計畫名稱: **支援**QoS **路由器的封包排程與服務分流之設計與實作**

Design and Implementation of Packet Scheduler and Service Classifier for QoS Router

計畫編號：NSC88-2219-E-002-008

主持人：孫雅麗　台灣大學資訊管學系　副教授

E-mail: sunny@im.ntu.edu.tw　　Fax:02-3621327

中文摘要：(關鍵詞：傳輸服務品質保證、排程方法、多媒體服務)

要在網路上提供好的多媒體應用的服務，我們需要一個能保證傳輸品質的網路架構;而一個網路的傳輸品質好壞取決於網路對end-to-end delay **、頻寬以及** delay jitter **等傳輸效能的保證。**

欲支援網路傳輸品質控制服務，其中最重要的是每個網路節點，如子網路或IP**路由器所使用的排程方法，必須能保障封包在傳輸路徑上傳輸的品質。在本計畫，我們設計了一個** frame-based **的** weighted-fair queueing algorithm**。跟以往** WFQ **和** WF^2Q **中，利用** virtual finishing time **來作為排程依據的方式不同，我們是以**queueing jitter **作為排程的依據。我們希望能將同一**flow **的** packets **盡量平均的分散在時間軸上，如此才能讓** worst-case fair index **和網路封包間的** delay jitter **最小。這對在網路上傳輸即時的多媒體的資料十分的重要。我們也證明所提出的方法在達到這個目的的同時，也能兼顧** bounded-delay **和** fairness **的要求。**

為了讓不同的分類及排程方法可以得到檢驗與比較，我們建置了一個易於實作且可提供正確效能檢測的路由器雛形，在這個雛形中，我們主要實作可提供priority queueing**及**WFQ**兩種排程方法的封包排程器模組，此外一個以"資料流"及"網際網路協定標頭欄位的服務類別"做為分類依據的簡易分類器也包含在雛形中，最後我們並做了一系列的實驗用以驗證路由器雛形的效能。**

英文摘要：(Keyword: QoS, Scheduling, Frame-based Queueing, Delay Jitter, packet scheduler, packet classifier)

As the Internet continues to grow in terms of the amount of traffic and the number of users, there are currently a lot of interests in providing multimedia communication services. Many of these new applications rely on the ability of the networks to provide QoS guarantees typically in the form of bounded end-to-end delay, bandwidth and delay jitter.

In this paper, we propose a new scheduling algorithm called Maximum Jitter First (MJF) algorithm. It is a frame-based weighted-fair queueing algorithm. Different from WFQ and WF^2Q, the algorithm uses delay jitter to describe the timestamps of the packets and to assign service order between flows within a frame. Slots assigned to a flow are spaced out within the frame so to minimize the Worse-case Fair Index and to achieve minimal delay jitter for packets of a flow. It as well possesses the bounded-delay and fairness properties. Furthermore, in MJF, the service order is calculated only upon call arrival and departure. It therefore has O(1) computational cost.

In order to allow various classifier and scheduling policies to be examined and evaluated, a QoS router is prototyped. Currently, two scheduling algorithms: priority queueing and WFQ and a simple classifier based on "per-flow queueing" and the "Type of Service" field, have been implemented. We also conducted a number of experiments to evaluate the

performance of the prototype in QoS provisioning.

Part I: 封包排程

1. 研究動機與目的

在網路的流量與使用者日益增多的今天，許多人開始對如何在網路上提供如網路電話、視訊會議及 WebTV…等的多媒體服務產生興趣。這些新形態的網路服務，都必須要在能確保證網路服務品質(QoS)的狀況下，才能正常的運作。而所謂的網路服務品質，包含了 bounded end-to-end delay, bandwidth 及 delay jitter 等。

在設計 QoS network 時，最重要的關鍵就是在 switches 或 routers 中使用合適的排程方法。在過去，已經有許多人提出了各種不同的的排程方法。根據其結構，我們可以將這些方法分成 timestamp-based 及 frame-based 兩類。在timestamp-based 的方法中，大家最關心的就是其系統所需的處理時間，當有N個 backlogged 的 flows 正等待被處理的狀況下，此種方法所需的處理時間為O(logN)。在高速網路的架構下，當有大量的flows時，處理時間將變成一個很重要的問題。

