Proceedings of International Conference
on Networking and Multimedia

Performance Study of Buffer Control Schemes During Congestion in a Shared Buffer ATM Switch

Chie Dou and Jeng-Shin Sheu
Department of Electrical Engineering
National Yunlin Institute of Technology
640 Touliu, Yunlin, Taiwan, R.O.C.

Abstract

This paper considers overload control during congestion in a shared buffer ATM switch. To prevent performance degradation of the shared buffer memory switch under imbalance traffic conditions, a "gated" cell discarding policy is proposed. The concept of the "gated" policy is that by adding a control gate in front of the logical queue pertaining to each overloaded output port, some incoming cells destined for the overloaded ports can be blocked so making rooms in the shared buffer for accommodating incoming cells destined for the non-overloaded ports. Numerical results show that the proposed "gated" scheme can not only satisfy the "fair" access requirement under network congestion conditions, but also reduce the total number of discarded cells thus increases overall performance.

1. Introduction

Asynchronous transfer mode (ATM) is a promising technique proposed by CCITT to provide the required flexibility for supporting heterogeneous services in a Broadband ISDN environment. Although ATM is expected to efficiently support "bursty" traffic sources such as interactive data and motion video, the dynamic nature of bursty traffic may cause severe network congestion when a number of "bursty" sources add cells. The goal of congestion control scheme is to enable the network to utilize its resources efficiently, in the meanwhile satisfy the Quality of Service (QOS) requirements such as cell loss ratio and cell transfer delay, of the users [1]. It is also commonly required that an congestion control strategy can provide "fair" access to the network resources for all users. In general, the logical queueing model of a 16x16 shared buffer ATM switch can be depicted in Fig. 1. This paper deals with overload control during congestion in a shared buffer ATM switch via selective cell discard and buffer management. Specifically, we consider the question of efficiency in buffer management in order to reduce the number of cells that have to be discarded during congestion.

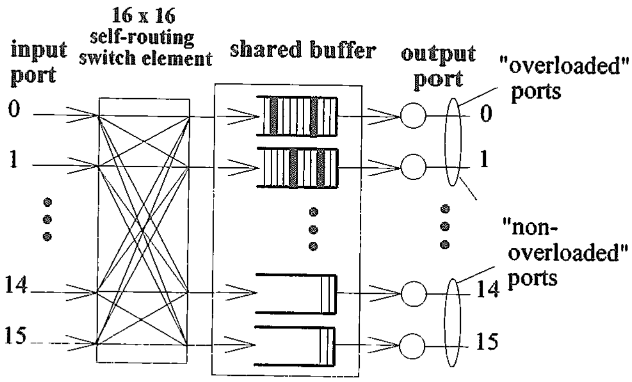


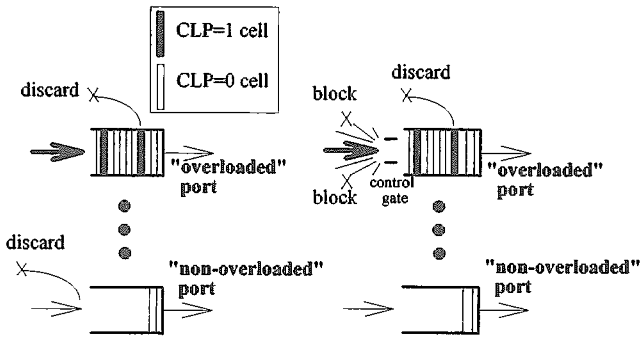
Fig. 1 A queueing model of the 16x16 shared buffer switch

ATM cells have an explicit cell loss priority bit (CLP bit) in the header. The SCD (selective cell discard) scheme adopted in our study is similar to the "pushout" mechanism proposed in [2], which assumes cells of both {CLP=0} and {CLP=1} share a common buffer. If the buffer is full and a {CLP=0} cell arrives, a cell with {CLP=1} (if available) will be pushed out and lost. In this paper, we modify the pushout mechanism according to the concept of "fair" access as required by the congestion control strategy. To provide "fair" access to the network resources for all users, the overload of a particular output port should not affect the performance of other output ports. Therefore, some overload control is necessary in the shared buffer memory switch. Several control schemes have been proposed in [3] to prevent performance degradation of the shared buffer memory switch under imbalance traffic conditions. In our study, we adopt the sharing with a maximum queue length scheme (SMXQ) [3], [4]. The SMXQ scheme assumes a queue length threshold is chosen for the logical queue pertaining to each output port, and if the queue length exceeds the threshold, the arriving cells are discarded. Of course, the sum of these maxima must be greater than the total buffer size in order to take advantage of buffer sharing. Thus the switch performs cells discarding when the buffer is full or the length of a particular output port exceeds its threshold. Clearly, the discard policy is completely specified by the set of chosen queue length thresholds. In this paper, we use SMXQ buffer control scheme with a set of dynamically adjusted queue length thresholds.

