

## **CHAPTER 4**

### **SIMULATION AND RESULTS ANALYSIS**

Simulation shows the efficiency of feedback-based QoS control. From analysis we can compare the results between simulations with feedback-based QoS control and without feedback-based QoS control. We use Network Simulator 2 (NS-2) for simulation.

#### **4.1 Introduction to NS-2**

NS is a discrete event simulator targeted at networking research. Ns provides substantial support for simulation of TCP, routing, and multicast protocols over wired and wireless (local and satellite) networks.

In this research we use ns2 all in one version 2.26 for simulation because the UMTS extension EURANE is proposed on ns2 version 2.26.

#### **4.2 Enhanced UMTS Radio Access Network Extensions (EURANE)**

Because NS-2 doesn't support UMTS simulation, so we search extensions to provide UMTS simulation. EURANE for NS-2, developed within the European Commission 5th framework project SEACORN, comprises of an additional three nodes, namely the Radio Network Controller (RNC), Base Station (BS) and the User Equipment (UE), whose functionality allow for the support of the following transport channels:

- FACH
- RACH
- DCH
- HS-DSCH

The main functionality additions to NS-2 come in the form of the RLC Acknowledged Mode (AM), Unacknowledged Mode (UM), MAC-d/-c/sh support for RACH/FACH and DCH, and MAC-hs support for HS-DSCH, i.e. HSDPA.

### **4.3 Simulation Scenario**

To see the impact of flows belong to the same service class. We want to figure out that contention will occurs in the same service class without per-flow control.

#### **4.3.1 Topology**

As Fig. 4-1 shows, error model is applied to wireless link for handover and signal fading simulation. In wired network error model is applied to simulate high loading and congestion.

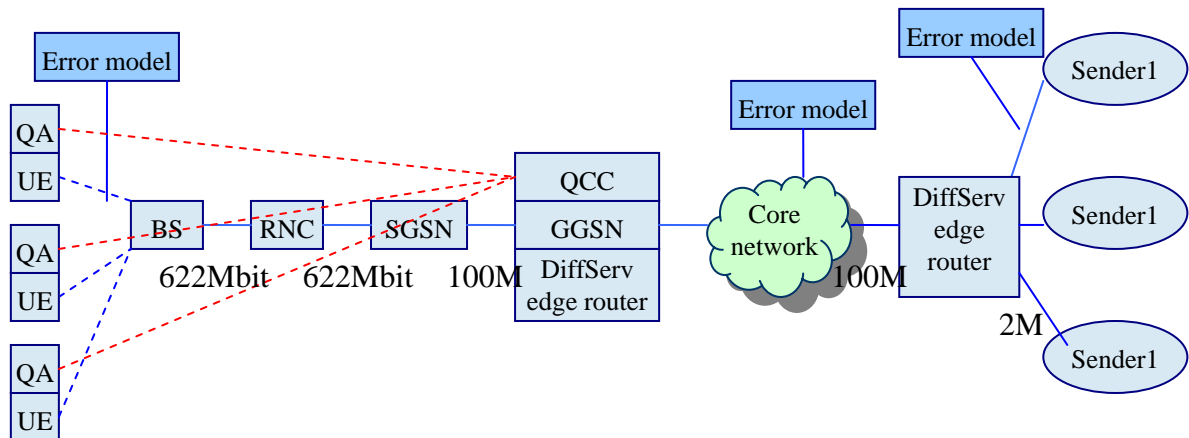


Fig. 4-1 Simulation Topology

### 4.3.2 Error Model

Error model could simulation error and loss in link layer. Unit can be set such as “time”, “byte”, “pkt” and “bit” as error unit. Pkt is packet. Code example was shown below

```

set em_ [new ErrorModel]
$em unit pkt
$em set rate 0.02
$em ranvar [new RandomVariable/Uniform]

$node interface-errormodel $em $i

```

Fig. 4-2 Error Model code

It set packet loss rate to 0.02 and distribute with uniform distribution. Interface-error model is extended by EURANE to provide error in interfaces.

### 4.3.3 Application

Practice service environment is as Fig. 4-3 shows. VoIP and streaming video are encoded to packet data first and then send to network through User Datagram Protocol (UDP) transport. Packets arrival to receiver after travel a several time in network and are decoded by decoder.

