

一個適用於多媒體資料擷取的及時性磁碟排程法之研究

李建億

國立台南師範學院 資訊教育研究所

leeci@ipx.ntntc.edu.tw

廖淑婷

國立台南師範學院 資訊教育研究所

摘要

本文乃針對多媒體儲存系統之磁碟排程問題，提出一更有效率的及時性磁碟排程方法，在這套方法中，我們設法減少磁碟頭不必要的移動，以求得最佳存取時間；其主要構想是以各要求的存取期限與磁碟位置之相互關係為考量，在磁碟頭移動路途中將期限稍晚的要求預先讀取，減少磁碟頭不必要的回移，經由數學證明與模擬分析的結果，我們的方法確實能比現有其他的磁碟排程法減少一些不必要的延遲，更能增加多媒體儲存系統的服務效率。再則，我們將進一步考量多重磁碟儲存環境下的情形來進行實驗模擬分析，可以發現我們所提出的方法，比傳統的磁碟排程法更能適用於多重磁碟的環境並提供更佳的產能。

1. 緒論

電腦技術的進步已使得多媒體 (multimedi) 如聲音 (audio)、動畫 (animation)、視訊 (video) 等應用的情形日漸普遍。某些型式的多媒體，尤其是視訊通常都需要很大的頻寬與儲存空間，而為了儲存這麼龐大的多媒體資料並考慮其讀寫傳輸的速度，目前硬碟仍是最為被廣泛採用的儲存媒體。多媒體資料的儲存形式通常是一些連續性資料的組合 (如 audio samples or video frames)，由於有限緩衝區和頻寬的限制，無法一次讀入全部的資料，因此必須將一連串的资料分成數次處理，即形成一連串的工作要求 (requests) 稱之為資料流 (stream)，亦即所謂的連續性媒體 (Continuous Media, CM)。

以播放一小時的 HDT (High Definition Television) 而

言，大約有 36Gbits 的資料內容和 100 Mbps 的頻寬需求，然而一個硬碟卻只能提供約 80 Gbits 的儲存空間與 20 Mbps 的頻寬，足見單一磁碟已無法應付這些多媒體資料存取的需求。然而，以多重磁碟 (multi-disk drive) 來儲存多媒體資料時，為平衡每個磁碟的負載，勢必得將原本連續的多媒體資料分割出來分別存放在各個磁碟。在這種多重磁碟的狀態下，怎樣才能達成服務的品質，諸如：及時地完成工作要求、避免播放的過程中產生延遲等，另一方面又可允許更多資料流同時存取的需求，是多媒體儲存系統面臨的首要問題。

就多媒體儲存系統而言，除了要有足夠的儲存空間外，而且必須避免在播放的過程中因為延遲所產生的跳動 (jitter)，因此及時地完成使用者所提出的工作要求是非常重要的。在磁碟資料傳輸速度和儲存空間的限制下，每個服務的週期只能取出有限的資料量以供每個連續性資料流使用，而當系統服務的資料流愈多就愈容易超過磁碟負載的上限，而使得有些資料流無法在需求的時間內被讀取出，整個連續性資料流的傳輸過程中就愈可能產生延遲。因此如何設計好的儲存與搜尋策略來滿足連續性媒體有限緩衝區與時間的限制，正是多媒體儲存系統所要面臨的重要課題。

傳統的磁碟排程方法，如 First-Come-First-Served (FCFS) [6]、Shortest-Seek-Time-First (SSTF) [6]、SCAN [6]、CSCAN [12] 等主要目的在減少磁碟頭移動時間，並能維持公平性且達到較高的產能 (Throughput)，其中 SCAN 法經證實為最佳的磁碟排程演算法 [4,5]，能有最少的磁碟移動時間，然而並不適用於有時間限制的及時系統中，舉例而言，當磁碟頭由一端移動至磁碟另一邊以服務工作要求時，位於磁碟頭另一端的迫切工作要求，

可能因為等待磁碟頭回移至此的時間過長而超過其時間限制。

目前的及時性磁碟排程方法，如 Earliest Deadline First (EDF)[9]乃是以 deadline 早晚來將一群工作要求加以排程，而未考慮各個工作要求的相對磁區位置，因此可能產生很多額外的磁碟移動時間。近來已有人針對連續性媒體提出適切的磁碟排程方法，如 SCAN-EDF[10,11]除以較早 deadline 的工作要求為優先外，相同 deadline 的工作要求則以 SCAN 的法則加以排程來改善原 EDF 多餘的磁碟移動時間，但事實上各工作要求的 deadline 相等的情況並不多，因此改進的狀況實屬有限；即使如 GSS 法[3]先將相似的 deadline 分群再 SCAN，其實驗所得結果亦發現有額外的磁碟移動時間。

