

End-to-End QoS Management Mechanism for Mobile Network

KwangSik Kim**, Shintaro Uno*, Shobhansingh khadka**, Moo Wan Kim**

* Graduate School of Engineering, Kanazawa Institute of Technology

**Department of Information Systems, Tokyo University of Information Sciences

jkskim62@yahoo.co.jp suno@y4.dion.ne.jp shobhansinghkhadka@hotmail.com mwkim@rsch.tuis.ac.jp

Abstract— Mobile network has been tremendously increased in the number of users, diversity of applications and of interface technologies. Also mobile network need to provide sufficient adaptive End-to-End QoS management mechanism to provide enhanced user satisfaction. In this paper, we propose a new adaptive End-to-End QoS management mechanism based on the concept of utility function which we have borrowed from the field of microeconomics. By this mechanism, the adaptive resource allocation based on user preference in mobile network can be realized. Also, we show the simulation result with some consideration.

Keywords— QoS, End-to-End, Utility Function, Mobility Resource Allocation

I. INTRODUCTION

In the mobile network, QoS is being considered and studied in 3GPP[1]. Also QoS class is defined, and the mapping between QCI (QoS Class Identifier) and DSCP (DiffServ Code Point) is studied. However it is not on End-to-End QoS because it is just focused on the wireless portion of the mobile network. On the other hand, in the area of the fixed network, there are several studies on End-to-End QoS[2], but also those do not include the wireless portion.

Moreover users have been managed equally by the mobile operators, but recently many operators have an intention to differentiate users by their rates. To perform the differentiation, flows are becoming more *adaptive* in the sense that they can adjust their bandwidth according to the amount of resource available. Considering such requirements, a QoS framework has to allocate resource based on users' preferences or payments. In order to achieve this, we borrow the concept of *utility function* from the micro-economics field[3]. We propose a new adaptive End-to-End QoS management mechanism based on utility function. Also, we show the simulation result of the proposal.

The rest of the paper is organized as follows: Chapter 2 presents the basic technologies description. Chapter 3 presents a description of the QoS management mechanisms with the mobility management. Chapter 4 describes the associated signaling mechanism. Chapter 5 presents the simulation result for call setup and handovers with some consideration. Finally conclusion is described in Chapter 6.

II. BASIC TECHNOLOGIES DESCRIPTION

This chapter describes the basic technologies utilized to develop a proposed End-to-End QoS management mechanism.

A. Utility Function

Utility function is a notation that we have borrowed from the field of microeconomics to refer to a user's benefit from all possible resource allocation amounts. This work assumes that the network knows the users utility functions by means of the signaling mechanism. Utility functions quantify the value that an application or users perceives for all possible amount of resource allocated. Utility can also be thought as the price that a user would be willing to pay to obtain a specific amount of resource. Therefore the users can tell the system that for obtaining χ units of resource, they are willing to pay $b\$/\text{second}$.

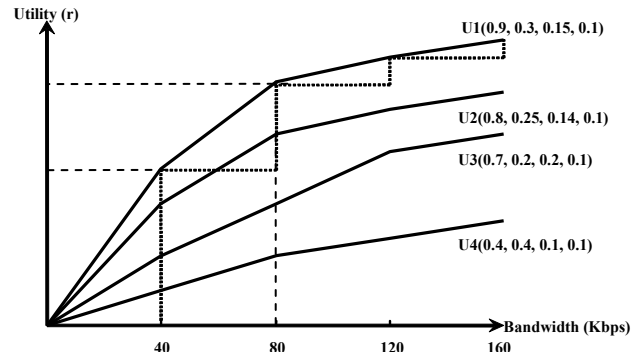


Figure 1. Examples of Utility Function

We let $U(\chi)$ denote the utility functions where χ is the amount of resource and $U(\chi)$ is the utility obtained from that allocation. The user's total utility will depend on several QoS metrics, such as bandwidth, delay, and jitter. As a first step we are going to concentrate on this work on the case that the users only are concerned with the bandwidth allocation they receive. This is a step towards a more general solution that will take into account the other factors as well. Utility functions need to satisfy the following properties:

1. Non-negativity: $U(\chi) \geq 0$ for all $\chi \geq 0$. Obviously the users cannot associate a negative utility with a positive

resource allocation.

2. Non-decreasing nature: $U(\chi)$ has to be a non-decreasing function. Clearly also is the fact that users cannot associate a higher utility with a smaller allocation than with a higher allocation.

Figure 1 shows some examples of utility functions. From there we can observe another property of the utility functions which is concavity. $U(\chi)$ is concave for all $\chi \geq 0$. The meaning of this is that a user attaches a lot more of value to obtaining a unit of resource when it does not have any allocation than when it already has a certain amount allocated.