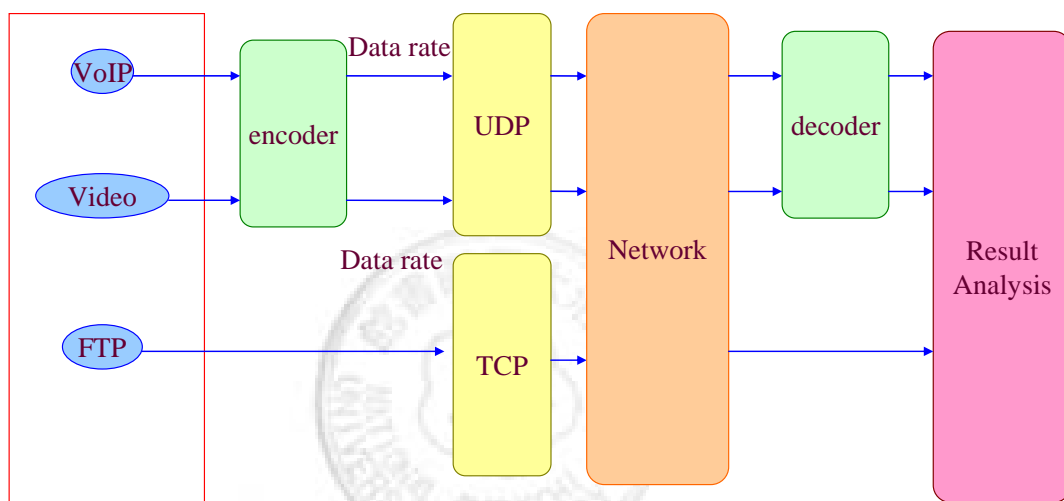


Fig. 4-3 Practice service environment

In simulation the environment is shown in Fig. 4-4. After simulation is done we post analysis the trace file. Transform the bit rate to frame rate for performance analysis.

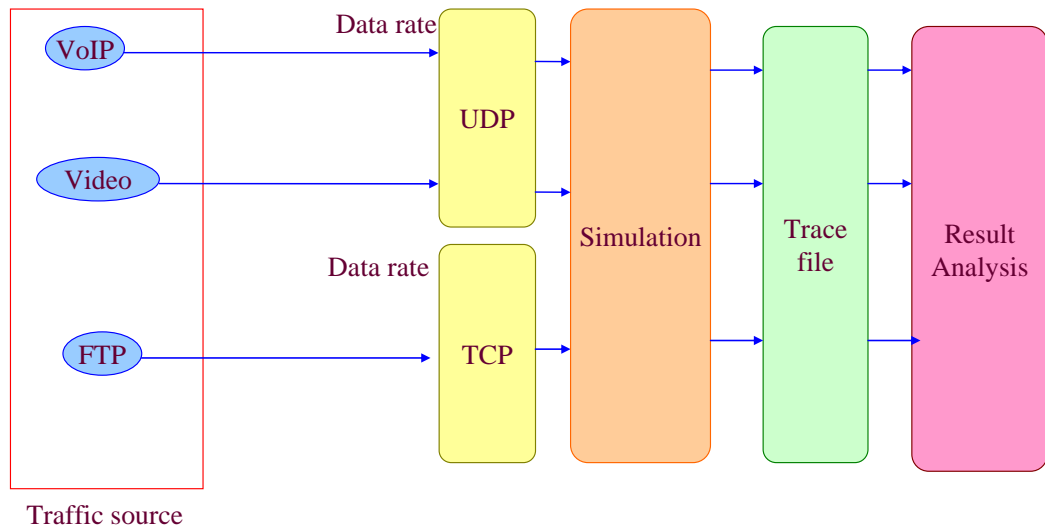


Fig. 4-4 Simulation environment

In this thesis we focus on the QoS control in the same service class. So we use video as our main simulation application which is the most potential application in the future wireless network.

#### 4.3.3.1 Video Application

In International Telecommunication Union (ITU) H.324, H.263 and MPEG-4 codec techniques are used as recommended video codec and adopted by UMTS.

#### 4.3.3.2 Traffic Model

We use video trace file from [21] as video traffic input. The video trace file is generated from tracing the video of Star War IV which length is 60 minutes with H.263 codec and the target bit rate is 256kbit/s. The original frame rate is 30 frames per second and the average

frame size is 4563.53 bytes. For the reason to reduce the influence of different traffic characteristic by using different video trace file. We use the same video trace file to form the same traffic behavior and make the effect of QoS control outstanding.

## **4.4 Result Analysis**

The main results of analysis will focus on throughput, QoS stability, robustness.

### **4.4.1 Throughput**

We record the bandwidth throughput at receiver end. Bandwidth throughput may not represent the real QoS which user received. We translated bandwidth throughput into frame throughput. The average frame size of video source is 4563bytes in H.263 and we obtain the frame throughput is calculated below.

$$TP_{\text{Frame}} = \frac{TP_{\text{BW}}}{MUZ_{\text{ST}}} \quad (4-1)$$

### **4.4.2 Stability**

Good QoS should keep stable throughput matched to user desires. We use Coefficient of Variation (CV) to figure the variation to the expected frame number.

#### 4.4.2.1 Coefficient of Variation (CV)

Coefficient of Variation (CV) is a statistical measure of the deviation of a variable from its mean which is calculated as follows:

$$CV = \frac{\sqrt{\text{avg}(x^2) - \text{avg}(x)^2} \text{ (Standard Deviation)}}{\text{expected value}}$$

The coefficient of variation is useful to compare the standard deviation to an investment's expected return.

#### 4.4.2.2 CV to Frame Number

We use CV to the expected frame number to figure out the variation of throughput to expected frame number which is calculated as follows:

$$CV_{FN} = \frac{\sqrt{\text{avg}(FN(TI_j))^2 - \text{avg}(FN(TI_j))^2}}{FN_{\text{expected}}} \quad (4-2)$$

Where  $FN(TI_j)$  is frame number at time instant  $TI_j$  and  $\text{avg}(FN(TI_j))$  is the average frame number.  $FN_{\text{expected}}$  is the expected frame number. The variance is smaller which implies the result is better while the value of CV is closer to zero.