Although there are many cell discarding policies and buffer management mechanisms proposed for ATM overload control in the literature, e.g. [5]-[9]. They all considered a single outgoing link and the corresponding dedicated buffer in a network node. Petr and Frost in [6] considered a single-server queue with a number of cell discarding priorities. The priority discarding policy is controlled by a set of nested queue fill thresholds. Various space priority buffer control policies for a single outgoing link in an ATM switching node was proposed and studied in [8], [9]. A space priority policy consists of a queueing scheme and a selective cell discarding scheme. When the buffer is not full, the queueing scheme determines how a new cell is placed in the buffer. The selective cell discarding scheme is used when it is necessary to discard a cell from the buffer. Endo et al. in [4] proposed an ATM shared buffer memory switch architecture. In [4] buffer sharing was evaluated by a traffic analysis and experimental measurements. These evaluations showed that the required buffer capacity for a shared buffer memory switch is reduced than that required for a separated buffer memory switch. SMXQ scheme was also adopted in [4] with the threshold of the output queue length is a predefined constant for each output port during a simulation run. In our study we use SMXQ buffer control scheme with a set of dynamically adjusted queue length thresholds. The threshold for each output queue is adjusted dynamically according to the overload condition of its associated output port and available memory size in the shared buffer throughout the simulation run.

2. Cell Discarding versus Buffer Management

Consider a general queueing model of a shared buffer ATM switch with bursty arrivals depicted in Fig. 1. The queueing model consists of R (R=16 in our model) parallel single server queues, where R is the number of the input (also output) ports. A RxR self-routing switch element routes each incoming cell to its appropriate output port by inspecting the internal switch routing header of the cell. We assume a cell upon arrival at the input port i joins the queue of the output port j with transition probability P_{ij}, i, j = 1, 2, ..., R such that $\sum_j P_{ij} = 1$, i = 1, 2, ..., R. Cells destined for the same output port join the same output



(a) when buffer is full (b) when buffer is congested or getting full

Fig. 2 Traditional versus "gated" cell discarding policies

queue and are served in the FIFO order (if without being discarded). Time is slotted and the transmission time of a cell takes one slot. In general, the queue length of an overloaded port is always greater than that of a non-overloaded port, and this situation is commonly more apparent when buffer is getting full. The probability of finding a {CLP=1} cell in the output queue belonging to an overloaded port is thus higher than that in the output queue belonging to a non-overloaded port. Here we assume most of cells resided in the shared buffer are {CLP=0} cells, i.e., the tagging probabilities of cell generators are small, say 0.001 to 0.01. If we apply the pushout mechanism to the shared buffer in a per port basis, that is, when the buffer is full and a {CLP=0} cell arrives, a {CLP=1} cell, if available in the same output queue as incoming cell destined for, will be pushed out and lost. Obviously, the discarded cells of the queue belonging to an overloaded port are most likely to be {CLP=1} cells. On the contrary, the discarded cells of the queue belonging to a non-overloaded port are most likely to be {CLP=0} cells. This result seems very "unfair" in the sense that overloaded ports should get some penalty, but the {CLP=0} cells destined for the non-overloaded ports are more likely to be dropped than that destined for the overloaded ports when buffer is full. In this paper, we try to find out an efficient buffer control scheme to get rid of this "unfair" phenomenon, in the meantime reduce the total number of discarded cells during buffer congestion. Fig. 2 illustrates the above observations. Fig. 2(a) shows traditional pushout mechanism adopted when buffer is full. For an overloaded port, most cells that have to be discarded are belonging to {CLP=1}cells. On the contrary, most cells that have to be discarded for a non-overloaded port are belonging to {CLP=0} cells. In Fig. 2(b), we add a control gate in front of the output queue belonging to an overloaded port. The control gate can block some incoming cells from entering into the corresponding output queue when buffer is congested or getting full. Thus, some available buffer space in the shared buffer can be saved for accommodating the incoming cells destined for the non-overloaded ports. How to find an efficient "gated" control policy is our goal in this paper. In the following Section, we model this "gated" cell discarding policy as a variation of SMXQ buffer control scheme described in the previous Section.