Frame-based 的排程方法，主要是將時間軸分成一段一段的frame來處理。在每個frame內，資料封包(packets)是跟據一個特定的順序來傳送。這種排程方法最主要的優點就是它的處理時間只需要O(1)的時間就能完成，且相較於 timestamp-based 的方法來說，在實作上也比較容易。但卻有較差的 fairness 及 delay bound 的問題。

而除了 fairness 和 delay bound 的問題外, worst-case fair index 和 delay jitter 也是個非常重要的考量。尤其對於多媒體資料的傳送來說，delay jitter往往成為影響服務品質的重要因數。因此我們希望能有一個方法，其處理的時間只有O(1)，而且其 worst-case fair index 和 delay jitter 越低越好，同時又兼顧到 fairness 和 delay bound 的要求。

2. 相關研究

由於我們的目的是要設計一個處理時間只有O(1)的方法，因此如WFQ、SCFQ[1]及WF^2Q[2]…等, 用 timestamp-based 的方法並不詳加介紹。我們主要的研究方向是採用 frame-based 方式的排程。在過去已經有不少人提出了相關的排程方式，以下我們將介紹幾種較常見到的方法。

(1) HRR (Hierarchical Round Robin)

HRR 的主要特點是其 frame table 是階層式的。越高的階層分配到的頻寬越多。舉例來說, 若第一層的 frame table 有10個 slots ，而其中一個 slot 再分給同樣擁有 10 個 slots 的第二層 frame table使用, 則第一層的 slots 所分配的到頻寬就會是第二層 slots 的 10 倍。

(2) WRR(Weighted Round Robin) [3]

在傳統的 Round Robin 的方法中，每個 flow 在每個frame中，只會輪到一次。而 WRR 的方式則是根據每個 flow 的 weight 來調整其在 frame 中出現的次數。

(3) DRR(Deficit Round Robin) [4]

在之前提到的 Round Robin 方法中，都只能在packet size 都一樣的狀況下才能運作。若packet size 大小不同，將會導致 fairness 上的問題。DRR 則解決了這個問題。在這個方法中用到了配額(Credit)的觀念，在輪到屬於某個

flow 的 slot 時，是給予此flow一個定量(Quantum)的配額，再去判斷是否送出packet，而不是直接送出packet。如此一來就可處理大小不同的 packet ，且又能達到 fairness 的要求。

(4) QLWFQ (Queue Length Based Weighted Fair Queueing) [5]

相較於之前提到的各種 Round Robin 方法，QLWFQ 則是沒有固定的 frame table。此方法的 frame table 是根據 packet 到達的先後順序再加上每個 flow 的 weight 來排。Frame table 不是固定死的，如此一來對於 flow 的處理就較有彈性。

3. Frame-based WFQ 之架構設計

在我們的方法中，我們有一個 frame table 記錄著所有 flow 的處理順序，而與舊有的 Round Robin 方法不同的是，我們並不是一開始就固定整個系統的處理順序，而是在有新的 flow 加入，或是舊的 flow 離開時都會根據每個 flow 的要求來排定整個 frame table 的內容。且在傳送 packets 時，我們採用 DRR 的做法，給予此 flow 一個定量(Quantum)的配額，再去判斷是否送出 packet 及要送出多少個 packets，因此也可以在 packet size 不同的網路架構上運作。

而由於 frame table 的大小是固定的，在排完所有的 slots 後，有可能會有沒辦法剛好滿足每個 flow 的要求的狀況，因此我們會在每個 frame 結束時更新每個flow的配額，如此一來就可以達到 fairness 上的要求。

4. Frame table 中 Slot 的排程方式之研究與設計

根據之前提到的架構，我們共提出了三個不同的排程方法。第一種是利用 EDF (Earliest Deadline First)[6]的方式來作排程，我們根據每個 flow 的 weight 來設定其increment ，進而算出 virtual start time 和 virtual finish time ，再根據這些數值來排定先後順序。第二種方式是利用 RM (Rate monotonic) [6]的方式來作排程，這種方式是依據 increment 的大小再配合 virtual start time 來作排程，increment 越小的越先處理，因此 weight 較高的 flow 將會優先被考慮。最後，我們提出了一個叫做Maximum Jitter First (MJF) 的 frame- based weighted -fair 的排程方法。與舊有用 virtual finish time來作為決定先後順序的方法不同，在這個方法中我們是採用 delay jitter 來作為判斷先後的依據。由於是採用 frame-based 的方法，因此我們方法的主要核心就是如何去安排 frame table 中的每個 slot。我們主要的目的是希望能盡量降低 Worst-case Fair Index 和 delay jitter 同時也能兼顧到 delay bound 和 fairness 的問題。