Frame variation shows the steady of QoS but doesn't mean QoS is good. Generally

poor throughput also makes CV small. The best QoS is good and steady performance which means the throughput is high and the value of CV is small.

### **4.4.3 Robustness**

Robustness is figure out by throughput and variation while in the situation of high loading and much congestion occurs. We can see the video with high sensitivity get faster QoS recover. Non-control video has no sense about the network is congestion and lose many packet in the situation.

### **4.4.4 Simulation and analysis**

#### **4.4.4.1 Simulation 1: Video flow without feedback-based QoS control**

In the simulation flows are not controlled with feedback-based QoS control, video flows are all classified into the same service class. Video flows are assign number from 1 to 3 which are Flow1, Flow2 and Flow3.

Video flows start one by one in the time interval 20 seconds and first video flow which is Flow1 starts at time 0. Flow2 starts at time 20 second and Flow3 starts at time 40 as Fig. 4-5 shows.





Fig. 4-5 Frame throughput

While Flow2 starts to transmit data, the throughput of Flow1 decreasing to average 10 frames per second. Throughput of Flow3 decreasing again while Flow3 starts to transmit data. Finally three video flows all receive poor throughput because resource contention in the same service class.

The CV of this simulation is shown in Fig. 4-6. We can see though the throughput of flow1 is higher but the variation is larger because in the begin flow1 use all the bandwidth allocated to this service class and obtain a very good throughput. But while other flows belongs to this service class starting to transmitting data. The bandwidth of flow1 decreasing and obtain bad throughput. The variation of flow2 is small but the throughput of

flow2 is also poor. Because the resource are occupy by flow1 and flow2 is allocated less bandwidth and obtain poor throughput from begin to end. So the variation of flow2 is small but throughput is poor all time.

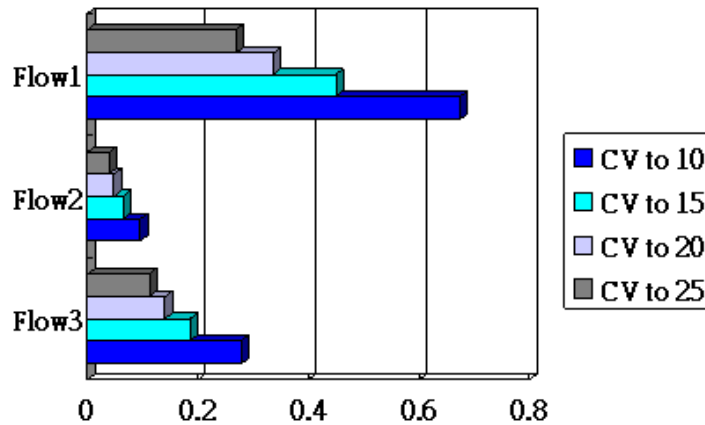


Fig. 4-6 CV to different frame number

#### 4.4.4.2 Simulation 2: Video flows with feedback-based QoS control

This simulation scenario is like the simulation 1 but we enable feedback-based QoS control. The simulation setting is shown in Table 4-1. The min. threshold of access network of network resource is 512kbps which means access network should provide throughput of 512kbps for video service. Min. QoS requirement we defined as 15 frame per second. And the target bandwidth of video service is 128kbps per flow. QoS monitor time interval is 2 second and QoS alert monitor time interval is 6 second. So the QoS alert time threshold is 3 times. Queue weight is 1 for EF queue and flow weight is 1 for video service.

Table 4-1 Simulation Setting of Feedback-Based QoS Control

Field	Parameter	Value	meaning/unit
Min. threshold of access network	$TH_{\min}$	512	kbps
Min. QoS requirement	$QR_{ST}^{\min}$	15	FPS
per flow target bandwidth	$TBW_{ST}$	128	kbps
QoS monitor time interval	$MT_Q$	2	second
QoS alert monitor time interval	$MT_{QA}$	6	second
Queue weight	$w_{QU}$	1	number
flow weight	$w_f$	1	number

So the drop probability is

$$DP_{f_i} = \frac{w_{QU} + w_f}{TBW_{ST} * NT_{f_i}} * \bar{q}_{f_i} = \frac{1+1}{128 * NT_{f_i}} * \bar{q}_{f_i} = \frac{1}{64} \frac{\bar{q}_{f_i}}{NT_{f_i}} \quad (4-3)$$

DP is a function of average QoS  $\bar{q}_{f_i}$  and negotiation time  $NT_{f_i}$ . Because  $NT_{f_i}$  has a threshold and will achieve a value so DP will close to a value. But  $\bar{q}_{f_i}$  will change over time so DP is still dynamically changing and closing to a value.

