assigned to the same segment. Regardless of the method of segmentation, messages from terminal

z always call for fields a and b in that order. It is assumed that during the peak load period each terminal is saturated with messages; the operator of a terminal can key in a new message as soon as the processing of the preceding message is completed; and the keyed-in message waits in the system at least an epoch before being processed. In any event, only one message from a terminal can be in the system at any time.

Two types of messages are considered; one type is query and the other type update. Let $d_g(q)$ and $d_g(u)$ represent the proportions of query and update messages sent by terminal g, $g \varepsilon G = \{x, y, z\}$. Then we have

$$(1) \qquad d_g(q) + d_g(u) = 1 \qquad g \varepsilon G$$

The processing of a query message sent by a terminal can be carried out independent of the current messages from the other terminals. But update messages from different terminals trying to process the same segment type will cause intent conflict. In this case, priorities in message processing are in the order of messages from terminals z, y, and x.

Case I: Two Fields Assigned to Different Segments

In this case, terminal z must always key in a pair of messages to process fields a and b, in that order.

The time required for searching and processing a field is simply called the process time and assumed as a stochastic variable with an exponential distribution function given by $e^{-\alpha t}$ where α is the probability of completing the process given by a reciprocal of a mean process time. The mean process time and the process completion probability for terminal g, $g \varepsilon G$, are represented by $t_g(m)$ and $\alpha_g(m)$ for the message type m given as an element of M, $M = \{q, u\}$ where q represent a query and u an update.

For terminal z, m is subscripted with h, an element of set H = {a, b}, to indicate the field involved. Thus we write:

$$
(2)\qquad
\begin{aligned}
&\alpha_x(m) = \alpha_z(m_a) = 1/t_x(m) \\
&\alpha_y(m) = \alpha_z(m_b) = 1/t_y(m) \qquad\qquad m \in M
\end{aligned}
$$

Since the terminal can put in only one message in the system at any time, its status is determined by the type and current state of its message in the system. Since terminals x and y deal with only one field, their statuses are given by elements of set $I = \{q, q', u, u'\}$ where q or u represent "a query or an update in process", and q' or u' "a query or an update in queue". The status of terminal z is given by an element of set $K = \{q_h, q_h', u_h, u_h\}$ where h is an element of set H = {a, b}, indicating the field currently involved.

Let $P_g[s_1, s_2]$ represent the transition probability with which the status of terminal g changes from s_1 at an epoch to s_2 at the next epoch. If we let k_1 and k_2 represent s_1 and s_2 for terminal z, transition probabilities of terminals x and y are given as follows:

$$
(3)\qquad
\begin{aligned}
&P_g[q', q] = 1 && P_g[q, q] = 1 - \alpha_g(q) \\
&P_g[q, q'] = \alpha_g(q)d_g(q) && P_g[q, u'] = \alpha_g(q)d_g(u) \\
&P_g[u, u] = 1 - \alpha_g(u) && P_g[u, q'] = \alpha_g(u)d_g(q) \\
&P_g[u, u'] = \alpha_g(u)d_g(u) && g \in G_1 = \{x, y\} \\
&P_x[u', u'] = 1 \quad \text{if} \quad k_1 = k_2 = u_a \vee k_1 = u_a' \\
&P_x[u', u] = 1 \quad \text{if} \quad \sim(k_1 = k_2 = u_a) \vee k_1 \neq u_a'
\end{aligned}
$$

$$P_y[u_a', u'] = 1 \quad \text{if} \quad k_1 = k_2 = u_b \vee k_1 = u_b'$$

$$P_y[u_a', u] = 1 \quad \text{if} \quad \sim(k_1 = k_2 = u_b) \vee k_1 \neq u_b'$$

Otherwise $P_g[s_1, s_2] = 0$, $g \in G_1$, for the above and all other combinations of s_1 and s_2 (s_1, $s_2 \in I$).

For terminal z, transition probabilities are given by the following where i_1 and i_2 represent s_1 and s_2 for terminal x, and j_1 and j_2 those for terminal y:

$$P_z[q_h', q_h] = 1 \qquad P_z[q_h, q_h] = 1 - \alpha_z(q_h)$$

$$P_z[q_a, q_b'] = \alpha_z(q_a), \qquad P_z[q_b', q_a'] = \alpha_z(q_b)d_z(q)$$

$$(4) \qquad P_z[q_b, u_a'] = \alpha_z(q_b)d_z(u) \qquad P_z[u_h, u_h] = 1 - \alpha_z(u_h)$$

$$P_z[u_a, u_b'] = \alpha_z(u_a) \qquad P_z[u_b, q_a'] = \alpha_z(u_b)d_z(q)$$

$$P_z[u_b, u_a'] = \alpha_z(u_b)d_z(u) \qquad h \in H = \{a, b\}$$

$$P_z[u_a', u_a'] = 1 \quad \text{if} \quad i_1 = i_2 = u_a$$

$$P_z[u_a', u_a] = 1 \quad \text{if} \quad \sim(i_1 = i_2 = u_a)$$

$$P_z[u_b', u_b'] = 1 \quad \text{if} \quad j_1 = j_2 = u_b$$

$$P_z[u_b', u_b] = 1 \quad \text{if} \quad \sim(j_1 = j_2 = u_b)$$

Otherwise $P_z[s_1, s_2] = 0$ for the above and all other combinations of s_1 and s_2 ($s_1, s_2 \in K$).