B. End-to-End QoS

The approach for End-to-End QoS in this paper is based on the following:

- Core Stateless Utility based Rate allocation Framework (SURF) [4] for the wireline part of the network by using utility functions.
- Maximum Segmental Slope (MSS) resource allocation algorithm[5] for the wireless part of the network.

1) SURF: Key attributes of SURF are scalability, which is accomplished using a core-stateless architecture and the support of intra-flow priorities. In the SURF framework, only routers located at the edge of the network maintain per flow state information. SURF treat utility functions by collapsing the entire network into a single unified resource, neglecting any hop count. The SURF framework has two concepts: First, an ingress edge router logically partitions a flow into sub-streams. The sub-streams correspond to different slopes in the utility function of the flow. Sub-streaming is done by appropriately labelling the headers of packets using incremental utilities. Second, a core router has no notion of a flow, and treats packets independent of each other. The forwarding decision at any router is based on the incremental utility labels on the packet headers. Router do not drop a packet with a higher incremental utility label as long as a lower priority packet can instead be dropped. In other words, the core router attempts to provide the same forwarding behaviour of a switch implementing a multi-priority queue, using instead a simple FIFO scheduling mechanism. The problem of dropping packets with lower incremental utility labels before packets with a higher incremental utility can be approximated to the problem of dynamically computing a minimum threshold value that a packet's label must have, in order for a router to forward it. We call this value the threshold utility u_t . This threshold is the minimum incremental utility value that a packet must have in order to be accepted by the router. To determine a new threshold, we have used a heuristic to determine a target value u_{tgt} such that the allocation associated with this u_{tgt} is less than the capacity of the link C . u_{tgt} is based on the average value of u_i of all accepted packets and the maximum valued u_i seen in the last two epochs.

For updating the threshold, this is the pseudo-code and definitions:

```
Current queue length  $\rightarrow cur\_qlen$ 
Maximum Link capacity  $\rightarrow max\_rate$ 
Max Queue size  $\rightarrow q_{lim}$ 
Lower and Upper Q threshold  $\rightarrow q_{lth}, q_{uth}$ 
Burstness Factor  $\rightarrow K_q$ 
Decrement factor based on Qinc  $\rightarrow dec\_p$ 
Heuristic based target ut  $\rightarrow u_{tgt}$ 
Subroutine update _threshold()
```

```
if( $avg\_qlen < q_{th}$ ) or ( $q_{inc} < -K_q$ )
change =  $-dec\_p * u_i$ 
else if ( $avg\_qlen > q_{th}$ ) or ( $q_{inc} > K_q$ )
```

The forwarding algorithm is:

```
enqueue(pkt)
if ( $cur\_qlen < q_{th}$ ) or ( $avg\_qlen \leq q_{lth}$ )
Accept(pkt)
else if ( $u_i \geq u_t$ ) and ( $cur\_qlen + pkt.size \leq q_{lim}$ )
Accept(pkt)
else Drop(pkt)
```

2) MSS Algorithm: Let us consider where we have M users. We let $U_i(r_i)$ denote the utility derived by user's flow i for a bandwidth allocation r_i and C is the total link capacity. User i is allocated r_i^* units of resource that is the i -th component of the solution $r^* = [r_1^*, r_2^*, \dots, r_M^*]$ of the following optimization problem:

$$\begin{aligned} & \text{Max } \sum_{i=1}^M U_i(r_i) \quad (\text{Equation 1}) \\ & [r_1, r_2, \dots, r_M] \\ & \text{subject to } \sum_{i=1}^M r_i \leq C \\ & \chi_i \geq 0 \text{ for } i = 1, 2, \dots, M \end{aligned}$$

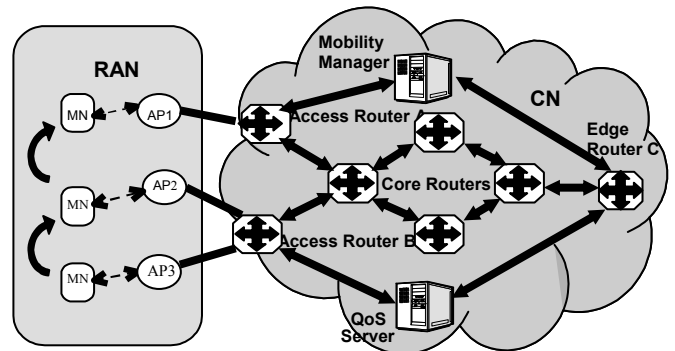


Figure 2. Target Network Architecture

III. QoS MECHANISMS WITH MOBILITY

We assume a network model (Figure 2) which comprises RAN (Radio Access Network) with MNs (Mobile Nodes) and