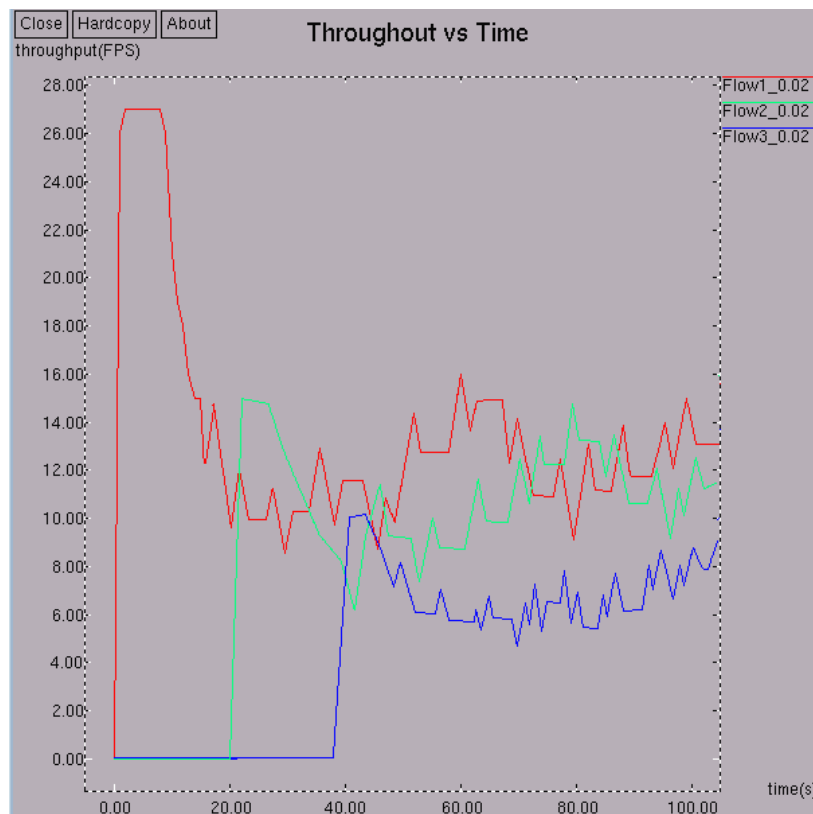


Fig. 4-7 Throughput with Feedback-based QoS control

Fig. 4-7 shows every time while frame throughput lower than 15, the feedback-based QoS control starting to negotiate QoS and resource allocation. Lower QoS has higher priority and the packet drop probability of bad QoS flow is lower. So the QoS won't get worse and worse. While flow2 starts to transmit data at time 20, throughput of flow1 decreases. We set the minimum QoS requirement is 15 frame numbers. While frame number of flow1 is lower than 15, QoS negotiation starts and QoS get recovering after several times. Flow3 has the same QoS sensitivity but because two flows are already in transmission and the bandwidth is insufficient. Though QoS recovery is also shows on flow3 but couldn't permute QoS very much.

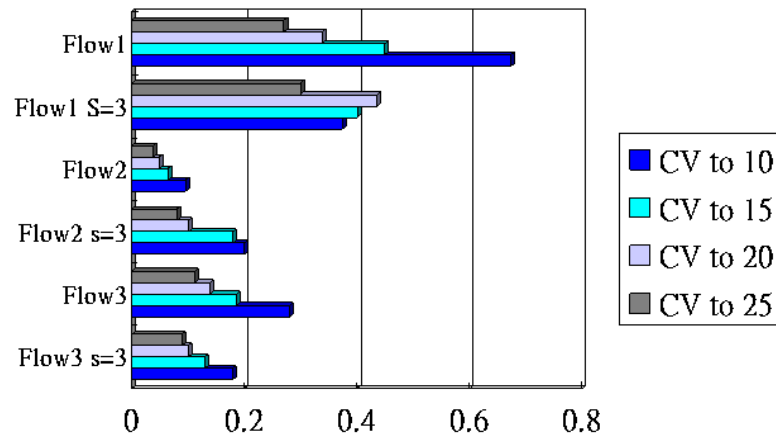


Fig. 4-8 CV to expected frame number with QoS sensitivity = 3

From Fig. 4-8 we can see the variation of flow1 is less because each time the QoS decrease to the lower threshold, QoS control will try to recover the throughput and make the variation of throughput smaller. Variation of flow2 is larger than without feedback-based QoS control but we compare the throughput from Fig. 4-7, we can find the throughput is better than without feedback-based QoS control. Throughput without feedback-based QoS control is generally poor while the throughput with feedback-based QoS control is changing more frequently and causes the value of CV higher. User has more chance to receive good QoS even the good QoS is not always good but better than always receives poor QoS.

#### 4.4.4.3 Simulation 3: Different QoS Sensitivity

We assign each flow with different QoS sensitivity and run the simulation again then the result is shown in Fig. 4-9. Different QoS sensitivity could be represented as different QoS Alert Time threshold, AT. First flow1 has higher QoS sensitivity which AT is 3 times and has faster QoS recovery. Second flow1 with AT is 5 which means flow1 will perform QoS control only when QoS alert times exceed 5 times and implies lower QoS sensitivity.

Uncontrolled Flow 2 runs in background and starts to send data at time 20. Because of resource competition throughput of each flow are decreasing. But it is better than no per-flow control which the throughput is poor from starting to the ending.

