Wireless Scheduling Method Considering Bursty Channel Errors

Moo Ryong Jeong,† Hiroyuki Morikawa†
and Tomonori Aoyama†

To provide Quality-of-Service(QoS) in wireless packet networks, fair scheduling algorithm should be able to deal with bursty and location-dependent wireless channel errors. In this paper, we propose WGPS(Wireless General Processor Sharing) as a fluid-flow level wireless fair scheduling and PWGPS(Packetized Wireless General Processor Sharing) as a packet-by-packet scheduling algorithm realizing WGPS. WGPS is an extension of GPS(Generalized Processor Sharing), the fluid-flow level fair scheduling in wired networks, and operates different from GPS only when there is any flow suffering channel errors. While a flow is in burst error state, the flow is excluded from scheduling and the other flows that have good channel for transmission are scheduled according to their service shares. In return for the lost service, however, the flow having suffered burst error state is compensated by increasing its service share by a prescribed amount when its channel has recovered. PWGPS approximates WGPS by serving packets in the order of service finish times as determined by WGPS. According to WGPS and PWGPS, QoS can be improved despite bursty channel errors since they provide fair service allocation among the flows regardless of their channel conditions.

1. Introduction

Fair scheduling is considered useful for providing quality of service (QoS) in an integrated services network since it ensures fair bandwidth allocation among all backlogged flows and, if combined with admission control, provides bounded end-to-end delay guarantee to a leaky-bucket constrained flow[1),2)]. In the wireline, fair scheduling is defined in the context of GPS(Generalized Processor Sharing) where each flow is assigned a fair service share and the service is instantaneously distributed among the backlogged flows in proportion to their shares[1),3)]. Since fair allocation in GPS requires that the scheduler serve multiple flows simultaneously and that the traffic be infinitely divisible as opposed to practical packet-by-packet schedulers, GPS is approximated by serving a flow's packet at a time in the order of service finish times determined by GPS[1),3)~5)].

In the wireless networks, fair scheduling is even harder. Unlike the wireline, wireless links suffer bursty channel errors at high bit error rates. If a scheduled flow is fallen into burst error state, its data transmission will be unsuccessful and the work done by the scheduler will be gone in vane. Further, the bursty channel errors are location-dependent. This location-dependency can make only part of the backlogged flows to suffer the burst errors, while the channel conditions of other flows are good enough for data transmission. As a result, wireless scheduler may be temporarily unfair during a flow's burst error state, serving other error-free flows more and leaving the QoS of the errored flow degraded. There were some researches which try to solve this problem by compensating the errored flows after the channel states have recovered[6)~8)]. The previous researches, however, are deficient in that QoS of the flows other than the ones being compensated can be degraded during compensation[6),7)] or the compensation itself is not performed in a fair manner among the errored flows[8)].

In this paper, we propose WGPS (Wireless General Processor Sharing) as a wireless fair scheduling in a fluid-flow level and PWPGS (Packetized Wireless General Processor Sharing) as a packet-by-packet version of WGPS. WGPS is an extension of GPS for dealing with location-dependent bursty channel errors of wireless networks. WGPS operates different from GPS only when there is a flow suffering channel errors. Although the flows are excluded from scheduling when they are in burst error states, they are compensated later by increasing their service shares by a prescribed amount Δ until the extra services given by the increased service shares become equal to the lost services during the errored periods. Since the bandwidth for increased service shares can be pre-allocated and the compensations are performed by increased service shares of the same

† School of Engineering, The University of Tokyo

amount Δ, WGPS and PWGPS ensure fairness in compensation and better QoS regardless of channel conditions.

The rest of the paper is organized as follows. In Section 2, we describe GPS in brief. In Section 3, we introduce WGPS and examine its fairness and QoS support. Section 4 discusses how to implement WGPS by a packetized scheduling algorithm. Section 5 discusses the related works and Section 6 concludes the paper.

2. Generalized Processor Sharing

Fair scheduling in the wireline network is defined in the context of GPS(Generalized Processor Sharing). In GPS, each flow is assigned a fair service share and the service is instantaneously distributed among the backlogged flows in proportion to their shares[1),3)]. Since GPS requires that the scheduler serve multiple flows simultaneously and that the traffic be infinitely divisible, GPS is defined based on the fluid model as follows.

Definition 2.1 A **GPS server** with rate R is a fluid-flow server having following properties:

(1) Each flow i is associated with its service share $\phi_i > 0$.