另外，其他的連續性媒體磁碟排程方法如 Preseek[7]、BSCAN[8]等，主要的作法是在一固定時間內服務所有的資料流各一次的工作要求；Preseek 法致力於使每個服務週期有最小的磁碟移動時間，而 BSCAN 法所採取的策略則是在每個服務週期之前，即先行計算出各資料流到下一個服務週期所要播放的資料量，預先讀入整批的資料，也就是說，每個資料流於每次要求時所讀入的資料量是變動的，它所讀入的資料量必須足夠大到可使此資料流在下一週期讀入下一筆資料前的播放過程中不致發生延遲。這兩種方法，一旦有新的工作要求都必須等待下一個週期才會被服務到，儘管這新的工作要求有較迫切的需求也不允許先服務，因此就整體而言產能並不佳。一個好的磁碟排程策略必須使各個資料流的工作要求都能在限期內完成，並且能允許服務更多的資料流，因此若我們不受限於較早 deadline 優先，而能使磁碟頭移動時間減小，必能增加可服務的資料流數量。所以我們的構想乃是允許預先存取 deadline 較晚的工作要求，以減少一些事後磁碟頭回移的時間，因此使得各工作要求的服務時間也都相對減少，而能夠完成更多資料流工作要求的存取服務。

本文的其餘各節內容如下，在第二章中，我們將提出一個新的磁碟排程方法，稱之為 Prefetched SCAN-EDF 法（簡稱 PS 法），並利用數學分析來證明其有效與正確性；並描述如何實驗模擬 Prefetched SCAN-EDF 法的運作以及初步的實驗結果。第三章則將進一步改良 PS 法

以提昇儲存系統的產能。最後一章則為本文的結論與探討未來的研究方向。

2. Prefetched SCAN-EDF 法（PSE 法）

2.1 Prefetched SCAN-EDF 之演算法則

目前的及時性排程方法如 EDF 法乃是為了符合及時性系統限期完成的要求而設計，但可能會使磁碟頭產生多餘的移動，增加服務的 seek time，因而延長工作要求的等待時間；而 SCAN-EDF 法則加入了減少 seek time 的考量，主要是在 EDF 方法中加入當工作要求的 deadline 相同時利用 SCAN 法來減少磁碟頭不必要的移動，因而比 EDF 法有較少的 seek time；然而，磁碟頭若能在移往下一個工作要求讀取磁區的路程中，先行處理 deadline 較晚的工作要求而且不致影響原工作要求磁區的讀取（亦即不會超過其 deadline），便可省去其後磁碟頭再回移至該較晚 deadline 工作要求的磁區，我們將此提出的方法稱之為 Prefetched SCAN-EDF 法（簡稱 PS 法），其演算法如圖 1 所示。

因為在播放連續性媒體的資料流時，一連串的资料存取工作要求必須滿足整個播放過程的頻寬要求，也就是說前一工作要求讀取後存於緩衝區的資料在被播放完畢之前，下一工作要求的資料必須被讀入緩衝區。因此，對於每一連續性資料流的各個工作要求之 deadline 是由上一工作要求所讀取的資料量與該資料流要求的播放頻寬所決定。當新的資料流服務需求提出時，我們即可依據該資料流的各工作要求先行計算出每個要求的 deadline，再將這些工作要求依其 deadline 加以排序。為說明起見我們以單一磁碟的例子來簡介 Prefetched SCAN-EDF 法的運作情形，如圖 2 所示，假設有 3 個資料流 $S1:\{r11, r12\}$, $S2:\{r21, r22\}$, $S3:\{r31, r32\}$ 分別包含兩個工作要求，且各工作要求所要讀取的磁區分別為 791($r11$), 619 ($r12$), 217 ($r21$), 963 ($r22$), 889 ($r31$), 653($r32$), 而對應之 deadline 分別為 132($r11$), 212 ($r12$), 68($r21$), 149 ($r22$), 60 ($r31$), 140($r32$)。如圖 2(a)所示，假設磁碟頭平均移動一個磁區的時間為 0.055 ms，讀取一個磁區的時間為 1 ms；因磁碟頭目前位在磁區 354，且 $r31$ 具有最早之 deadline，而具有次早 deadline 的 $r21$ 並不在

由磁碟頭移往 r_{31} 的路上，所以 Prefetched SCAN-EDF 法不作任何預先存取的動作，當存取完 r_{31} 的工作要求時在移往 r_{21} 的路上存在著順序為 3 的 r_{11} ，此時 PSE 法會先計算若預先讀取 r_{11} 是否會影響到 r_{21} 的讀取：以目前的時間 30 ms 再加上磁碟頭由 889(r_{31})至 791(r_{11})的移動時間($98 * 0.055$)和讀取時間 1ms，並計算由 791(r_{11})至 217(r_{21})的磁碟移動時間與資料讀取時間，故 $30 + (98 * 0.055 + 1) + (574 * 0.055 + 1) = 6$ 大於 r_{21} 的 deadline 68，表示不允許預先讀取 r_{11} ，因此 PS 法便直接由 r_{31} 移至 r_{21} ；接下來由 r_{21} 移往 r_{11} 的路程上可預先讀取順序為 4 的 r_{32} ，因為 $68 + (436 * 0.055 + 1)$ ($138 * 0.055 + 1$) = 102 < 132，符合 PS 法預先存取原則故可提前讀取 r_{32} ，因此磁碟頭可省去下次回移至 r_{32} 的時間。依此採用 PS 法服務此 6 個工作要求的順序為 r_{31} 、 r_{21} 、 r_{32} 、 r_{11} 、 r_{22} 、 r_{12} ，磁碟頭總移動距離為 2297 個磁區，而磁碟移動時間則為 132 ms，平均等待時間為 89.5 ms，如圖 2(b)所示。