- 1) QS receives resource allocation request from caller or caller's AR called erAR.
- 2) QS generates this new flow's entry called f_i which has the information of its caller's access router (f_i -erAR), and callee's access router (f_i -eeAR).
- 3) From 2, QS generates a list of the existing flows that need to be re-allocated due to f_i 's joining. It is called f_i -re-flow-list. This list is made of flows that are passing through the caller and callee ARs and that are affected by f_i joining. For example, f_i -erAR-re-flow-list (for example, it includes flow-a, flow-b, flow-c, ...) and f_i -eeAR-re-flow-list (for example, it includes flow-x, flow-y, flow-z, ...).
- 4) By using the utility function based on MSS algorithm, a temporary allocation is made to f_i and the flows in the caller AR list. So, f_i gets f_i -er-alloc, and the flows in f_i -erAR-re-flow-list get $\Sigma f_i\text{-erAR-re-flow-list-temp-alloc} = [\text{temp-alloc-a} + \text{temp-alloc-b} + \text{temp-alloc-c} + \dots]$.
- 5) Same procedures are done for the flow f_i at the callee AR. Flow f_i gets f_i -ee-alloc, and the flows in the list for callee AR (f_i -eeAR-re-flow-list) get $\Sigma f_i\text{-eeAR-re-flow-list-temp-alloc} = [\text{temp-alloc-x} + \text{temp-alloc-y} + \text{temp-alloc-z} + \dots]$.
- 6) QS calculates middle allocation for f_i , this value is called f_i -mid-alloc which is the minimum of [f_i -er-alloc, f_i -ee-alloc].
- 7) Now QS will detect if there is remaining un-utilized bandwidth in the ARs. This is done by re-calculating the bandwidth of the other flows in f_i -erAR-re-flow-list similarly to the process above. The calculation is done for the corresponding callee-AR of every other flow in the list. And again a minimum is selected for each flow. This is the last allocation for the other flows in the caller AR. Resulting in last-alloc-a, last-alloc-b, last-alloc-c, ... Finally, QS sums up these values [$\text{last-alloc-a} + \text{last-alloc-b} + \text{last-alloc-c} + \dots$] which can be expressed in $\Sigma f_i\text{-erAR-re-flow-list-last-alloc}$.
- 8) In the similar manner, QS re-allocates each flow in f_i -eeAR-re-flow-list and results in $\Sigma f_i\text{-eeAR-re-flow-list-last-alloc}$ which is the sum of [$\text{last-alloc-x} + \text{last-alloc-y} + \text{last-alloc-z} + \dots$].
- 9) Then the remaining bandwidth in caller side AR is $f_i\text{-erAR-remain-bw} = (f_i\text{-erAR-total-bw} - f_i\text{-mid-alloc} - \Sigma f_i\text{-erAR-re-flow-list-last-alloc})$.
- 10) Similar, the remaining bandwidth in the callee side AR is calculated in the way as above and results in $f_i\text{-eeAR-remain-bw} = (f_i\text{-eeAR-total-bw} - f_i\text{-mid-alloc} - \Sigma f_i\text{-eeAR-re-flow-list-last-alloc})$.
- 11) Then the remaining bandwidth for caller AR and callee AR for the flow f_i is $\text{remain-bw} = \min [f_i\text{-erAR-remain-bw}, f_i\text{-eeAR-remain-bw}]$.

12) At last, as the result, the final allocation for f_i is $= [f_i - \text{mid-alloc} + \text{remain-bw}]$.