(2) Let $W_i(t_1, t_2)$ be the amount of flow i traffic served in an interval $(t_1, t_2]$ and $B(t)$ be set of all flows backlogged at time t. Then, any flow i that is continuously backlogged in the interval $(t_1, t_2]$ satisfies

$$W_i(t_1, t_2) = \phi_i \cdot (V(t_2) - V(t_1)) \quad (1)$$

where $V(t)$ is given by

$$\frac{\partial V}{\partial t} = \frac{R}{\sum_{i \in B(t)} \phi_i} \quad (2)$$

Note from eq.(2) that $V(t)$, which is often referred as virtual time, increases in a rate which amounts to the server rate R divided by the instant total network load $\sum_{i \in B(t)} \phi_i$. Thus $V(t)$ can be interpreted as an amount of service provided by the time t for a flow which has service share of one unit. Likewise, $V(t_2) - V(t_1)$, the difference of two virtual times $V(t_2)$ and $V(t_1)$, can be interpreted as an amout of service provided for the flow of one unit service share in an interval $(t_1, t_2]$.

Note also from eq.(1) that each backlogged flow is provided with service which amounts to the virtual time difference multiplied by the flow's service share. This allocation of service ensures fairness among the backlogged flows. Finally it should be noted that since virtual time increases in the minimum rate when all the active flows are backlogged, service provided by GPS is guaranteed to be no less than the amount of service provided when all the active flows are backlogged. Thus, if combined with admission control which maintains the guaranteed service to be more than the requested service, QoS of the flows can be provided by GPS.

3. Wireless Generalized Processor Sharing

3.1 Wireless Scheduling Environment

We consider a packet cellular network where each cell is served by a base station. Specifially, we consider centralized packet scheduling in a base station for either uplink or downlink. In either case, scheduler is provided with knowledge of each flow's buffer state. Wireless links being scheduled may suffer bursty and location-dependent channel errors. The resultant channel state of each flow is detected and provided to the scheduler so that the channel state information can be utilized by the scheduler. We assume the channel detection is perfect so that there are no failures in packet transmission.

3.2 Insights of WGPS

To deal with the wireless scheduling environment, we propose WGPS(Wireless Generalized Processor Sharing), a new fluid scheduling model which extends GPS under the following insights:

(1) Since even the backlogged flows may be unable to be served due to their bad channels, WGPS so operates that only the backlogged and good-channel-conditioned (servable) flows are served.

(2) To resolve the unfairness problem during a flow's burst error state, WGPS provide the errored (or lagging) flow with an extra compensational service so that the flow can catch up with the other well-conditioned flows.

(3) If confined in a short-term in which a flow is being compensated, there is a unfairness between the flows being compensated and the other ordinary flows which are not being compensated. The unfairness, however, is suppressed within a limit and does not incur any QoS degradations under the requested level if the flows have clear channels. To this end, only lagging flows are allowed to have memory of past service and the compensation services for lagging flows are pre-allocated in WGPS. Fur-

ther, all the lagging flows have compensation rate exactly in proportion to their service share to ensure fairness in compensation.

3.3 WGPS server

To incorporate the first insight of WGPS, we introduce the concept of servability in backlogged flows. We call a backlogged flow **servable** if its channel condition is good. Otherwise, we call the backlogged flow **unservable**. Then the formal definition of WGPS is as follows.

Definition 3.1 A **WGPS server** with rate R and compensation index Δ is a fluid-flow server with following properties:

(1) Each flow i is associated with its service share $\phi_i > 0$.

(2) Each flow i is associated with its compensation counter $W_i^C(t)$, which is initialized to 0 when the flow i becomes backlogged.

(3) Let $W_i(t_1, t_2)$ be the amount of flow i traffic served in an interval $(t_1, t_2]$ and $S(t)$ be set of all flows servable at time t. Then, any flow i that is continuously servable in the interval $(t_1, t_2]$ satisfies

$$W_i(t_1, t_2) = \phi_i(t) \cdot (V(t_2) - V(t_1)) \quad (3)$$

where $\phi_i(t)$ and $V(t)$ is given by

$$\phi_i(t) = \begin{cases} \phi_i \cdot (1 + \Delta), & \text{if } W_i^C(t) > 0 \\ \phi_i, & \text{otherwise,} \end{cases} \quad (4)$$

$$\frac{\partial V}{\partial t} = \frac{R}{\sum_{i \in S(t)} \phi_i(t)} \quad (5)$$

and $W_i^C(t)$ is given by

$$\frac{\partial W_i^C}{\partial t} \quad (6)$$

$$= \begin{cases} \phi_i \frac{\partial V}{\partial t} & \text{if flow } i \text{ is } \textit{unservable} \text{ at } t \\ -\phi_i \cdot \Delta \frac{\partial V}{\partial t} & \text{if flow } i \text{ is servable at } t \text{ and } W_i^C(t) > 0 \\ 0 & \text{otherwise} \end{cases}$$

WGPS as defined in the definition 3.1 will be explained with reference to Fig.1. In Fig.1, flow i's traffic served per unit virtual time (hereinafter, virtual service rate) is plotted with respect to virtual time. Remind that, in GPS server, each backlogged flow is provided with service exactly in proportion to its service share ϕ_i. In WGPS, however, the backlogged flow may be unable to be served as shown in the interval $(V(t_b), V(t_r)]$ when it is in a burst error state or it may also be served more than its share when it is being compensated as in the interval $(V(t_r), V(t_c)]$.