Let $S[i, j, k]$ be the state of the system in which the statuses of terminals x, y, and z are i, j, and k, and $X[i, j, k]$ be the probability

of the system being in that state. Then we have

$$(5)\qquad \sum_i \sum_j \sum_k X[i, j, k] = 1$$

$$X[i, j, k] \geq 0 \qquad i, j \in I;\ k \in K$$

Let $S[i_1, j_1, k_1]$ and $S[i_2, j_2, k_2]$ be the states of the system at two consecutive epochs. Then, a stationary equilibrium exists between each state, $S[i_2, j_2, k_2]$, and various states, $S[i_1, j_1, k_1]$, leading to this state. This is formulated with transition probabilities in (3) and (4):

$$(6)\qquad \sum_{i_1 \in I} \sum_{j_1 \in J} \sum_{k_1 \in K} P_x[i_1, i_2]\, P_y[j_1, j_2]\, P_z[k_1, k_2]\, X[i_1, j_1, k_1] - X[i_2, j_2, k_2] = 0 \qquad i_2, j_2 \in I;\ k_2 \in K$$

Since we have one equation for each of the possible states $S[i, j, k]$, a solution to the set of equations in (6) in terms of $X[i, j, k]$ gives us the values of stationary probabilities for the system to be found in $S[i, j, k]$. From the stationary probabilities thus determined, the probability of terminal x, y, or z being in the status i, j, or k, denoted by $r_x(i)$, $r_y(j)$, or $r_z(k)$, is determined as follows:

$$r_x(i) = \sum_{j \in I} \sum_{k \in K} X[i, j, k]$$

$$(7)\qquad r_y(j) = \sum_{i \in I} \sum_{k \in K} X[i, j, k] \qquad i, j \in I$$

$$r_z(k) = \sum_{i \in I} \sum_{j \in I} X[i, j, k] \qquad k \in K$$

Using the average processing rates in (2) and the status probabilities in (7), we determine the expected response time to message m sent by

terminal a or b as follows:

(8) $$T_g(m) = \{r_g(m) + r_g(m')\}\, t_g(m)/r_g(m) \qquad m \in M,\; g \in G_1$$

Similarly, the expected response time to a message sent by terminal z is given by :

(9) $$T_z(m) = \{r_z(m_a) + r_z(m_a')\}\, t_z(m_a)/r_z(m_a) + \{r_z(m_b) + r_z(m_b')\}\, t_z(m_b)/r_z(m_b) \qquad m \in M$$

Case II: Two Fields Assigned to the Same Segment

In this case, fields a and b are assigned to the same segment, and therefore one message from terminal z can process both fields. This enables us to represent the status of terminal z as well as those of terminals x and y by an element of set I, $I = \{q, q', u, u'\}$. An intent conflict will be created if any two terminals send update messages concurrently, because their fields now belong to the same segment.

Let the average process time and the probability of process completion for message m sent by terminal g be given by $t_g(m)$ and $\alpha_g(m)$, then we have

(10) $$\alpha_g(m) = 1/t_g(m) \qquad g \in G, \quad m \in M$$

When there is no intent conflict, the transition probabilities of all terminals in this case are given by the following common forms:

(11) $$P_g[q, q] = 1 - \alpha_g(q), \qquad P_g[q, q'] = \alpha_g(q)d_g(q),$$
$$P_g[q, u'] = \alpha_g(q)d_g(u), \qquad P_g[u, u] = 1 - \alpha_g(u),$$

$$P_g[u, q'] = \alpha_g(u)d_g(q), \qquad P_g[u, u'] = \alpha(u)d_g(u),$$

$$P_g[q', q] = 1 \qquad g \varepsilon G$$

When there is an intent conflict, the transition probability of a terminal is given by one of the following:

$$
(12) \quad
\begin{aligned}
&P_x[u', u'] = 1 && \text{if} \quad (j_1 = u \wedge j_2 = u) \vee j_1 = u' \vee (k_1 = u \wedge k_2 = u) \vee k_1 = u' \\
&P_x[u', u] = 1 && \text{if} \quad \sim(j_1 = u \wedge j_2 = u) \vee j_1 \neq u' \vee \sim(k_1 = u \wedge k_2 = u) \vee k_1 \neq u' \\
&P_y[u', u'] = 1 && \text{if} \quad (i_1 = u \wedge i_2 = u) \vee (k_1 = u \wedge k_2 = u) \vee k_1 = u' \\
&P_y[u', u] = 1 && \text{if} \quad \sim(i_1 = u \wedge i_2 = u) \vee \sim(k_1 = u \wedge k_2 = u) \vee k_1 \neq u' \\
&P_z[u', u'] = 1 && \text{if} \quad (i_1 = u \wedge i_2 = u) \vee (j_1 = u \wedge j_2 = u) \\
&P_z[u', u] = 1 && \text{if} \quad \sim(i_1 = u \wedge i_2 = u) \vee (j_1 = u \wedge j_2 = u)
\end{aligned}
$$