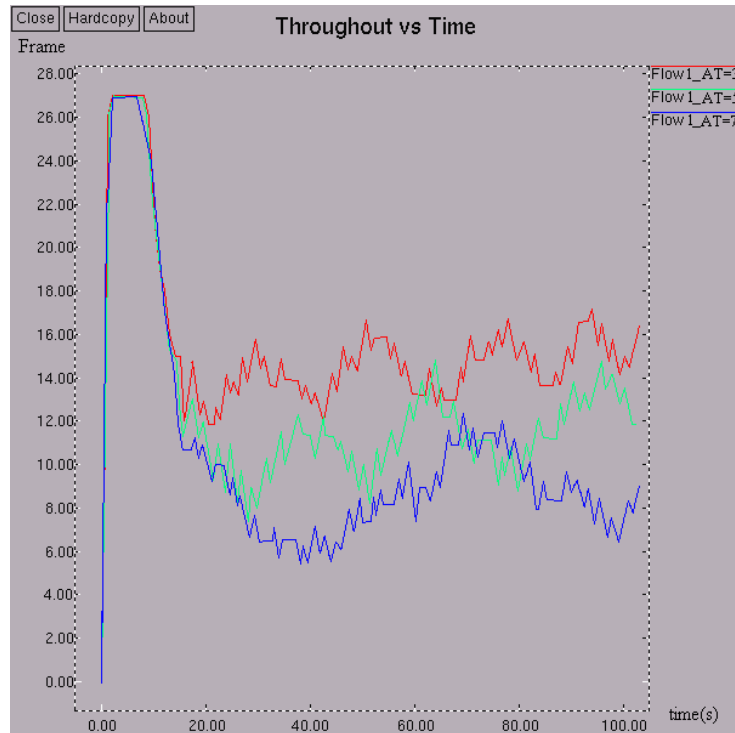


Fig. 4-9 Throughput with different QoS sensitivity

#### 4.4.4.4 Simulation 4: Robustness analysis without QoS feedback control

This simulation uses different error rate which is shown in Fig. 4-10. From the simulation in Fig. 4-5, flow2 starts at time 20 and cause the throughput of flow1 decreasing. Flow3 starts at time 40 and cause throughput decreasing again due to resource competition in the same service class. Throughput of flow1 in error rate = 0.2 is worse than the situation while error rate = 0.02 because many packets are dropped and the user may feel un-comfortable while

frame rate is decreased too much. We can see the throughput decreasing while the error rate increasing. Throughput is keeping poor to the end and doesn't get recover.

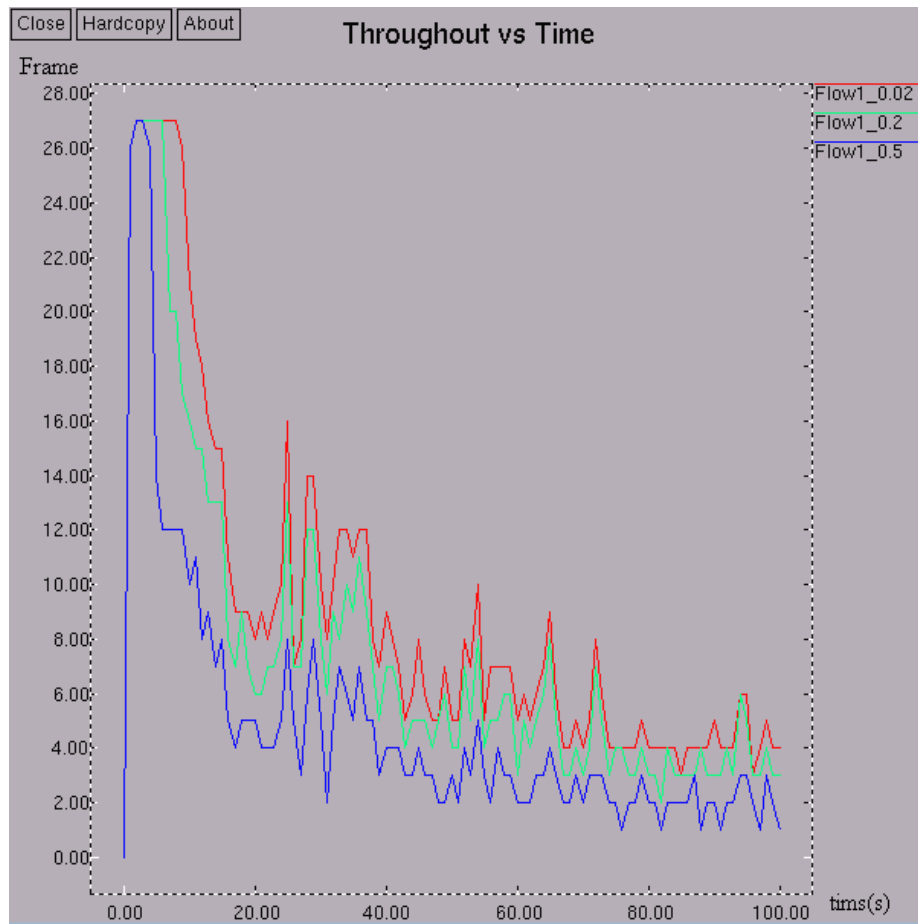


Fig. 4-10 Throughput of flow1 at different error rate

In error rate = 0.2, flow2 and flow3 receive very poor throughput as shown in Fig. 4-11



Fig. 4-11 Throughput in error rate = 0.2 without feedback-based QoS control

In error rate = 0.5, almost half chance the packet will be dropped. Throughputs of all flows are decreasing. But throughput of flow2 increase on the contrary because most packets of flow3 are dropped due to high error rate and flow2 will obtain more bandwidth as shown in Fig. 4-12. This is because lack of per-flow control so flow3 receive unfair treatment even it is in the same service class.