We chronologically investigate Fig.1 to understand WGPS in more detail. Let the transmission of first packet of flow i be finished at virtual time $V(t_a)$ by when the flow is serviced in the virtual service rate of ϕ_i. In the course of second packet's transmission, the channel falls into the burst error state at virtual time $V(t_b)$. From the virtual time $V(t_b)$ flow i is excluded from the scheduling and thus provided no service. Instead, its compensation counter W_i^C is increased in a virtual rate of $\phi_i \frac{\partial V}{\partial t}$, accumulating lost service for later compensation. At virtual time $V(t_r)$, the channel is recovered and the service of flow i's traffic is restarted. Note that from the time $V(t_r)$, the virtual service rate is increased to a virtual rate of $\phi_i \cdot (1 + \Delta)$ since its compensation counter is positive. The rate increase $\phi_i \cdot \Delta$ contributes to the compensation of service lost in the previous burst error state. In return, the compensation counter W_i^C is decreased with a rate of $\phi_i \cdot \Delta \frac{\partial V}{\partial t}$ while the flow is being compensated, charging for compensated service. When the compensation counter W_i^C is decreased to be 0, say at virtual time $V(t_c)$, flow i has been fully compensated and thus its virtual service rate is returned to its normal service share ϕ_i. Consequently, in WGPS, if a flow has no memory of suffering bad channels or has been fully compensated, it is served with a service share of ϕ_i while it is servable. If a flow has memory of suffering bad channels and has not yet been fully compensated, on the other hand, it is served in an increased service share of $\phi_i \cdot \Delta$. In this way, WGPS accomplishs full compensation of errored flow i in a finite time, with the normalized deviation from the steady state being suppressed by Δ.

3.4 QoS Support of WGPS

In this section, we present the QoS support of WGPS. Following two theorems explain the fair service allocation aspects of WGPS which realize QoS support in wireless networks. Following theorem regarding short-term fairness is obvious from the definition of WGPS.

Theorem 1 Let flow i and flow j be two flows in a WGPS server which are servable in interval $(t_1, t_2]$. If we denote with $C(t)$ the set of all flows whose compensation counter is positive at time t, then $W_i(\tau_1, \tau_2)$ and $W_j(\tau_1, \tau_2)$ satisfy

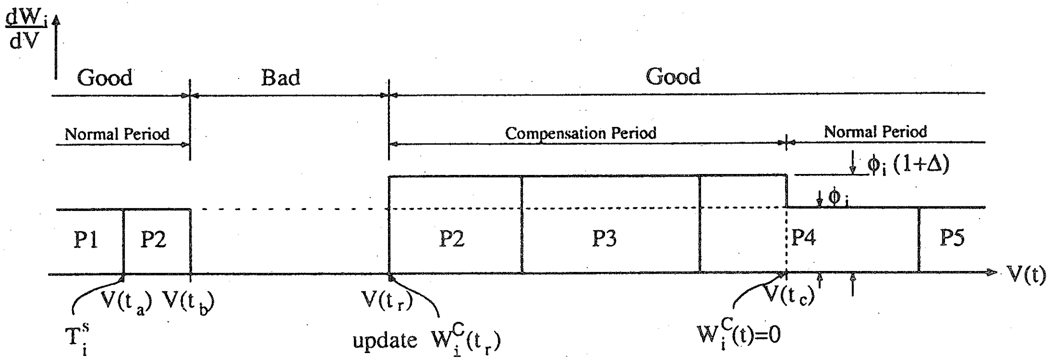

Fig. 1 compensation in WGPS

$$\begin{cases} \frac{W_i(\tau_1,\tau_2)}{\phi_i} = \frac{W_j(\tau_1,\tau_2)}{\phi_j} \\ \quad = V(\tau_2) - V(\tau_1) \\ \qquad if \forall t \in (\tau_1,\tau_2], i,j \notin C(t) \\ \frac{W_i(\tau_1,\tau_2)}{\phi_i} = \frac{W_j(\tau_1,\tau_2)}{\phi_j} \\ \quad = (1+\Delta)\cdot(V(\tau_2) - V(\tau_1)) \\ \qquad if \forall t \in (\tau_1,\tau_2], i,j \in C(t) \\ \frac{W_i(\tau_1,\tau_2)}{\phi_i} - \frac{W_j(\tau_1,\tau_2)}{\phi_j} \\ \quad = \Delta\cdot(V(\tau_2) - V(\tau_1)) \\ \qquad if \forall t \in (\tau_1,\tau_2], i \in C(t), j \notin C(t) \end{cases} \quad (7)$$