3. SMXQ Buffer Control Scheme

Clearly, the efficiency of the "gated" cell discarding policy is completely determined by the set of chosen gate widths for the output queues belonging to those overloaded ports. The width for the control gate of an output queue should be adjusted dynamically during congestion according to the overloaded condition of the associated output port and the available buffer size in the shared buffer. If the overload condition of an output port becomes heavy during buffer congestion, the width of its control gate should be narroweddown. Also, if the available buffer size in the shared buffer becomes less, the widths of the control gates belonging to the overloaded ports should be narrowed down. On the contrary, if the available buffer size in the shared buffer becomes more, the widths of the control gates should be widened. In other words, the degree to which the buffer is full directly influences the widths of the control gates of the output queues belonging to the overloaded ports. Of course, if the shared buffer is not congested (i.e., there has enough available buffer space in the shared buffer), the "gated" cell discarding policy will not be necessary. Because in such situation the incoming cells destined for the non-overloaded ports can always find space in the shared buffer, and it won't be necessary to block any cell destined for the overloaded ports. In our study, we assume buffer is congested only if the available buffer size is less than 32, where the total buffer size is 1024. That is, the "gated" cell discarding policy is applied to the shared buffer only if the available buffer size is less than 32. This is according to our simulation experience that cell discarding result of the "gated" control policy becomes no improvement or even worse if the available buffer size is greater than 32. The value of this congested threshold must be chosen carefully, too long or too short are both infeasible for the "gated' cell discarding policy.

We classify the degree to which the buffer is congested into four stages. The available buffer size for stage 1 is ranging from 24 to 31, for stage 2 is ranging from 16 to 23, for stage 3 is ranging from 8 to 15, and for stage 4 is ranging from 0 to 7. During each stage, we assume the width of the control gate for an overloaded output queue is a constant (a non negative integer). Thus, for each overloaded output queue, the "gated" control policy can be characterized by a 4-tuple control vector (X1, X2, X3, X4), where Xi, i= 1, 2, 3, and 4, denotes the gate width for stage i during buffer congestion. Obviously, X1 $\geq$ X2 $\geq$ X3 $\geq$ X4, and the values of Xi's can be varied for different queues according to their various overloaded conditions.

3.1 Dynamically Adjusted Queue Length Thresholds

In this subsection, we model the "gated" cell discarding policy as a variation of the SMXQ buffer control scheme. Consider an overloaded output port with a nonempty logical queue in the shared buffer. Assuming that, at the beginning of slot i, the available buffer size is belonging to stage k, and the queue length of the particular output queue is denoted by QL(i). Thus, we have

$$MQL(i) = QL(i) + Xk \qquad (1)$$

where MQL(i) denotes the maximum queue length of the particular output queue during slot i. Xk denotes the gate width of the particular output queue during slot i, and it also represents the maximum number of incoming cells that can be put into the particular output queue during slot i. If we let A(i) represent the number of incoming cells destined for the particular output queue during slot i, we obtain

$$QL(i+1) = QL(i) + \min\{A(i), Xk\} - 1 \qquad (2)$$

where QL(i+1) denotes the queue length of the particular output queue at the beginning of slot i+1. Of course, the available buffer size at the beginning of slot i+1 may not be the same as that of slot i. If we assume the congested degree is changed from stage k to stage k' at the beginning of slot i+1. Then we have

$$MQL(i+1) = QL(i+1) + Xk' \qquad (3)$$

where MQL(i+1) denotes the maximum queue length of the particular output queue during slot i+1.

From Eqs. (1) and (3), we observe that the maximum queue length of an overloaded output queue in our SMXQ scheme can be varied from slot to slot. Depending on the queue length of the particular output queue and the available buffer size in the shared buffer at the beginning of each slot.

3.2 Modified Pushout Mechanism

In this subsection we propose a modified pushout mechanism to effectively perform cell discarding when the buffer is full or the length of a particular output queue exceeds its threshold. Our modified pushout mechanism assumes a newly arrived {CLP=0} cell destined for a particular output port cannot be blocked or discarded from admission to the buffer if there exists at least one {CLP=1} cell in the corresponding output queue of the port. The control rules of this modified mechanism are classified according to the buffer occupancy conditions as follows.

When buffer is full: All the newly arrived {CLP=1} cells are discarded. For a newly arrived {CLP=0} cell destined for a particular output port, the following cell expelling procedure is performed. If there are {CLP=1} cells in the corresponding queue, then the {CLP=1} cell closest to the head of the queue is pushed out. Otherwise, the newly arrived {CLP=0} cell is blocked.

When buffer is congested but not full: If the queue length of a particular output queue does not exceed its threshold, then admit the newly arrived cell destined for it. Otherwise, the newly arrived {CLP=1} cell is discarded, and the cell expelling procedure is performed for the newly arrived {CLP=0} cell.
When buffer is not congested: Admit all the newly arrived cells without blocking.