若採用 SCAN-EDF 演算法，依照 6 個工作要求的 deadline 則服務順序應為 r_{31} 、 r_{21} 、 r_{11} 、 r_{32} 、 r_{22} 、 r_{12} ，磁碟頭移動總距離增加為 2573 個磁區，而磁碟移動時間增加為 ($535 * 0.055 + 1$) + ($672 * 0.055 + 1$) + ($574 * 0.055 + 1$) + ($138 * 0.055 + 1$) + ($310 * 0.055 + 1$) + ($344 * 0.055 + 1$) = 144 ms，平均等待時間則增加為 ($30 + 68 + 101 + 110 + 128 + 148$) / 6 = 97.5 ms，如圖 2(c)所示。

```

Algorithm 1: Prefetched SCAN-EDF
SR: array; the set of requests; /* SR is the buffer to record the coming requests */
begin
  while (SR != empty) do
    begin
      sort SR according to the deadline of requests in ascending order,
      j := 1;
      for i := 2 to SR_end do
        begin
          if (location[i] is between location(disk_head) and location[1])
            and (prefetch i will not disobey the deadline of the request SR[i] to SR[i-1])
              /* location[i] is the storage location of SR[i] */
            then j := i;
          else
            break for loop;
        end if;
      end for; /* next request */
      if (j=1)
        /* if j=1 denotes it can't prefetch any request else it can prefetch j requests */
      then
        follow the SCAN direction from location(disk_head) to location[1] to
        prefetch the request SR[j] to SR[1] and delete these requests from SR.set;
      end if;
      serve the request SR[1] and delete it from SR.set;
    end while;
  end

```

圖 1、Prefetched SCAN-EDF(PSE)的演算法

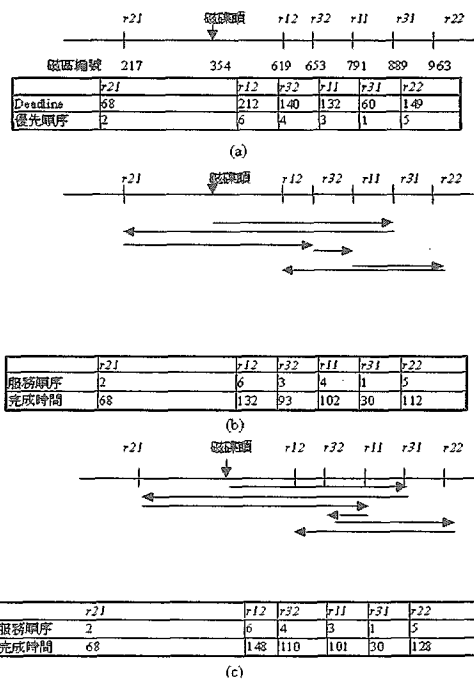
2.2 Prefetched SCAN-EDF 法有效性的分析

為了證明 PS 法在任何情形下均能比 SCAN-EDF 法有

較少的磁碟頭移動時間，在此節中，我們將利用數學分析來證明此方法之有效性。

假設以 b_0 表示磁碟頭起始位置， b_k 表示 request k 要求的磁區($k = 1, 2, \dots, n$)，而各 request 的 deadline 為 $d_1 < d_2 < \dots < d_n$ ，我們將利用歸納法來證明。所以，首先考慮前 2 個工作要求的情形，依照 SCAN-EDF 法不管各個工作要求位於哪個磁區，磁碟頭的移動順序必為 b_1 、 b_2 則磁碟頭之移動距離 S_2 為 $|b_0 - b_1| + |b_2 - b_1|$ 。而 PSE 法則會考慮 b_0 與前 2 個工作要求 b_1 、 b_2 的相對位置，所以(1)當 b_2 介於 b_1 與 b_0 之間且不影響 b_1 之讀取下依 PS 法便可能預先讀取 b_2 ，則磁碟頭之移動距離 P_2 為 $|b_0 - b_1|$ 比 S_2 減少了 $|b_2 - b_1|$ ，故 $P_2 = S_2 - |b_2 - b_1| < S_2$ ；(2)若否，則 PS 法會與採 SCAN-EDF 法的服務順序相同，都為 b_1 、 b_2 ，磁碟頭之移動距離 P_2 為 $|b_0 - b_1| + |b_2 - b_1| = S_2$ 。由上述(1)(2)情況可知，在只有兩個工作要求的情形下，PS 法磁碟頭總移動距離恆小於等於 SCAN-EDF 法，亦即 $P_2 \leq S_2$ 。再則當第 3 個工作要求加入考慮時，依 SCAN-EDF 法磁碟頭的移動順序必