proof. This can be easily proved by applying eqs.(3,4) to $W_i(\tau_1,\tau_2)$ and $W_j(\tau_1,\tau_2)$. □

Theorem 1 shows that WGPS is fair between two flows if both of the two or neither of the two flows is being compensated. Note that amount of service provided by WGPS to a servable flow is guaranteed to be no less than the amount of service provided when all the other active flows are backlogged and being compensated. Thus, if combined with admission control, QoS can be provided to good-channel-conditioned flows in WGPS since this service guarantee is independent of other flows. Theorem 1 also shows that if one of the two flows has sufferred errored channel and therefore is being compensated, there exists an unfairness between the two. The unfairness, however, increases only in proportion to virtual time, which is inevitable in order to gradually compensate the errored flows for bad-channel-state periods. Thus if fairness is considered in a long-term which includes the previous errored periods, WGPS shows following long-term fairness characteristics.

Theorem 2 Let a flow i in a WGPS server is unservable in interval $(t_b, t_r]$. If the flow i is continuously servable for a interval $(t_r, t_c]$ whose virtual time lengh equals $\frac{V(t_r) - V(t_b)}{\Delta}$, then compensation of flow i is finished at time t_c for the service lost in the unservable interval $(t_b, t_r]$.

proof. For the interval $(t_b, t_c]$, let us denote with W_1 the amount of service which should have been provided to the flow i when there were no channel errors and denote with W_2 the amount of service actually provided by WGPS server. Then it is sufficient to prove that W_1 and W_2 are same. Obviously, $W_1 = \phi_i \cdot (V(t_c) - V(t_b))$. In examining W_2, remind that the flow i was served only in the servable interval $(t_r, t_c]$. From the eqs.(3, 4), we can get

$$\begin{aligned} W_2 &= \phi_i \cdot (1+\Delta)\cdot(V(t_c) - V(t_r)) \\ &= \phi_i \cdot (V(t_c) - V(t_r)) \\ &\quad + \phi_i \cdot \Delta \cdot (V(t_c) - V(t_r)) \end{aligned}$$

Since $V(t_c) - V(t_r)$ is given by $\frac{V(t_r) - V(t_b)}{\Delta}$ in this theorem, we get

$$\begin{aligned} W_2 &= \phi_i \cdot (V(t_c) - V(t_r)) \\ &\quad + \phi_i \cdot (V(t_r) - V(t_b)) \\ &= \phi_i \cdot (V(t_c) - V(t_b)) \end{aligned}$$

Therefore, $W_1 = W_2$, which proves the theorem. . □

Theorem 2 shows that, in WGPS, a flow which was in errored state for a finite time can be fully compensated also in a finite time. In other words, by compensation, WGPS ensures long-term QoS support even for a flow which has suffered errored channel. Remind from the theorem 1 that the compensation in WGPS is performed so carefully as to ensure short-term fairness among the flows. Since WGPS realizes fair service allocation regardless of the channel states, it can support better QoS in wireless networks.

4. Packetized Wireless Generalized Processor Sharing

In this section, we investigate Packetized Wireless General Processor Sharing (PWGPS) algorithm. One of the well-known methods of simulating a fluid-level scheduler by packet-by-packet scheduling is WFQ (Weighted Fair Queueing)[3]. In WFQ, service start time and service finish time of each packets are calculated by using the notions of virtual time and are used in selecting a packet to be transmitted. WFQ choose, for transmission, a flow whose HOL (Head Of Line) packet has the smallest finish time in GPS server. Generally, this method of WFQ is applicable in order for PWGPS to simulate WGPS. To devise the PWGPS algorithm, it is required to obtain the service start time and the service finish time of each packet in the WGPS server. Following theorems shows us how to calculate them without real WGPS server.