4. Numerical Results

Figs. 3-8 display some of the preliminary numerical results we have obtained. The objective is to study the performance including the total number of cells discarded during congestion and the "fairness" of our proposed "gated" cell discarding policy. To closely simulate bursty traffic, 16 identical On-Off sources were considered generating arrivals for 16 input ports. A 16x16 self-routing switch element with transition probability matrix [Pij], where Pij is defined in Section 2, routes each incoming cell to its appropriate output port. Cells destined for the same output port join the same output queue in the shared buffer. With asymmetry [Pij], a merging of the arrival streams to each output port causes imbalanced traffic load for the output queues in the shared buffer. In this section, we investigate the performance characteristics of SMXQ scheme with dynamically adjusted queue length thresholds under three different imbalance traffic conditions. In the first case, we assume Pon-off = 0.145 (transition probability from on to off state) and Poff-on = 0.815 for all 16 On-Off sources. The transition probability Pij takes only on two values 0.073 and 0.052. For all i, Pij = 0.073 for j= 0, 1, ..., 7, and Pij = 0.052 for j= 8, 9, ..., 15. Thus, in this case the MOL (mean offered load) for OPi (output port i) is 0.9916 for i= 0, ... , 7, and 0.70636 for i= 8, ... , 15. For simplicity, we use MOL[0.9916*8, 0.70636*8] to express this case. Obviously, OP0 - OP7 can be viewed as overloaded ports, and OP8 - OP15 can be viewed as non-overloaded ports. Similar to the first case, we assume Pon-off = 0.252, Poff-on = 0.743; Pij = 0.083 for j= 0, ... , 7, and 0.042 for j= 8, ... , 15, in the second case. Thus, in the second case the mean offered load for individual upper eight output ports is 0.9916, and 0.5017 for individual lower eight output ports, i.e., MOL[0.9916*8, 0.5017*8]. In the third case, we assume Pon-off = 0.255, Poff-on = 0.367; Pij = 0.105 for j= 0, ... , 7, and 0.02 for j= 8, ... , 15. Thus, the mean offered load for individual upper eight output ports is 0.9912, and 0.1888 for individual lower eight output ports, i.e., MOL[0.9912*8, 0.1888*8]. For each case, we further assume the total buffer size is 1024, the simulation time per run is 5 million time slots, the tagging probability for each On-Off source is 0.05, and initially buffer is empty.

Fig. 3 shows the total number of discarded cells versus different control vectors for the first case. During each simulation run, an only control vector is applied to all the overloaded ports. For each control vector the total number of discarded cells belonging to the following five groups are considered. These groups are: all 16 ports, all 8 overloaded ports, all 8 non-overloaded ports, the {CLP=0} cells, and the {CLP=1} cells. Within a control vector, 'x' denotes infinite gate width, i.e., without gate control, and '0' denotes zero gate width, i.e., gate is closed. Take for example, a control vector 'xx10' means X1= x, X2= x, X3= 1, and X4= 0. Clearly, control vector 'xxxx' means no gate control is applied during congestion, that is only modified pushout mechanism is adopted. In the following discussion the performance improvement for all control vectors are referenced to that of control vector 'xxxx'. Notice that with an effective control vector, for example 'xx11' and xx10, about 40% of the overall cell loss can be avoided. This improvement is mainly due to the reduction of the total number of discarded cells belonging to the {CLP=0} cells. We also observe that this reduction is occurred at the non-overloaded ports. No cells are discarded at the non-overloaded ports during congestion for most of the control vectors we considered. Thus we prove our concept that an efficient control vector can not only reduce the total number of discarded cells, but also satisfy the "fair" access requirement under network congestion conditions.

In Fig. 4 we show the cell loss percentage versus different control vectors for the first case. For each control vector, the cell loss percentage of a particular class of cells, for instance, Overloaded class as listed in the legend, is obtained by dividing the total number of discarded cells belonging to that class by the total number of discarded cells in a simulation run. The total number of discarded cells belonging to the 'Overloaded' class means that belonging to all the overloaded output ports. Similar explanation can be applied to the other classes in the legend. To cite one example, the cell loss percentages of the 'Overloaded' and 'Non-overloaded' classes under control vector 'xxxx' are 0.6 and 0.4, respectively. Again, we notice that the cell loss percentage of the 'Non-overloaded' class becomes zero under most of the control vectors. For control vector 'xxxx', it is interested that '{CLP=1}, Overloaded' > '{CLP=0}, Overloaded', but on the contrary '{CLP=0}, Non-overloaded' > '{CLP=1}, Non-overloaded'. This trend is agreed with our previous discussion that the probability of finding a {CLP=1} cell in an overloaded queue is higher than that in a non-overloaded output queue.