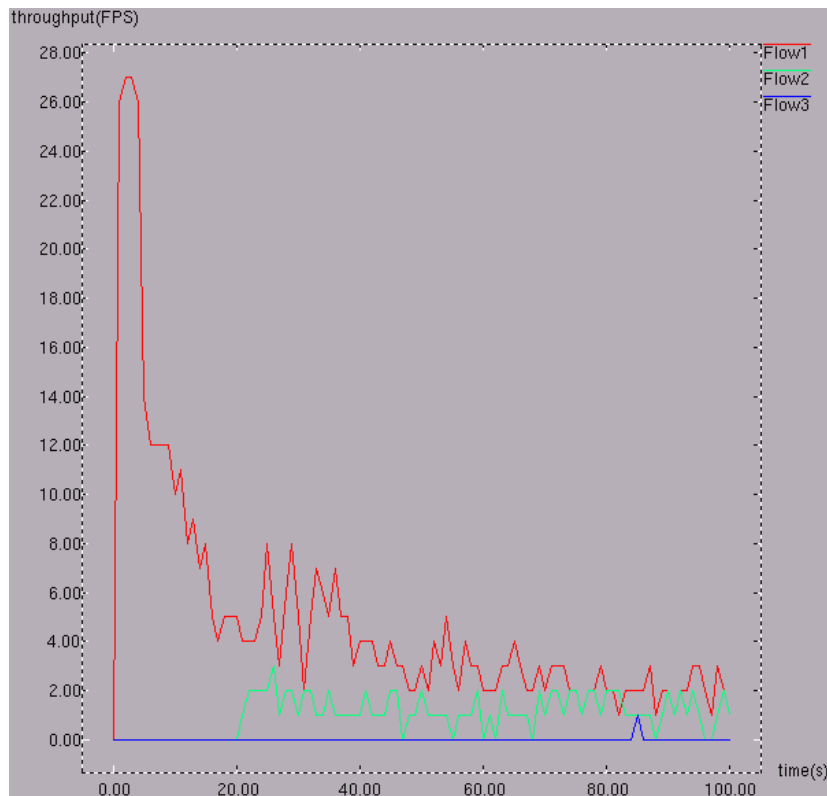


Fig. 4-12 Throughput in error rate = 0.5

The variation to frame number in different error rate is shown in Fig. 4-13. Though the CV of flow1 is steadier with the increasing of error rate but throughput of flow1 is always poor due to high packet dropped rate as shown in Fig. 4-10. Flow3 seems is the most steady but in fact is flow3 always has low throughput. Higher error rate make high packet drop rate in all flows. Original flow3 throughput at error rate 0.02 is low. At error rate = 0.5 the throughput of flow3 is even lower all the time. So Fig. 4-13 shows the QoS of flow3 is steady poor.

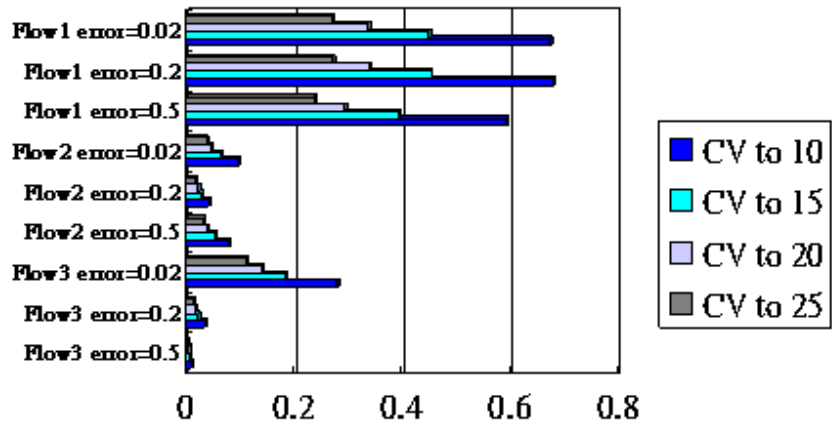


Fig. 4-13 CV to frame number in different error rate

#### 4.4.4.5 Simulation 5: Robustness Analysis with Feedback-Based QoS Control

Fig. 4-14 shows the throughput of flow1 in different error rate. While the error rate increase, the throughput of flow1 decreasing. But by the feedback-based QoS control, we give the higher priority to flow1 when throughput is low and flow2 has chance to recover QoS and won't continue decrease.