Theorem 3 Let us denote with $t^s_{i,k}$ the service start time of a packet p^k_i of flow i and denote with $t^f_{i,k}(> t^s_{i,k})$ the service finish time of the packet p^k_i. Let us also denote with $W^C_{i,k}$ the amount of compensation service provided in the interval $(t^s_{i,k}, t^f_{i,k}]$ for the packet p^k_i. If the flow i is continuously servable during the interval $(t^s_{i,k}, t^f_{i,k}]$, then $W^C_{i,k}$ is given as follows

$$W^C_{i,k} = \begin{cases} \dfrac{L(p^k_i)\cdot\Delta}{1+\Delta} & \text{if } W^C_i(t^s_{i,k}) \geq \dfrac{L(p^k_i)\cdot\Delta}{1+\Delta} \\ W^C_i(t^s_{i,k}) & \text{otherwise} \end{cases} \quad (8)$$

where $L(p^k_i)$ is a length of packet p^k_i.

proof. Without loss of generality, we can assume that $W^C_i(t) = 0$ at time $t = t^w_{i,k}(\geq t^s_{i,k})$.

Case 1: $W^C_i(t^s_{i,k}) \geq \dfrac{L(p^k_i)\cdot\Delta}{1+\Delta}$. We firstly show that $t^w_{i,k} \geq t^f_{i,k}$ in this case. We prove it by contradiction. Let's assume that $t^w_{i,k} < t^f_{i,k}$. Then, from the definition of $\dfrac{\partial W^C_i}{\partial t}$ and from the fact that $W^C_i(t^w_{i,k}) = 0$, we get

$$W^C_i(t^w_{i,k}) = W^C_i(t^s_{i,k}) + \int_{t^s_{i,k}}^{t^w_{i,k}} \frac{\partial W^C_i}{\partial t}dt = 0$$

By applying eq.(6) to the above equation, we get

$$\begin{aligned} &W^C_i(t^w_{i,k}) \\ &= W^C_i(t^s_{i,k}) + \int_{t^s_{i,k}}^{t^w_{i,k}} -\phi_i\cdot\Delta\frac{\partial V}{\partial t}dt \\ &= W^C_i(t^s_{i,k}) - \phi_i\cdot\Delta\cdot\left(V(t^w_{i,k}) - V(t^s_{i,k})\right) \\ &= 0 \end{aligned} \quad (9)$$

By rearranging eq.(9),

$$W^C_i(t^s_{i,k}) = \phi_i\cdot\Delta\cdot\left(V(t^w_{i,k}) - V(t^s_{i,k})\right)$$

Since $W^C_i(t^s_{i,k}) \geq \dfrac{L(p^k_i)\cdot\Delta}{1+\Delta}$, we get

$$\phi_i\cdot\Delta\cdot\left(V(t^w_{i,k}) - V(t^s_{i,k})\right) \geq \frac{L(p^k_i)\cdot\Delta}{1+\Delta}$$

By rearranging, the above equation becomes

$$\phi_i\cdot(1+\Delta)\cdot\left(V(t^w_{i,k}) - V(t^s_{i,k})\right) \geq L(p^k_i)$$

The L.H.S. of the above equation is the amount of service provided in the interval $(t^s_{i,k}, t^w_{i,k})$ for the flow i. Thus it means that the service provided in the interval $(t^s_{i,k}, t^w_{i,k})$ is not less than the length of packet p^k_i. Thus, service of packet p^k_i has been finished on or before the time $t^w_{i,k}$, which is a contradiction.

Then, using the fact that $t^w_{i,k} \geq t^f_{i,k}$, we prove this theorem for the case $W^C_i(t^s_{i,k}) \geq \dfrac{L(p^k_i)\cdot\Delta}{1+\Delta}$. Since $t^w_{i,k} \geq t^f_{i,k}$, $W^C_i(t) > 0, \forall t \in (t^f_{i,k}, t^f_{i,k}]$. Thus, as in eq.(4), the flow i is served with an increased service rate of $\phi_i\cdot(1+\Delta)$ in the interval $(t^s_{i,k}, t^f_{i,k}]$ and the service provided during the interval $(t^f_{i,k}, t^f_{i,k}]$ amounts to $L(p^k_i)$.

$$\begin{aligned} L(p^k_i) &= W_i(t^s_{i,k}, t^f_{i,k}) \\ &= \phi_i\cdot(1+\Delta)\cdot\left(V(t^f_{i,k}) - V(t^s_{i,k})\right) \end{aligned} \quad (10)$$

Since $W^C_i(t)$ is decreased as much as the provided compensation service and the decrease rate of $W^C_i(t)$ is $\phi_i\cdot\Delta\dfrac{\partial V}{\partial t}$ in the interval $(t^f_{i,k}, t^f_{i,k}]$, we get the following result from eq.(6).