Figs. 5 and 6 show the total buffer occupancy and the total number of discarded cells, respectively, for each slot in a specific time interval during a simulation run, for control vectors 'xxxx' and 'xx00' individually. From Fig. 5

we observe that the excess cells are discarded when buffer is full under control vector 'xxxx'. This causes some flatted mountain tops at height 1024 (the total buffer size). On the contrary, there have no such mountain tops exist at the same time interval under control vector 'xx00'. The control vector 'xx00' implies that the gate width of the control gate for each overloaded output queue is 0 when the total buffer occupancy over 1008. Due to this early blocking policy the total buffer occupancy is lowered down. The detailed cell discarding processes occurred in the specific time interval for both control vectors 'xxxx' and 'xx00' are presented in Figs. 6(a)-6(b), and 6(c), respectively. Figs. 6(a) and (b) show the total number of cells, including {CLP=0} and {CLP=1}, discarded at the overloaded and non-overloaded ports, respectively, under control vector 'xxxx'. Fig. 6(c) shows the total number of cells discarded at the overloaded ports under control vector 'xx00', in the same time interval. Notice that no cells will be discarded at the non-overloaded ports under the control vector 'xx00' during whole simulation run. The improvement in cell discarding between Figs. 6(a)-(b) and Fig. 6(c) can be easily seen. By blocking some incoming cells destined for the overloaded ports before the buffer is full or getting full, the performance of Fig. 6(c) is better than that of Figs. 6(a)-(b), in both the total number of discarded cells and the "fair" access requirement. We also observe that the number of {CLP=1} cells discarded during congestion at the overloaded ports is more than that of {CLP=0} cells, as this can be seen from Figs. 6(a) and 6(c). This observation is reversed in Fig. 6(b) for the cells discarded at the non-overloaded ports.

Figs. 7 and 8 show the total number of discarded cells versus different control vectors for the second case (MOL[0.9916*8, 0.5017*8]) and the third case (MOL[0.9912*8, 0.1888*8]), respectively. Compare Figs. 7 and 8 to Fig. 3, we obtain the following observations. First, the performance improvement becomes less significant as the input traffic to the non-overloaded ports becomes less. This is because the benefit obtained by blocking some incoming cells destined for the overloaded ports so making rooms for just a few incoming cells destined for the non-overloaded ports becomes less significant in Figs. 7 and 8. Next, the total number of discarded {CLP=0} cells is greater than that of {CLP=1} cells under control vector 'xxxx' in Fig. 3, but this situation is reversed in Figs. 7 and 8. Since the input traffic to the non-overloaded ports in Fig. 3 is greater than that in Figs. 7 and 8, and most of cells discarded at the non-overloaded ports during congestion are {CLP=0} cells. Third, the total number of discarded cells occurred at the non-overloaded ports is reduced to zero under most of the control vectors for all three figures, thus reduce the total number of discarded {CLP=0} cells.

5. Conclusions and Future Work

This paper considers overload control during congestion in a shared buffer ATM switch. In Section 2 we propose a "gated" cell discarding policy. This "gated" cell discarding policy can be modeled as a variation of SMXQ (sharing with a maximum queue length) scheme with a set of dynamically adjusted queue length thresholds. Numerical results show that by appropriately selecting the values of dynamically adjusted queue length thresholds, the proposed SMXQ scheme can not only satisfy the "fair" access requirement under network congestion conditions, but also reduce the total number of discarded cells thus increases overall performance.

More thorough investigations for the "gated" cell discarding policy during congestion under multifarious imbalance traffic conditions are jobs remain to be completed. How to determine an optimal set of control vectors for the output queues belonging to the overloaded ports under a specific traffic condition, so that the total cell loss is minimum (or nearly minimum) needs to be further studied. The tradeoff between total cell loss and mean waiting times for different output ports, and that for different cell loss priorities are not considered in the paper. They are also good subjects for future research.

References

[1] A. E. Eckberg, "B-ISDN/ATM Traffic and Congestion Control," IEEE Network, pp. 28-37, Sep. 1992.

[2] H. Kroner, "Comparative Performance Study of Space Priority Mechanisms," IEEE INFOCOM'91, pp. 1136-1143.

[3] F. Kamoun and L. Kleinrock, "Analysis of Shared Finite Storage in a Computer Node Environment under General Traffic Conditions," IEEE Trans. Commun., vol. COM-28, pp. 992-1003, July 1980.

[4] N. Endo et al., "Shared Buffer Memory Switch for an ATM Exchange," IEEE Trans. Commun., vol. COM-41, pp. 237-245, Jan. 1993.

[5] D. W. Petr and V. S. Frost, "Optimal Packet Discarding: An ATM-Oriented Analysis Model and Initial Results," IEEE INFOCOM'90, pp. 537-542, 1990.

[6] ----------, "Nested Threshold Cell Discarding for ATM Overload Control: Optimization Cell Loss Contraints," IEEE INFOCOM'91, pp. 1403-1412, 1991.

[7] V. Yau and K. Pawlikowski, "Improved Nested-Threshold-Cell-Discard Buffer Management Mechanisms," IEEE TENCON'92, pp. 820-824, 1992.

[8] L. Tassiulas et al., "Optimal Buffer Control During Congestion in an ATM Network Node," IEEE/ACM Trans. on Networking, Vol. 2, No. 4, pp. 374-386, Aug. 1994.