圖 2、3 個資料流 (6 個工作要求) 同時要求磁碟存取服



務的例子(a)各個工作要求讀取資料所在磁區 位置及其 deadline (b)Prefetched SCANEDF 法之運作 (c) SCAN-EDF 方法之運作

為 b_1 、 b_2 、 b_3 ，因此磁碟頭總移動距離必須加上 $|b_2 - b_3|$ ，所以 $S_3 = S_2 + |b_2 - b_3|$ ；而依照 PS 法則須考慮下

列各種情形：

(1) 若服務完前 2 個工作要求後的磁碟頭移動距離為 $P_2 = S_2 - |b_2 - b_1|$ (即預先讀取 b_2):

(a) 當 $b_1 > b_3 > b_2 > b_0$ 或 $b_1 > b_2 > b_3 > b_0$ 或 $b_0 > b_2 > b_3 > b_1$ 或 $b_0 > b_3 > b_2 > b_1$, 亦即 b_3 介於 b_0 與 b_1 之間, 且不影響 b_1 與 b_2 的讀取, 則 $P_3 = P_2 < S_3$;

(b) 當 $b_3 > b_1 > b_2 > b_0$ 或 $b_3 < b_1 < b_2 < b_0$ 則 $P_3 = P_2 + |b_1 - b_3| = S_3 - 2|b_1 - b_3| < S_3$;

(c) 當 $b_1 > b_2 > b_0 > b_3$ 或 $b_3 > b_0 > b_2 > b_1$ 則 $P_3 = P_2 + |b_1 - b_3| = S_2 - |b_2 - b_1| + |b_1 - b_3| = S_2 + |b_2 - b_3| = S_3$;

(2) 倘若服務完前 2 個工作要求後的磁碟頭移動距離為 $P_2 = S_2$:

(a) 當 $b_3 > b_2 > b_1 > b_0$ 或 $b_3 < b_2 < b_1 < b_0$ 或 $b_3 < b_2 < b_0 < b_1$ 或 $b_1 < b_0 < b_2 < b_3$

則 $P_3 = P_2 + |b_2 - b_3| = S_3$;

(b) 當 $b_2 > b_3 > b_1 > b_0$ 或 $b_0 > b_1 > b_3 > b_2$ 或 $b_1 > b_0 > b_3 > b_2$ 或 $b_2 > b_3 > b_0 > b_1$ 而且若預先讀取 b_3 不會影響到 b_2 的讀取, 則 $P_3 = P_2 < S_3$ 。

因此由以上的分析可得證在考慮前 3 個工作要求的情形下, PS 法磁碟頭總移動距離亦恆小於等於 SCAN-EDF 法, 即 $P_3 \leq S_3$ 。

接著, 假設 $b_i (t \leq i)$ 代表磁碟頭服務完 $i-1$ 個工作要求時最後停靠的位置, 則我們可將 PS 法的磁碟總移動距離 P_i 及 SCAN-EDF 法磁碟頭的總移動距離 S_i 以一般化表示為:

$$P_i = \begin{cases} P_{i-1} & \text{if } b_i \in (b_{i-1}, b_0) \text{ and } F_k \leq D_k, k=1,2,\dots,i; \text{ where } F_k \text{ is} \\ & \text{the fulfill time of request } k, D_k \text{ is the deadline of request } k \\ P_{i-1} + |b_i - b_{i-1}| & \text{otherwise} \end{cases}$$

$$S_i = S_{i-1} + |b_i - b_{i-1}|$$

最後, 假設當 $k=i$ 時 $P_i \leq S_i$ 成立, 我們將證明當 $k=i+1$ 時, $P_{i+1} \leq S_{i+1}$ 成立: (1) 若 b_{i+1} 介於 b_0 與 b_i 之間, 且不影響 b_1, b_2, \dots, b_i 的讀取, 亦即 $F_1 \leq D_1, F_2 \leq D_2, \dots, F_i \leq D_i$, 則 $P_{i+1} = P_i \leq S_i < S_{i+1}$; (2) 若否, 則 $S_{i+1} - P_{i+1} = S_i + |b_{i+1} - b_i| - (P_i + |b_{i+1} - b_i|) = (S_i - P_i) + |b_i - b_{i+1}| \geq 0$, 所以 $P_{i+1} \leq S_{i+1}$ 。由以上的歸納證明得知以 PS 法來存取資料會比類似 SCAN-EDF 這種