$$\begin{aligned} W^C_{i,k} &= W^C_i(t^s_{i,k}) - W^C_i(t^f_{i,k}) \\ &= -\int_{t^s_{i,k}}^{t^f_{i,k}} \frac{\partial W^C_i}{\partial t}dt \\ &= \int_{t^s_{i,k}}^{t^f_{i,k}} \phi_i\cdot\Delta\frac{\partial V}{\partial t}dt \\ &= \phi_i\cdot\Delta\cdot\left(V(t^f_{i,k}) - V(t^s_{i,k})\right) \end{aligned} \quad (11)$$

By applying eq.(10) into eq.(11), we get

$$\begin{aligned} W^C_{i,k} &= \phi_i\cdot\Delta\cdot\frac{L(p^k_i)}{\phi\cdot(1+\Delta)} \\ &= \frac{L(p^k_i)\cdot\Delta}{1+\Delta} \end{aligned} \quad (12)$$

Case 2: $0 \le W_i^C(t_{i,k}^s) < \frac{L(p_i^k)\cdot\Delta}{1+\Delta}$. In this case, $t_{i,k}^w < t_{i,k}^f$, which means that $W_i^C(t_{i,k}^s)$ is used up before the service of packet p_i^k is finished. Since $W_i^C(t)$ is decreased as much as the provided compensation service, we can conclude that

$$W_{i,k}^C = W_i^C(t_{i,k}^s) \qquad (13)$$

□

Theorem 3 teaches us how much compensation service is used to serve a packet. By using theorem 3, we can know how much the compensation counter is to be decreased after serving a packet. By using theorem 3, we can also derive the service start time and the service finish time of a HOL packet. Following theorem tells us the details.

Theorem 4 Let us denote the service start time and the service finish time of a flow i in WGPS server when the flow's current HOL packet is p_i^k be T_i^{s-} and T_i^{f-}, respectively. If we assume that the packet p_i^k is successfully transmitted at time t_d and that the compensation counter after the transmission is $W_i^C(t_d^+)$, then the service start time T_i^{s+} and the service finish time T_i^{f+} of the next HOL packet p_i^{k+1} are given as follows

$$T_i^{s+} = T_i^{f-} \qquad (14)$$

$$T_i^{f+} = \begin{cases} T_i^{s+} + \dfrac{L(p_j^{k+1})}{\phi_i\cdot(1+\Delta)} \\ \quad if W_i^C(t_d^+) \ge \dfrac{L(p_i^{k+1})\cdot\Delta}{1+\Delta}, \\ T_i^{s+} + \dfrac{L(p_i^{k+1}) - W_i^C(t_d^+)}{\phi_i} \\ \quad otherwise \end{cases} \qquad (15)$$

where $L(p_j^{k+1})$ is a length of packet p_j^{k+1}.

proof. **Proof of eq.(14)** In WGPS, packets are serviced in a back-to-back manner if the flow is backlogged. Thus it is obvious that eq.(14) holds.

Proof of eq.(15) if $W_i^C(t_d^+) \ge \frac{L(p_i^{k+1})\cdot\Delta}{1+\Delta}$ then, from theorem 3, flow i is provided with $W_{i,k}^C = \frac{L(p_i^{k+1})\cdot\Delta}{1+\Delta}$ amount of service as compensation, in addition to $\phi\cdot(T_i^{f+} - T_i^{s+})$ amount of normal service that is provided as its fair share. Since these services is to be used for serving the packet p_i^{k+1}, we get

$$\frac{L(p_i^{k+1})\cdot\Delta}{1+\Delta} + \phi\cdot(T_i^{f+} - T_i^{s+}) = L(p_i^{k+1})$$

By rearranging the above equation, we get

$$\phi\cdot(T_i^{f+} - T_i^{s+}) = \frac{L(p_i^{k+1})}{1+\Delta}$$

which yields the desired result.

if $W_i^C(t_d^+) < \frac{L(p_i^{k+1})\cdot\Delta}{1+\Delta}$, $W_{i,k}^C = W_i^C(t_d^+)$ from theorem 3. Thus, we get

$$W_i^C(t_d^+) + \phi\cdot(T_i^{f+} - T_i^{s+}) = L(p_i^{k+1})$$

By rearranging the above equation, we get

$$\phi\cdot(T_i^{f+} - T_i^{s+}) = L(p_i^{k+1}) - W_i^C(t_d^+)$$

which yields the desired result. □

4.1 PWGPS Algorithm

V(t)	virtual time at time t
S(t)	set of all servable flows at time t
C(t)	set of all compensation flow at time t
T_i^s	service start time of flow i's HOL packet
T_i^f	service finish time of flow i's HOL packet
W_i^C	amount of service to be compensated for flow i
L_i^C	length of unserved HOL packet of flow i
V_i^l	reference virtual time from which compensation of flow i begins

Table 1 System parameters in PWGPS

We describe the detailed algorithm of the P-WGPS with reference to Fig.2. For the summary of system parameters in PWGPS, refer Table 1. PWGPS is comprised of four main modules: enqueueing module, channel corruption process module, channel recovery module, and dequeueing module.