[9] T. C. Hou and Y. C. Tung, "UPC and Selective Cell Discarding in ATM Networks," ISCOM'95, Taipei, Taiwan, pp.69-76, 1995.

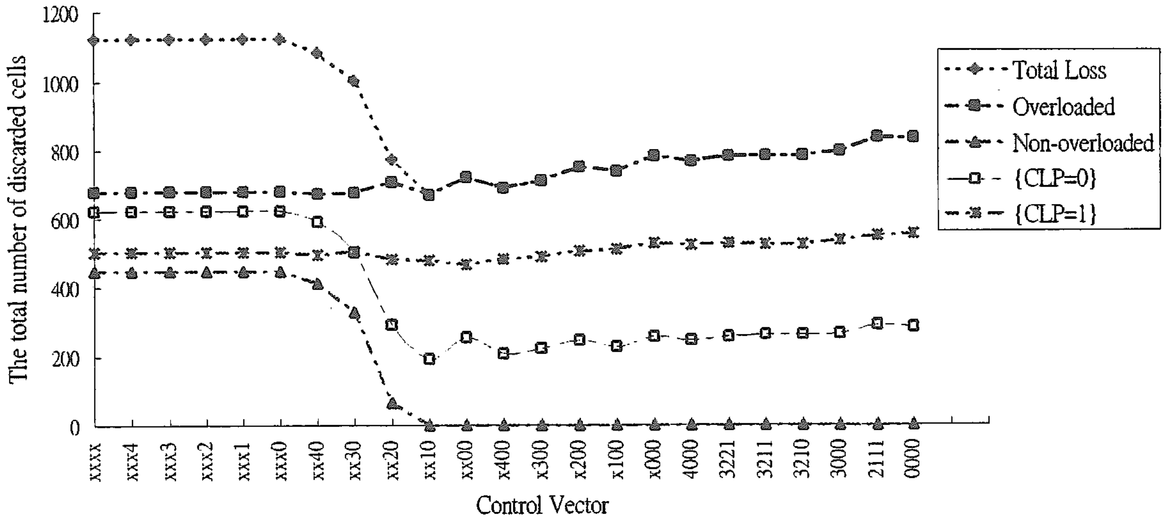


Fig.3 The total number of discarded cells versus different control vectors for MOL[0.9916*8, 0.70636*8]

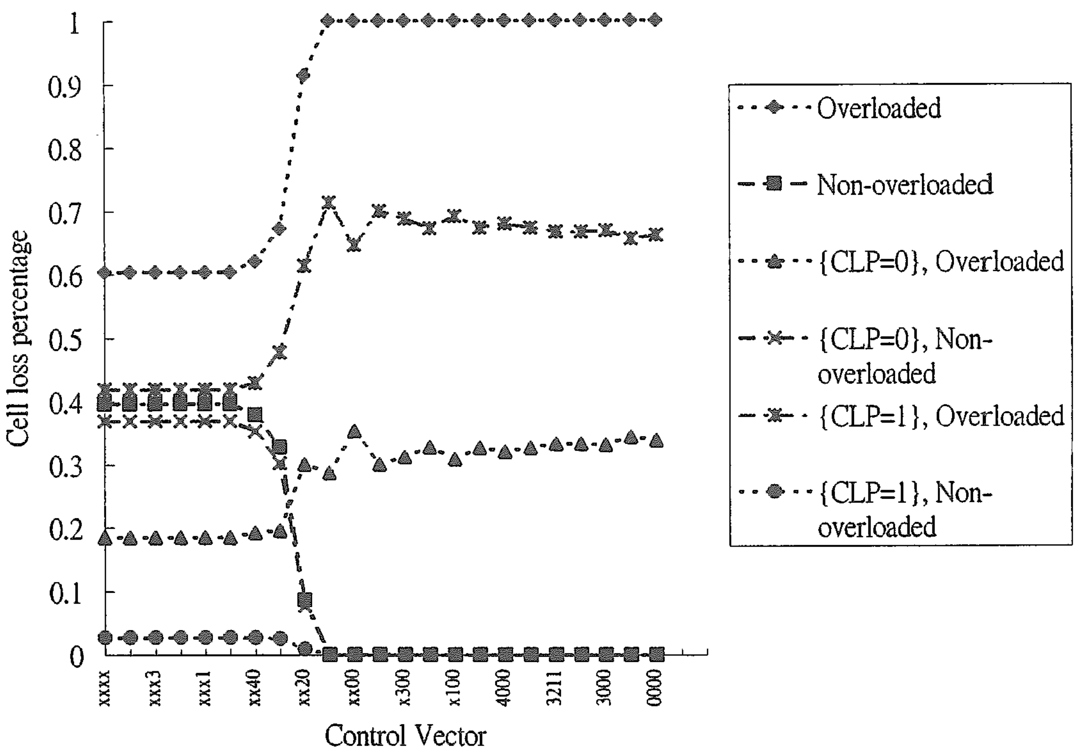


Fig.4 The cell loss percentage versus different control vectors for MOL[0.9916*8, 0.70636*8]