Otherwise $P_g[s_1, s_2] = 0$, $g \varepsilon G$, for the above and all other combinations of s_1 and $s_2 (s_1, s_2 \varepsilon I)$.

In the present case, the stationary equilibrium condition takes the following form :

$$(13) \quad \sum_{i_1 \varepsilon I} \sum_{j_1 \varepsilon I} \sum_{k_1 \varepsilon K} P_x[i_1, i_2] \, P_y[j_1, j_2] \, P_z[k_1, k_2] \, X[i_1, j_1, k_1]$$

$$- X[i_2, j_2, k_2] = 0 \qquad i_2, j_2, k_2 \varepsilon I$$

subject to

$$\sum_i \sum_j \sum_k X[i, j, k] = 1$$

$$X[i, j, k] \geqq 0 \qquad i, j, k \varepsilon I$$

Using a solution to the set of equations in (13), the expected response time to a message from each terminal is determined by the formulas in (7) - (9).

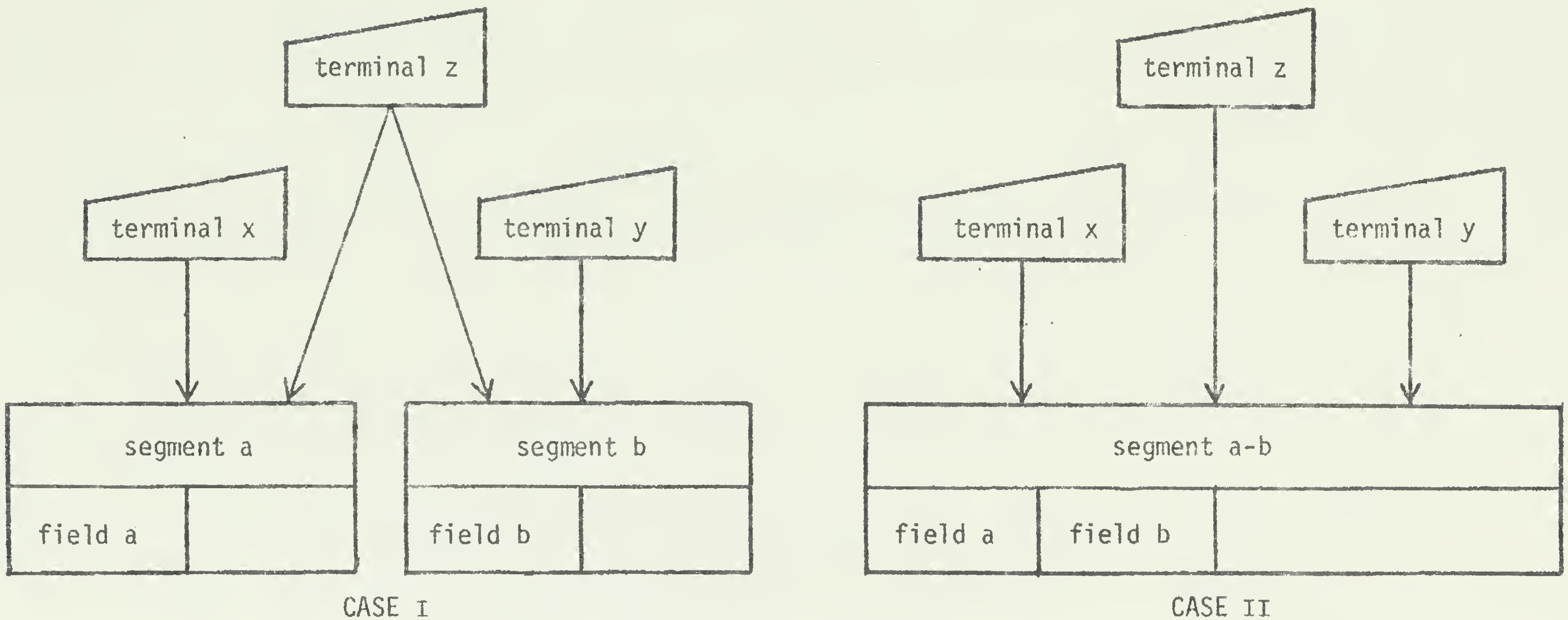

Figure 1. Alternative Methods of Segmentation