由於這些排程的動作都是在 flow 建立或離開時才做，因此在packet 傳送時，系統的處理時間仍然是O(1)。

5. 結論

本論文提出了一個 frame-based 的 queueing algorithm。根據這個方法，處理每個packet所需的時間是固定的，不會因為 flow 的個數而增加。而在 frame table 大小固定為 M 時，排定 frame table 的前置處理所需花的時間也可限制在 $O(M^2)$ 之內。我們可以處理各種不同 flow weight 的組合，且在 packet 大小不同的網路架構下也能正常運作。

在我們的架構中，我們將 delay

jitter 作為我們主要的考量，但我們盡量降低 delay jitter 的同時，也能保證 end-to-end delay 和 fairness。此外，根據這種架構，我們提出了三種不同的排程方法，這些方法都可適用於即時的多媒體資料傳輸。

參考文獻

[1] S. Golestani, “A self-clocked fair queueing scheme for broadband applications,” in Proc. IEEE INFOCOM ’94, pp. 636-646, April 1994. Wroclawski, J., “The Use of RSVP with IETF Integrated Services”, RFC 2210, September 1997.

[2] J. C. R. Bennett and H. Zhang, “WF²Q: Worst-case fair weighted fair queueing,” in Proc. IEEE INFOCOM ’96, pp. 120-128.

[3] M. Katevenis, S. Sidiropoulos, and C. Courcoubetis, “Weighted round-robin cell multiplexing in a general-purpose ATM switch chip,” IEEE Journal on Selected Areas in Communications, vol. 9, pp. 1265-79, October 1991

[4] M. SHreedhar and G. Varghese, “Efficient fair queueing using deficit round robin,” int Proc. SIGCOMM’95, pp. 231-242, Sep. 1995.

[5] Yoshihiro Ohba, “QLWFQ: A queue length based weighted fair queueing algorithm in ATM Networks,” in Proc. IEEE INFOCOM ’97, pp 566 -575 vol.2

[6] C. L. Liu and J. W. Layland, "Scheduling algorithm for multiprogramming in a hard real0time environment," Journal of ACM, 20(1): 46-61,January 1973

[7] J. Lehoczky, L. Sha, and Y. Ding, "The rate monotonic scheduling algorithm: : Exact characterization and average case behavior," In Proc. of the Real-Time Systems ymposium.

[8] D. Ferrari and D. Verma, “A scheme for real-time channel establishment in wide-area networks,” IEEE Journal on Selected Areas in Communications, vol. 8, pp. 368-379, April 1990.

[9] J. L. Lexford, A. G. Greenberg, and F. G. Bonomi, “Hardware-efficient fair queueing architectures for high-speed networks,” in Proc. IEEE INFOCOM ’96, pp. 638-646.

[10] S. Golestani, “A framing strategy for congestion management,” IEEE Journal on Selected Areas in Communications, vol. 9, pp. 1064-1077, September 1991.

Part II: QoS 路由器實作

1. 簡介

為了提供傳輸的服務品質(quality of service, QoS)，擔任封包遞送的路由器(router)必須提供封包傳輸控制的機制，為資料流(flow)保留足夠的資源，並記錄資料流使用資源的狀況。而要達到上述目標，必須在路由器加入三個重要元件：一是傳輸允諾控制(admission control)，二是封包分類器(packet classifier)，三是封包排程器(packet scheduler)。

我們的研究目的在於建立一個提供傳輸服務控制的路由器的雛型(prototype)，主要依照Priority Queueing及Weighted Fair Queueing(WFQ) 這兩種排程理論，實作封包排程器這個元件，並在此路由器雛型上進行一系列實驗以驗證排程理論的效能。