以較早 deadline 優先為主要的方法有較小的磁碟頭移動距離, 相對地也就能減少磁碟頭總移動時間。

2.3 實驗模組

為證明 PS 法實際運作的有效性, 我們將實驗模擬 PSE 法的運作, 並與其他的方法作一建構於連續性媒體環境的效能比較, 其中包括目前雖未考慮多媒體及時性但磁碟移動時間為最佳的排程方法 SCAN、考慮 deadline 的 EDF 法以及在 EDF 中加入考慮磁碟移動時間的 SCAN-EDF 法。

整個實驗模擬架構所採用的磁碟功能規格參數如表 1 所列, 並假設磁碟頭移動所需的時間與磁區位置的遠近呈線性關係。在模擬系統中各資料流提出要求服務的時間是隨機產生, 並且每個連續性資料流會持續在固定時間間隔週期性地提出位於不同磁區的工作要求, 直到整個資料流播放完畢。

表 1、磁碟功能規格 (FUJITSU MAB3091)

| | |
|----------------------|-------------|
| Drive capacity | 9.1 GB |
| Disks | 5 |
| Max seek | 16 ms |
| Track to track | 0.8 ms |
| Average | 7.5 ms |
| Transfer rate | 12 Mbytes/s |
| Latency time | 4.17 ms |
| Tracks per cylinders | 5 |
| Cylinders | 8490 |
| Sectors/track | 167 to 248 |

對某一資料流而言一旦緩衝區內的資料消耗殆盡卻沒有新的資料供應, 將會使得播放中斷, 因此必須在緩衝區內尚有足夠的資料量以前, 完成該資料流下一讀取的工作要求。是故為了避免播放過程有延遲的情形, 所以每個資料流下一筆資料讀取完成的最晚期限是依據目前緩衝區內仍可供該資料流播放的資料量來決定的。

實驗模擬系統分別以工作要求數與資料流數量為輸入參數, 以求得關於每個資料流的平均等待時間、平均完成時間、產能 (亦即可完成的要求數與資料流數), 以及所需緩衝區大小等的實驗結果。為減少誤差, 每次實驗結果均執行 100 次模擬之後再求其平均。

2.4 實驗結果

實驗模擬是以 SCAN 法、EDF 法、SCAN-EDF 法與 PSE 法來分別處理 2、3、4、5、10、15、20 個資料流, 並假設每個資料流所提出的連續性工作要求數是相等的 (亦

即播放的時間長度是相同的)。而實驗比較結果如完成資料流、完成要求數、平均完成時間、平均等待時間及緩衝區需求量，將分別呈現於圖 3、4、5、6 與 7。

由圖 3 可看出，隨著提出需求的資料流之增加，磁碟負載也漸漸增大，因此各方法所能服務成功的資料流也隨之降低，其中 PS 法能完成比傳統磁碟排程法較多的資料流，而不考慮 deadlin 的 SCAN 法所能完成的資料流數最少。圖 4 呈現的是所能完成的要求數比率，可視為處理非連續之及時性的工作要求（亦即每個工作要求各自獨立），在未考慮連續性媒體的情況下，PSE 法亦能完成最多的要求數。

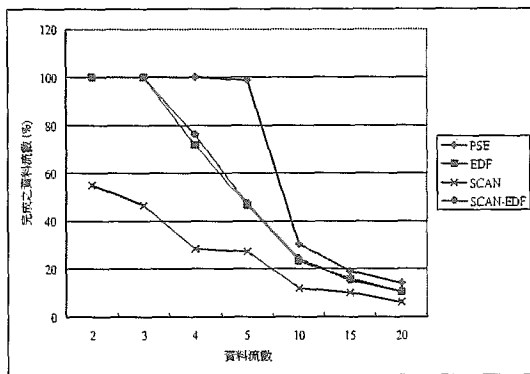


圖 3、完成之資料流數比率

圖 5 與圖 6 所呈現的結果是各方法的平均完成時間與資料流的平均等待時間，其中完成時間是指磁碟服務完成一個成功資料流的時間，而等待時間則是指資料流第一個工作要求被服務到的時間。由圖 5 可看出，其中 PSE 法可以與具有最佳之磁碟頭移動時間的 SCAN 法有差不多的完成時間，足以證明 PS 法確實能在不違背及時性的情況下減少一些不必要的磁碟頭移動時間。而在圖 6 所呈現的結果中，隨著提出需求之資料流數的增加，各資料流的等待時間也相對增加。SCAN 法在磁碟頭移動距離最佳化的情況下，似乎能具有最少的等待時間，然而卻無法保證資料流可以順利被讀取播放完成；PSE 法的平均等待時間僅次於 SCAN 法，而且是在服務較多資料流的情況下。