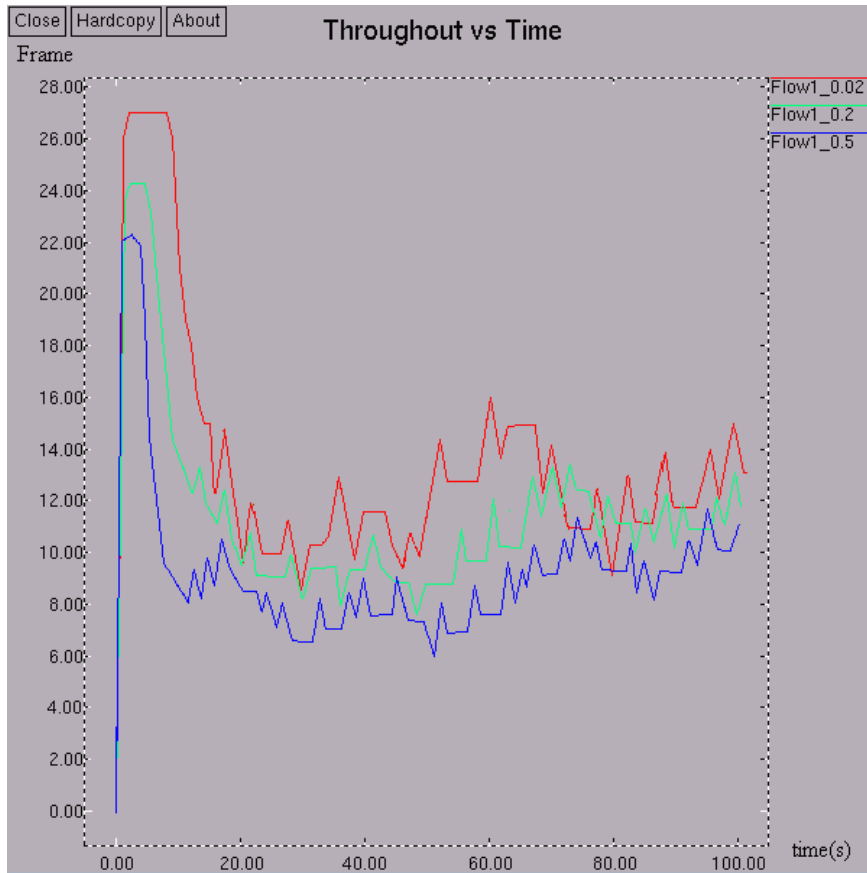


Fig. 4-14 Robustness of flow1 with feedback-based QoS control

Due to the high error rate, throughput of flows are all decrease but will get recover several time later such Fig. 4-15 shows. Throughput of flow3 gets recover several time later and not get starvation like situation without feedback-based QoS control.

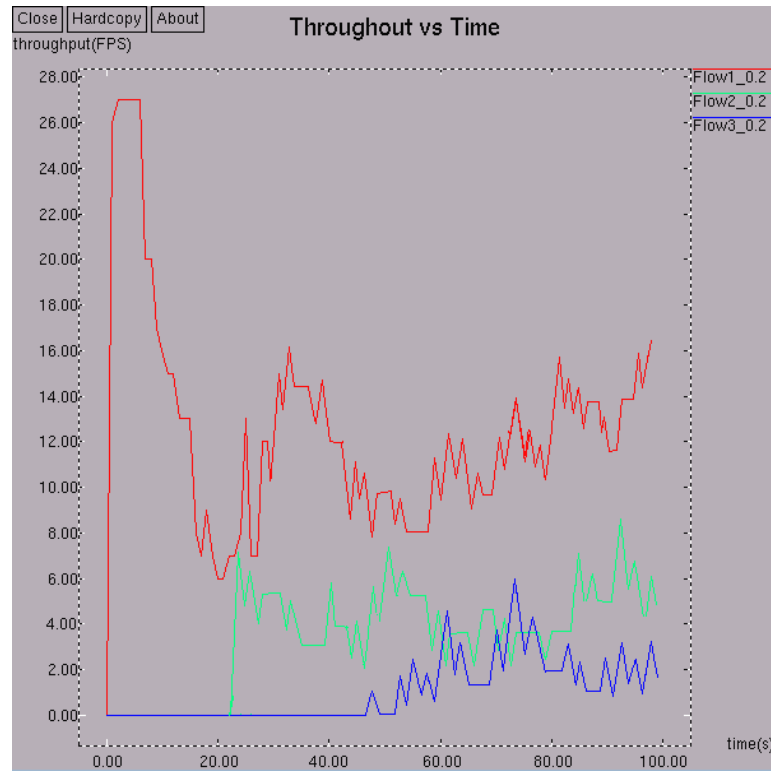


Fig. 4-15 Robustness with feedback-based QoS control while error rate = 0.2

In simulation while error rate = 0.5, more packets are dropped and the changing of throughput is more violent such as Fig. 4-16.