Enqueueing module enques the arriving packets (step 1) and prepare its service start time and service finish time if the arriving packet makes its flow to be newly backlogged (step 2-1, 2-3, 2-4). If the flow has already been backlogged, these service times are updated after serving its packet. Note that compensation counter is initialized to be 0 to make PWGPS memoryless of the previous backlogged period.

Channel corruption process module is initiated when a flow's channel is gone bad. The flow is excluded from the set of servable flows $S(t)$ (step 1) and become a flow to be compensated (step 2). In order to measure the lost service while a flow is in a bad channel state, virtual time when the bad channel state begins is stored in V_i^l (step 3).

Channel recovery process module is initiated when a flow's channel is recovered. The flow is inserted in the set of servable flow (step

Enqueueing module: (on **arrival** of packet p_i^k of flow i at time t_a)
1. Enqueue($Queue_i(t_a), p_i^k$)
2. if ($i \notin S(t_a^-)$ and $i \notin C(t_a^-)$) //flow i is newly backlogged//
2-1. $S(t_a^+) = S(t_a^-) \cup \{i\}$
2-2. $W_i^C(t_a^+) = 0$
2-3. $T_i^s = V(t_a)$ //prepare HOL packet's time//
2-4. $T_i^f = T_i^s + \frac{L(p_i^k)}{\phi_i}$

Channel corruption process module: (on a flow i's channel **going bad** at time t_b)
1. $S(t_b^+) = S(t_b^-) \cap \{i\}^C$
2. if ($i \notin C(t_b^-)$) $C(t_b^+) = C(t_b^-) \cup \{i\}$
3. $V_i^l = V(t_b)$

Channel recovery process module:
(on a flow i's channel **recovering** at time t_r;
flow i's HOL packet is p_i^k)
1. $S(t_r^+) = S(t_r^-) \cup \{i\}$
2. $W_i^C(t_r^+) = W_i^C(t_r^-) + \phi_i \cdot (V(t_r) - V_i^l)$ //calculate amount of compensation//
3. $L_i^C = L(p_i^k) - \phi \cdot (V_i^l - T_i^s)$ //length of unserved part of HOL packet//
4. $T_i^s = V(t_r)$ //prepare HOL packet's time//
5. if ($W_i^C(t_r^+) \geq \frac{L_i^C \cdot \Delta}{1+\Delta}$) $T_i^f = T_i^s + \frac{L_i^C}{\phi_i \cdot (1+\Delta)}$
5′. else $T_i^f = T_i^s + \frac{L_i^C - W_i^C(t_r^+)}{\phi_i}$ //compensated if $W_i^C(t_r^+) > 0$//

Dequeueing module:
1. if ($T_i^f = \min_{j \in S(t)}\{T_j^f\}$) //serve flow having minimum service time//
1-1. Dequeue($Queue_i(t)$, HOL packet of flow i)
(on **departure** of a packet p_i^k of flow i at time t_d)
2. if ($W_i^C(t_d^-) \leq \frac{L(p_j^k) \cdot \Delta}{1+\Delta}$) //charge for compensation//
2-1. $W_i^C(t_d^+) = 0$
2-2. $C(t_d^+) = C(t_d^-) \cap \{i\}^C$
2′. else $W_i^C(t_d^+) = W_i^C(t_d^-) - \frac{L(p_i^k) \cdot \Delta}{1+\Delta}$
3. if ($Queue_i(t_d^+) \neq$ empty)
3-1. $T_i^s = T_i^f$ //prepare next packet's service time//
3-2. if ($W_i^C(t_d^+) \geq \frac{L(p_i^{k+1}) \cdot \Delta}{1+\Delta}$)
$T_i^f = T_i^s + \frac{L(p_j^{k+1})}{\phi_i \cdot (1+\Delta)}$
3-2′. else $T_i^f = T_i^s + \frac{L(p_i^{k+1}) - W_i^C(t_d^+)}{\phi_i}$ //compensated if $W_i^C(t_r^+) > 0$//
3′. else
3′-1. if ($V(t_f) = T_i^f$ and $Queue_i(t_f)$ = empty) //end of flow i's backlogged period//
3′-1-1. $S(t_f^+) = S(t_f^-) \cap \{i\}^C$
3′-1-2. $C(t_f^+) = C(t_f^-) \cap \{i\}^C$
3′-1-3. $W_i^C(t_d^+) = 0$

Fig. 2 PWGPS Algorithm

1). Then amount of service to be compensated is calculated using the virtual times V_i^l, the time of channel corruption, and $V(t_r)$, the time of channel recovery (step 2). Since the HOL packet of the recovered flow has been partially served before the channel corruption, we adjust the packet size to be that of the unserved part of the packet (step 3). Step 4, 5, and 5' are regarding to the packet retagging. PWGPS retags the service times at the time of channel recovery (step 4). In doing so, the packet size that was adjusted in the (step 3) are used in the theorem 4 (step 5, 5').