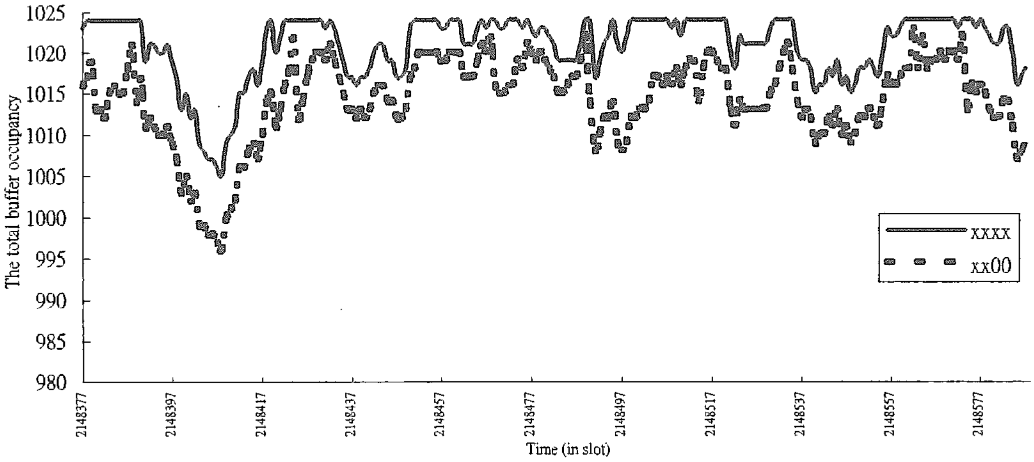


Fig. 5 The variation of the total buffer occupancy during a specific time interval for control vectors xxxx and xx00.

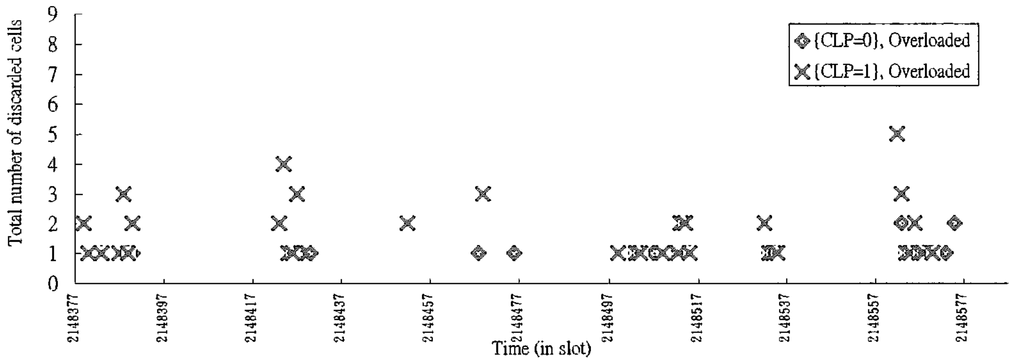


Fig. 6(a) Control vector: xxxx.

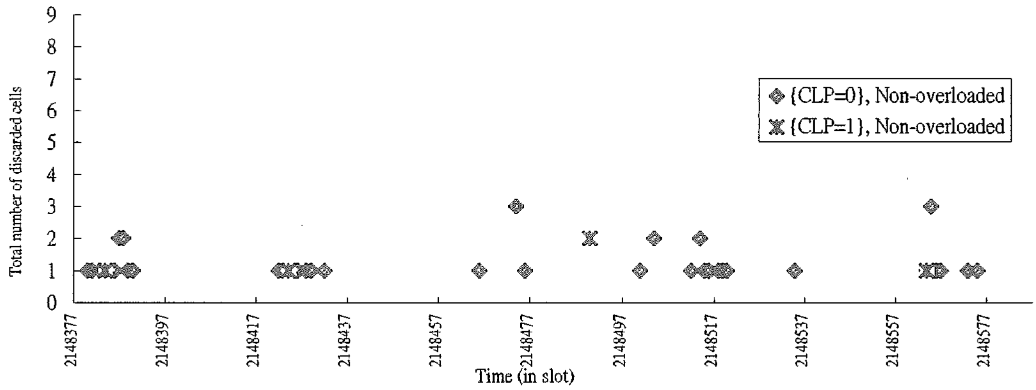


Fig.6(b) Control vector: xxxx.