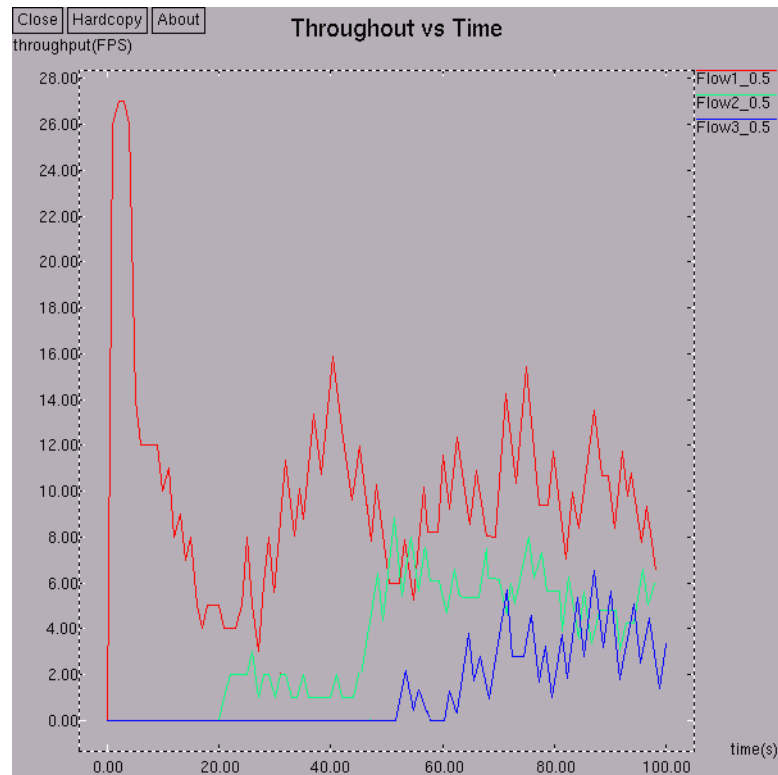


Fig. 4-16 Robustness with feedback-based QoS control while error rate=0.5

Finally we obtain the CV as shown in Fig. 4-17. The variation seems generally more violent than un-control flows. This is because throughput is always be recovered by feedback-based QoS control but the error rate is high so the throughput will decrease again and we promote the throughput again. We obtain better throughput as return. Throughput of flow1 is better and the CV is decrease with the feedback-based QoS control. Flow3 obtain better throughput but its CV is also increase because the insufficient of bandwidth. So the throughput of flow3 changing more frequently.

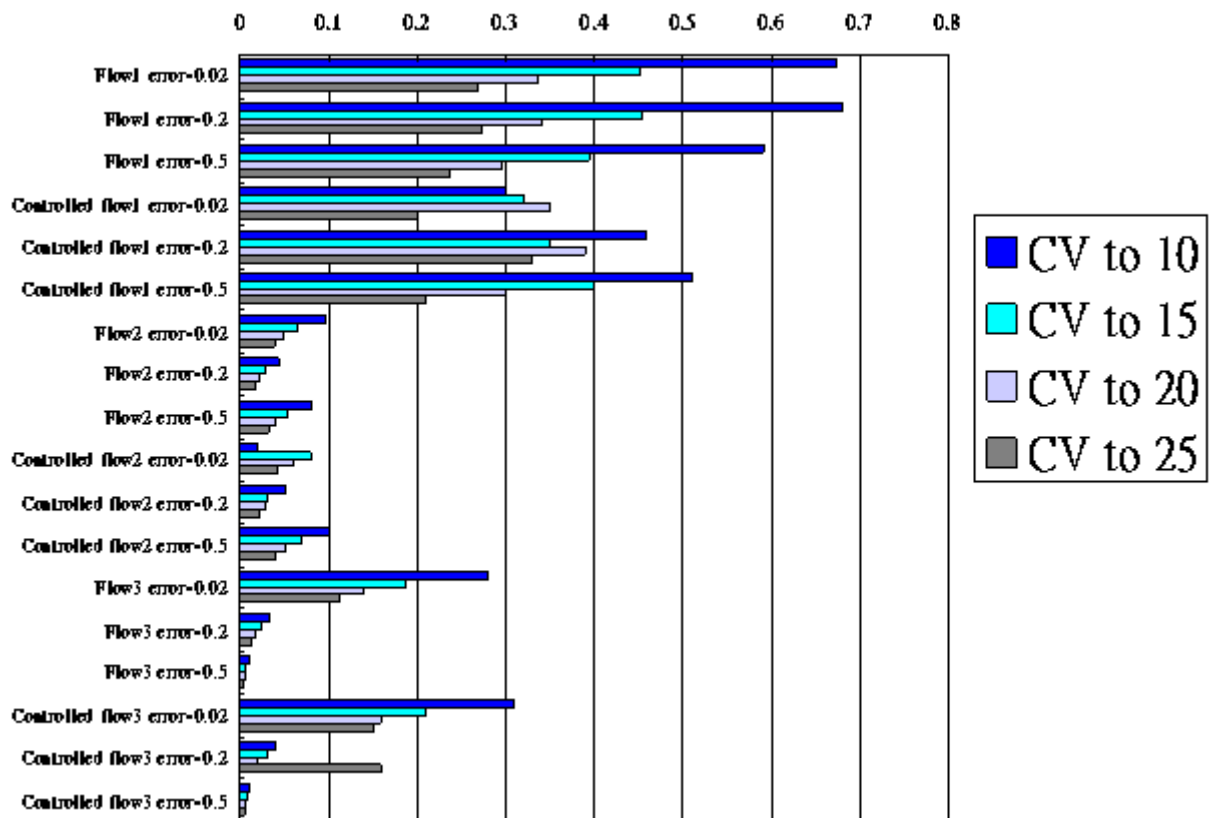


Fig. 4-17 CV of controlled flows and un-controlled flows in different error rate