Dequeueing module selects a packet as determined by WGPS and deques the packet (step 1). After the transmission of a flow's packet, its compensation counter is updated as determined by theorem 3(step 2, 2'). If the flow's queue is still occupied, service start time and service finish time of its HOL packet is updated as in theorem 4(step 3). If the queue is empty, the flow is excluded from the set $S(t)$ because it is not backlogged any more (step 3').

5. Related Work

Lu et al.[6] and Ng et al.[7] noticed the wireless scheduling problem and proposed IWFQ (Idealized Wireless Fair Queueing) and CIF-Q (Channel-condition Independent Packet Fair Queueing), respectively. IWFQ try to resolve the problem through compensation of errored flows by leaving the service times of the errored packets unchanged. Since the packets of errored flows were not served for a time and thus have smaller service finish times than others, they are given absolute priority in scheduling and thus compensated for the lost service. Although QoS of the errored flows can be improved by compensation in IWFQ, QoS of the other flows are degraded since they cannot receive service until the compensation of the errored flow is over.

CIF-Q tries to alleviate the QoS degradation of both of the errored and error-free flows by allowing error-free flows to retain minimal fraction of their service even during the compensation. The other fraction of their service, however, are to be used for the compensation of errored flows and, therefore, QoS of error-free flows are nonetheless degraded in CIF-Q.

Ramanathan et al.[8] proposed LTFS (Long Term Fairness Server) and tried to provide the promised amount of service to all flows whenever their wireless channel is in good state. Thus, portion of total bandwidth is pre-allocated to

LTFS for compensation. Since the compensation uses the pre-allocated bandwidth instead of the fraction of leading flows, there is no QoS degradation during compensation. However, LTFS is deficient in mechanism of assuring fairness among the flows being compensated by a LTFS.

6. Conclusion

In this paper, we introduced WGPS scheduling discipline and its packetized implementation algorithm, PWGPS, for wireless scheduling under the bursty channel errors. According to our scheduling, QoS can be provided to the both of errored and error-free flows regardless of the location-dependent bursty channel errors of the wireless networks. In addition to the QoS support, WGPS and PWGPS have simplicity. In WGPS and PWGPS, measuring of the lost service and control of the compensation can be done in a uniform way by using virtual time and compensation index Δ. Although the PWGPS is explained with reference to WFQ in this paper, it is apparent that other packet scheduling algorithms that approximate GPS can also be used for PWGPS to approximate WGPS.

References

1) A. Parekh and R. Gallager. A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks: The Single-Node Case. *IEEE/ACM Trans. on Networking*, 1(3):344–357, June 1993.
2) A. Parekh and R. Gallager. A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks: The Multiple Node Case. *IEEE/ACM Trans. on Networking*, 2(2):137–150, April 1994.
3) A. Demers, S. Keshav, and S. Shenkar. Analysis and Simulation of A Fair Queueing Algorithm. *Proc. ACM SIGCOMM'89*, pages 1–12, Sep. 1989.
4) J.C.R. Bennett and H. Zhang. WFFQ: Worst-case Fair Weighted Fair Queueing. *Proc. of IEEE INFOCOM'96*, pages 120–128, March 1996.
5) P. Goyal, H. M. Vin, and H. Cheng. Start-time Fair Queueing: A Scheduling Algorithm for Integrated Services Packet Switching Networks. *Proc. ACM SIGCOMM'96*, pages 157–168, Aug. 1996.
6) Songwu Lu, Vaduvur Bharghavan, and Rayadurgam Srikant. Fair Scheduling in Wireless Packet Networks. In *Proc. of ACM SIGCOMM '97*, 1997.
7) T. S. Eugene Ng, Ion Stoica, and Hui Zhang. Packet Fair Queueing Algorithms for Wireless Networks with Location-Dependent Errors. In *Proc. of Infocom '98*, 1998.
8) Parameswaran Ramanathan and Prathima Agrawal. Adapting Packet Fair Queuing Algorithms to Wireless Networks. In *Proc. of MobiCom '98*, 1998.