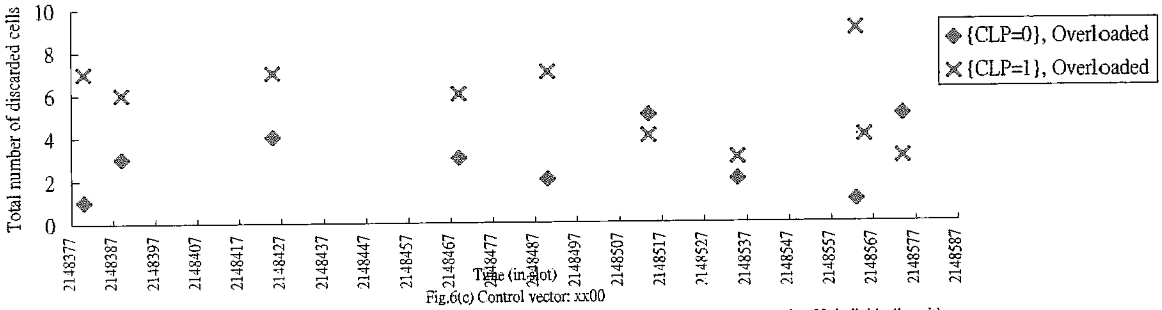


Fig.6(c) Control vector: xx00

Fig.6 The variation of the total number of discarded cells during a specific time interval for control vectors xxxx and xx00, individually, with MOL[0.9916*8, 0.7063*8].

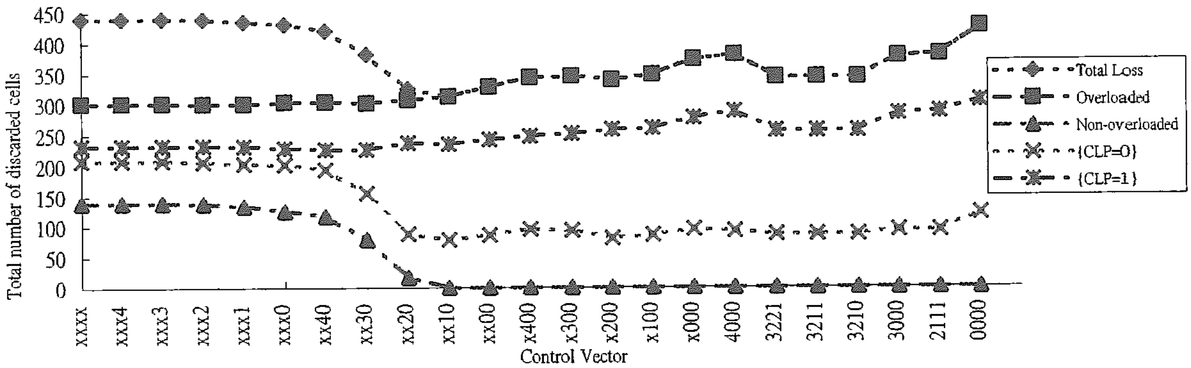


Fig. 7 The total number of discarded cells versus different control vectors for MOL[0.9916*8, 0.5017*8].

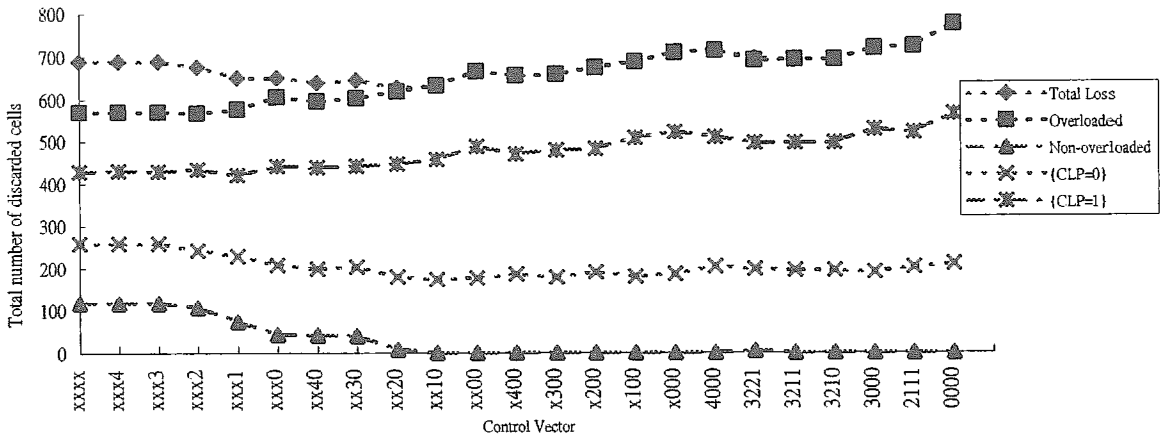


Fig. 8 The total number of discarded cells versus different control vectors for MOL[0.9912*8, 0.1888*8].