2. 封包排程器

2.1 Priority Queueing

Priority Queuein-based packet scheduling的做法是當封包要自網路卡送出去之前必須先經過封包分類器(packet classifier)的過濾，讓封包安置到適當的貯列(queue)之中，而封包排程器(packet scheduler)在選擇下一個被傳送的封包時，選擇的順序由優先權(priority)最高的貯列開始找起，如果具有高優先權的貯列沒有封包待傳的話，就再探訪優先權次高的貯列，當找到一個貯列之中有封包待傳時，就從該貯列挑出最早進入貯列的個封包，將該封包自網路卡傳送出去。

我們可以發現這種排程方式可以讓擁有較高優先傳送權的封包先讓路由器遞送出去，減少了封包傳輸的延遲並降低封包因緩衝區(buffer)滿溢(overflow)被丟棄(drop)的機會，因而可提供不同程

度的服務品質，不過這種排程方法所展現出的服務品質差異只能看出“質化”(quality)的不同，無法做到“量化”(quantity)的區別。

2.2 WFQ

WFQ (Weighted Fair Queueing) [3]又稱為PGPS (Packet-version Generalized Processor Sharing)，是一種近似於FFQ (Fluid Fair Queueing, 又稱為 GPS, Generalized Processor Sharing) [4] 的封包排程方式。

FFQ的設計是讓不同的資料流 (flow) 可以享有不同的服務率 (service rate)，假設在FFQ排程器中共有N個貯列，每個貯列各自對應到不同的服務率 $\phi_1, \phi_2, \phi_3, ..., \phi_N$,則在任何時間$t$，每一個非空 (non-empty)貯列$i$的服務率都恰好為 $\frac{\phi_i}{\sum_{j \in B(t)} \phi_j} C$，其中$B(t)$是所有非空貯列的集合，而$C$則是輸出端的傳輸速率 (link speed)。故在FFQ排程下，每個非空貯列都可以完美的依照其分配到的服務率而得到應有的封包傳輸服務。不過事實上FFQ是不可能做到的，主要原因在於它有兩項假設，一是排程器可以同時對所有貯列提供傳輸服務，另一則是資料流量 (traffic) 是可以無限切割的。

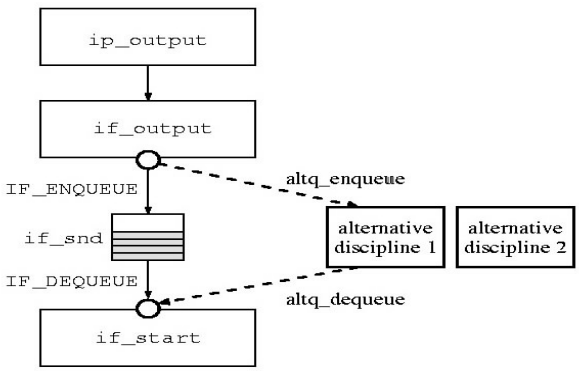
WFQ是最為近似於FFQ的排程方法之一，其運作原理是當每個封包在進入封包排程器時，封包同樣會依照標頭所攜帶的資訊安置到不同的貯列，每個貯列都會對應到不同的服務率，而當封包在放進貯列前會先計算出該封包在FFQ排程中所對應的傳送結束時間，此時間點又稱為虛擬結束時間(virtual finish time)，當排程器在時間t要傳送下一個封包時，會比較各個非空貯列中第一個封包的虛擬結束時間，取出虛擬結束時間最早者加以傳送。

3. 系統架構

在系統架構方面，我們選擇以FreeBSD 3.2 Release此作業系統為系統發展的平台，並將ALTQ 1.2安裝進作業系統的核心 (kernel) 中，以作為實作封包分類器及封包排程器的基礎架構。

ALTQ[1] 的全名為 “Alternate Queueing”，是一個實作封包排程的基礎架構 (framework)，它讓我們可以在UNIX的作業系統環境下把自己的封包分類器及封包排程器加入作業系統的核心之中，使得以個人電腦實作出來的路由器雛型可以提供基本的傳輸服務品質。