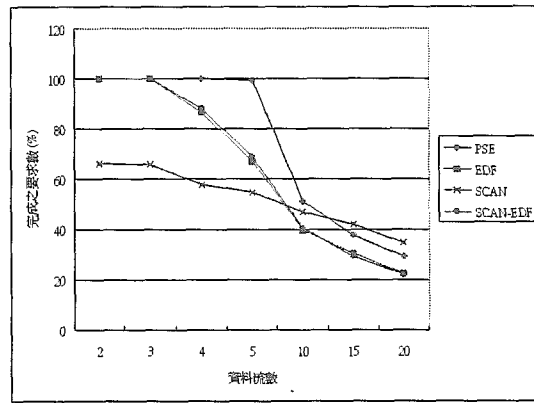


圖 4、完成要求數比率

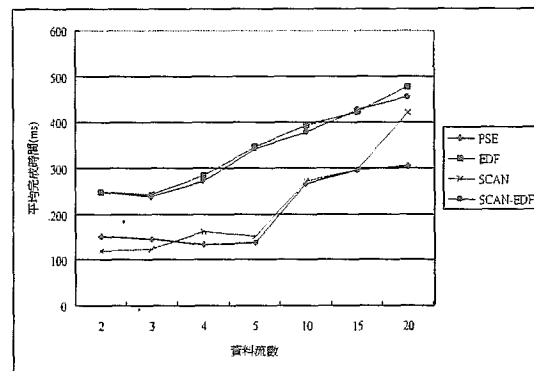


圖 5、平均完成時間

而由圖 7 可看出，隨著提出要求數的增加，不考慮及時性的 SCAN 法所需的緩衝區也會隨之拉大，可知 SCAN 法不僅可完成的資料流數最少，而且需要很大的緩衝區，足以證實未考慮 deadlin 的 SCAN 法的確並不適用於連續性媒體。而 EDF 法與 SCAN-EDF 法因為所服務的都是較迫切的工作要求，所以其緩衝區需求量較少。圖中並顯示當磁碟未達負載時，PS 法的緩衝區需求量會隨著服務的要求數增加而拉大，而當磁碟承載超過負荷時，資料消耗的速度會大於資料讀取速度，緩衝區需求量也會因此而減少。另一方面，若比較緩衝區需求量與完成資料流數，由實驗的結果來看，PS 法雖然因為預先讀取勢必得使用較大的緩衝區，然而其資料流完成率卻比 EDF 法與 SCAN-EDF 法提高了 2 倍。

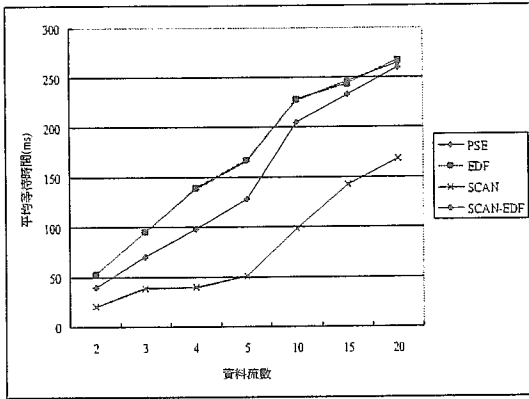


圖 6、資料流平均等待時間

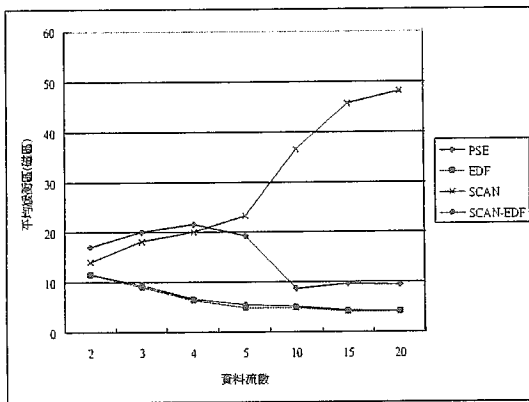


圖 7、平均所需緩衝區大小

deadline = 149)位於磁碟頭移往順序為 1 的 request (r_{11} , deadline = 132)之路程外,因此限制了順序為 4 的 request (r_{12} , deadline = 212)預先讀取的機會,只能預先讀取優先順序為 2 的 r_{32} (亦即 deadline = 140 僅次於 priority 為 1 的 request, r_{11}), 如此只減少了由 r_{11} 到 r_{32} 的距離;若能在不影響其他工作要求讀取的情況下,讓 request r_{12} (讀取順序為 4)也能預先讀取,其後便又可省去由 r_{22} 回移至 r_{12} 的距離,使磁碟頭移動時間更為縮短,如圖 9(b)。

```

Algorithm 2 : Extend-PSE
SR array: the set of requests; /* SR is the buffer to record the coming requests */
begin
while (SR != empty) do
begin
sort SR according to the deadline of requests in ascending order.
j := 1;
for i := 2 to SR_end do
begin
if (location[i] is between location[disk_head] and location[1])
and (prefetch i will not disobey the deadline of the request SR[1] to SR[i - 1])
/* location[1] is the storage location of SR[1] */
then
j := i;
end if;
end for; /* next request */
if (j=1)
/* if j=1 denotes it can't prefetch any request else it can prefetch jth request */
then
prefetch the request SR[j] and delete it from SR set;
end if;
serve the request SR[1] and delete it from SR set;
end while;