ALTQ的設計原理如圖一所示：



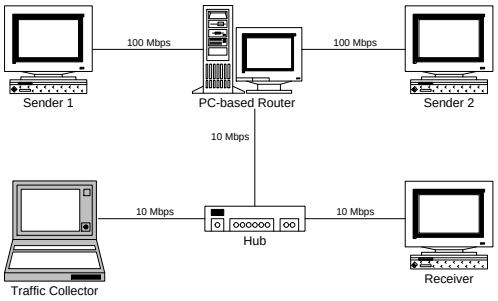
圖一、 ALTQ基本設計原理示意圖

這個架構位於網路卡運作的輸出(output)端，其中 *if_snd*是代表貯列(queue)的資料結構，*enqueue*及*dequeue*的各代表將封包放進貯列及拿出貯列的動作，分別由*if_output*及*if_start*兩個函數執行，在不指定特定排程器的情況下，這兩個函數會呼叫*IF_ENQUEUE*及*IF_DEQUEUE*兩個作業系統核心內既有的巨集(macro)，使用的排程方法是FIFO (first-in-first-out)。但若路由器的管理者指定其他的排程方式，則ALTQ會依據排程方式所對應的封包分類器及封包排程器，呼叫對應的函數(如圖一中的

altq_enqueue 及 *altq_dequeue*) 來取代 *IF_ENQUEUE*及*IF_DEQUEUE*。

4. 實驗概述

為驗證排程器的效能，我們設計了一系列實驗，實驗環境如圖二所示。



圖二、實驗環境示意圖

根據不同的排程方法，我們的實驗主要分為兩大部份。在priority queueing方面，我們用IP封包標頭的「服務類別」(Type of Service) 的欄位做為封包分類器(packet classifier)區分封包的依據來搭配封包排程器；而WFQ的封包分類器則使用per-flow queueing的分類方法，亦即每個資料流 (flow) 會有自己所屬的貯列。

從實驗結果可以驗證幾個結論：(a) priority queueing的確可以對不同服務類別的封包提供不同的傳輸服務品質，不過很難將傳輸服務品質予以量化。 (b) WFQ的特性是近似於FFQ的排程方式，因此每一個資料流都可以公平地(fairly)依照其分配到的服務率得到對應比率的頻寬，使傳輸服務品質能夠精準的量化。 (c) WFQ具有work- conserving的特性，亦即封包排程器內若有任何一個貯列有封包待傳，則封包排程器必須盡力將封包傳送出去，因此WFQ在多餘頻寬未被使用時，可調配給其他有需要的資料流使用藉此提高排程器的使用率(utilization)。 (d) WFQ具有隔離性(flow isolation)，亦即每個資料流的流量變化不會影響其他資料流應享的頻　。

5. 結論

本研究以ALTQ為系統架構，實作Priority Queueing與WFQ這兩種理論的排程器，同時我們也設計數個實驗驗證這兩種排程理論，由實驗結果發現這兩種排程方法的確可以對不同類別的封包給予不同的傳送待遇，藉此讓封包得到不同的傳輸服務品質。

目前我們正在進行和RSVP的整合。因此如IETF(Internet Engineering Task Force) 所定義的整合服務(Integrated Service)，每一個資料流可以藉由 RSVP(Resource Reservation Protocol)和路由器溝通 [2]，一方面做資源保留，另一方面進行資料流的建立及資源保留狀態的維持，以便於路由器的封包排程達到資源保留的效果，讓每個資料流得到應享的傳輸服務品質。而另一個 IETF 所定義的差異化服務(Differentiated Service)[5]，雖然較具擴充性(scalability)，但仍需借助路由器的封包排程，才有可能精確提供不同等級的服務水準。

6. 參考文獻

[1] Kenjiro Cho, "*A Framework for Alternate Queueing: Towards Traffic Management by PC-UNIX-Based Routers*", Proceedings of USENIX 1998 Annual Technical Conference, New Orleans LA, June 1998

[2] *Wroclawski*, J., "*The Use of RSVP with IETF Integrated Services*", RFC 2210, Sep. 1997

[3] Parekh, A., Gallager, R., "*A generalized processor sharing approach to flow control – the single node case*", IEEE/ACM Transactions on Networking, Vol. 1, No.3, pp.344-357, June 1993

[4] Demers, A., Keshav, S., and Shenker, S., "*Analysis and simulation of a fair queueing algorithm*", Journal of Internetworking Research and Experience, pp.3-26, Oct. 1990

[5] Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z., Weiss, W., "*An Architecture for Differentiated Services*", RFC 2475, Dec. 1998