```

圖 8、Extend-PS 的演算法

3. Extend-PSE 法

3.1 Extend-PS 法之演算法則

依照 PS 法的演算法則,有機會預先讀取的工作要求必須是即將被服務到的要求,也就是說該要求的 deadline 必須僅次於目前本應處理的要求,若符合這個條件的工作要求位於磁碟頭移動路程之外,則其他任何工作要求皆無預先讀取的機會。然若能在不影響其他讀取順序在其之前的各個工作要求之情況下(亦即這些工作要求被務的時間皆不會超過各自的 deadline),提供更多的彈性,讓位於磁碟頭移動路程上 deadline 更晚的工作要求也能預先讀取,那麼必能減少更多的磁碟頭不必要的移動時間,我們將此法稱為 Extend-PS 法,其演算法如圖 8 所示。

我們將延續圖 2 的例子來說明 Extend-PS 法的運作情形,如圖 9(a)所示,由於服務順序為 3 的 request (r_{22} ,

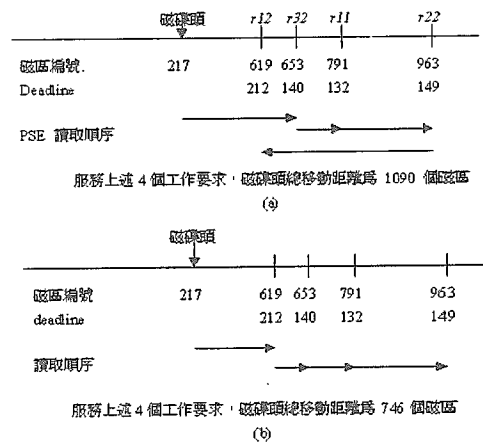


圖 9、(a) Prefetched SCAN-EDF 之作法(b) Extend-PSE 法之作法

若預先讀取任何位於磁碟頭移動路程上的工作要求,對於原來應該被讀取的所有工作要求之 deadline 不會產生影響,也就是說預先讀取所花費的時間,並不會使得原來服務順序較該工作要求為早的任一要求超過

deadline，則 Extend-PS 法即可在符合及時性需求的情況下，減少更多磁碟移動時間，完成更多資料流的存取。

3.2 實驗結果

由前實驗結果我們已知 PS 法的效能較傳統磁碟排程法為佳，然而，改進自 PS 法的 Extend-PS 法是否真能獲得最佳的效能，尚有賴於實驗模擬的驗證；因此，本節將利用前節所採的實驗模組及其相關假設來模擬 Extend-PS 法的運作，並且將與之前實驗結果最佳的 PS 法來作比較，其結果包括完成資料流數、完成要求數以及緩衝區需求量，都將分別呈現於圖 10、11 與 12。由圖 10 與 11 我們得知 Extend-PS 法確實能更進一步提升系統效能，完成更多的資料流；而圖 12 則呈現出，Extend-PS 法的緩衝區需求量也因為預先讀取之工作要求數的增加，而需要較大的緩衝區。

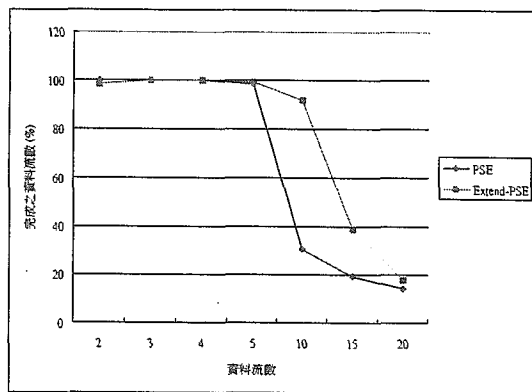


圖 10、完成之資料流數比率

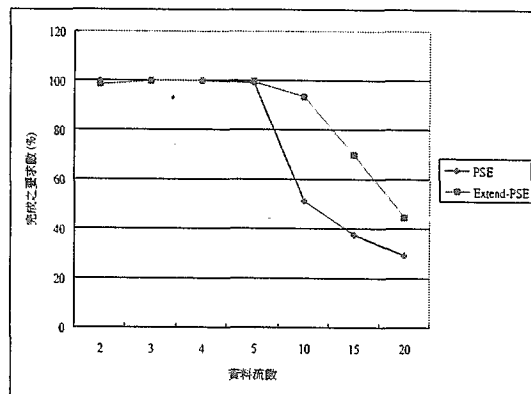


圖 11、完成要求數比率

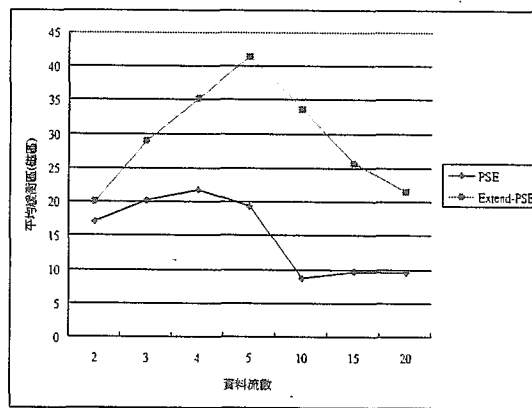


圖 12、平均所需緩衝區大小

我們可看出在這樣的模擬環境之下，當資料流數小於 5 時磁碟並未超載，無論是 PS 法或 Extend-PS 法皆能完成全部的資料流，Extend-PS 法卻因為預先讀取的工作要求數增加，而使得緩衝區需求量也呈現大幅度的增長。因此，我們認為當提出要求的資料流數並不會超過磁碟負載時，採用 PS 法就足以應付所有的資料流，如此即可減少緩衝區的需求量；而若該磁碟使用率較為頻繁，則建議採用 Extend-PS 法來增加系統效能。

4. 結論與未來研究方向

由一連串有序的工作要求所組成的連續性媒體，在讀取播放的過程中，除了不允許中斷以外，其播放必須依照資料流原來的順序，而且須有一定的播放速度，如此播放出來的資料才能視為有意義。而在多媒體儲存系統中，除了必須滿足上述連續性媒體的基本需求外，也必須提供最佳的效能與最好的服務品質，讓系統中所儲存的資料能夠作最充分的利用。然而，傳統的磁碟排程法均未能對連續性媒體特性作多方的考量，以致於無法充分地發揮多媒體儲存系統的效能；因此，於本文中，我們提出了一套適用於連續性媒體的磁碟排程法，而且也從相關的驗證與實驗分析中，證明了這套方法優於其他磁碟排程法。

在第二章中所呈現的 PS 法是以 deadline 為優先，但另外加入考量各個工作要求的相關位置，作法是在磁碟頭移往較早 deadline 工作要求的路程上，預先讀取下一個即將會被讀取的工作要求；我們並透過數學分析來證明預先讀取的有效性，且以實驗模擬的方式來與其他傳統

磁碟排程法作比較與分析。而在第三章中我們則針對 PS 法作更進一步的改良（稱之為 Extend-PS 法），增加更多預先讀取的可能性。從實驗模擬的結果可以驗證，PS 法與 Extend-PS 法，均可比傳統磁碟排程法能服務完成更多的工作要求，其中尤以 Extend-PSE 法更能增加約為 SCAN-EDF 兩倍的效能。

參考文獻：

- [1] Berson, S., Ghandeharizadeh, S., Muntz, R. & Ju, X., "Staggered Striping in Multimedia Information Systems", *ACM SIGMOD*, pp. 79-90, 1994.
- [2] Chang, R. I., *Real-time disk scheduling in multimedia system*. Unpublished doctoral dissertation, Chaio Tung University, Taiwan, 1996.
- [3] Chen, M. S., Kandlur, D. D. & Yu, P. S "Optimization of the group sweeping scheduling (GSS) with heterogeneous multimedia streams," *Proc. ACM Int. Conf. Multimedia, Anaheim*, CA, pp. 235-242, 1993.
- [4] Chen, T. S. & Yang, W. P., "Amortized analysis of disk scheduling algorithm V(R)," *Journal of Information Science and Engineering*, vol. 8, pp. 223-242, 1992.
- [5] Chen, T. S., Yang, W. P. & Lee, R. C. T., "Amortized analysis of some disk scheduling algorithms: SSTF, SCAN, and N-step SCAN," *BIT*, vol. 32, pp. 546-558, 1992.
- [6] Denning, P. J., "Effects of scheduling on file memory operations," *Proc. AFIPS*, Montvale, N.J., pp. 9-21, 1967.
- [7] Gemmel, D. J., Han, J., Beaton, R. J. & Christodoulakis, S., "Delay-Sensitive multimedia on disks," *IEEE MultiMedia*, pp. 56-67, 1994.
- [8] Kenchamma-Hosekote, D. R. & Srivastava, J., "I/O scheduling for digital continuous media*," *Multimedia Systems*, vol. 5, pp. 213-237, 1997.
- [9] Liu, C. L. & Layland, J. W., "Scheduling algorithms for multiprogramming in a hard real-time environment," *Journal of ACM*, vol. 20, no. 1, pp. 46-61, 1973.
- [10] Reddy, A. L. N. & Wyllie, J., "Disk scheduling in a multimedia I/O system," *Proc. ACM Int. Conf. Multimedia*, Anaheim, CA, pp. 225-233, 1993.
- [11] Reddy, A. L. N. & Wyllie, J., "I/O issues in a multimedia system," *IEEE Computers*, vol. 27, no. 3, pp. 69-74, 1994.
- [12] Teory, T. J., "Properties of disk scheduling policies in multiprogrammed computer systems," *Proc. AFIPS Fall Joint Computer Conf.*, pp. 1-11, 1972.