B. Call Setup and HO Signalling

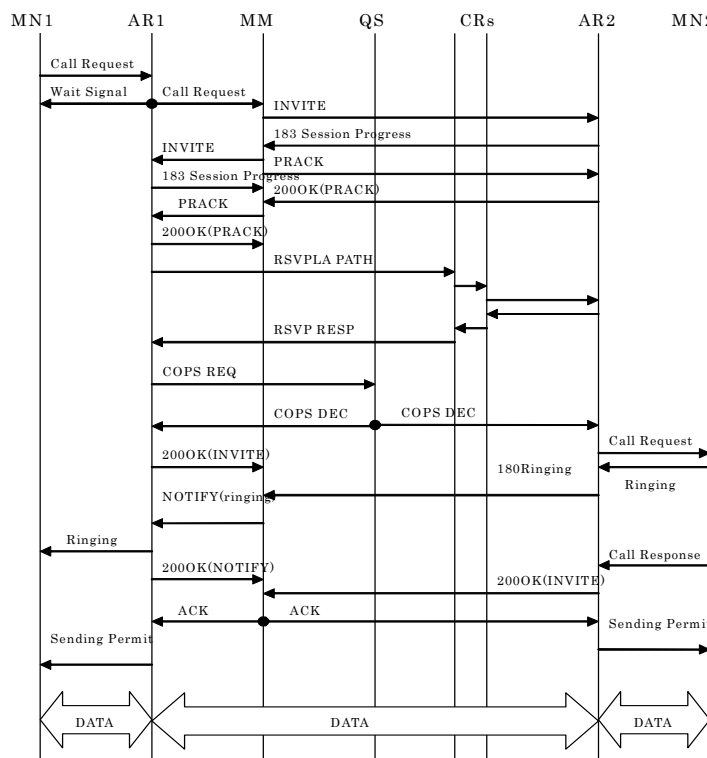


Figure 3. Call Set-up Signalling

In order to investigate the interactions, we have decided to utilize SIP since it can be adapted easily to our architectural model which allows us to keep the mobile terminal involvement to a minimum. Also we have used proprietary version of RSVP called RSVP-PLA which is a light End-to-End signalling protocol for signalling the QoS processes. The communication between access routers (AR) and QoS Server (QS) is realized using COPS(Common Open Policy Service) protocol. Figure 3 shows the call setup signalling and Figure 4 shows the HO(Hand Over) signalling.

IV. EVALUATION

A. Simulation Results

We have used the NS2 simulator with the topology shown in Figure 5. We have implemented call setup and hand over signaling with the proposed QoS management algorithm. Each access router has 4 units of bandwidth to administrate (1 unit = 40 kbps). In this simulation, the utility threshold (U_{th}) of core routers is set to be zero where all packets are passed through. MN1, 2, 3 and 4 under AR1 have session with MN5, 9, 13, and 17 respectively. Each node request 4 units of bandwidth and

each AR has 4 units of bandwidth. The data flows are called flow1, 2, 3 and 4. Flow1, 2, 3 and 4 have utility functions U1, U2, U3 and U4 (shown in Table 1) respectively. In the non handover call simulation, flow 1 enters the network at time 10s. As it is the only flow in the system, it is allocated the entire AR1's bandwidth. At time 20s, flow2 enters the network. From MSS algorithm, flow1 and flow2 should be allocated by 2 units and this is shown in Figure 6. Then at time 30s, with flow3's join, the allocation becomes $f1:f2:f3 = 2:1:1$. Finally after f4 enters, each flow has same bandwidth at receiver and this is the allocation theoretically.

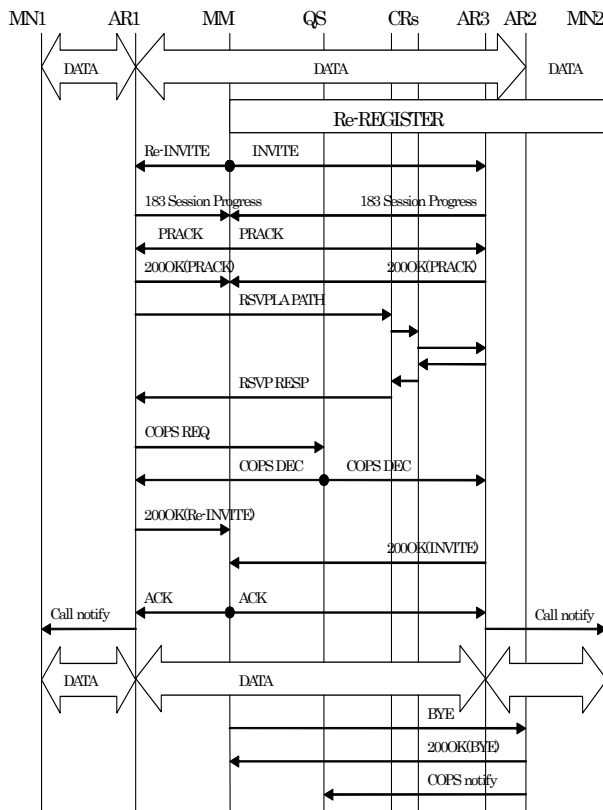


Figure 4. Hand Over Call Signalling

Figure 7 shows the result when a handover takes place. Utility value of Flow-big, between MN2 and MN9, Flow-small between MN6 and MN10, and Flow-h (handover), between MN8 and MN14 are listed in Table-2. Flow-h starts its call at 30 sec and MN8 starts to move toward to new-AR (AR3) at 40 sec. Before Flow-h enters to AR3, there are two flows in AR3, our algorithm will therefore allocate 2 units of bandwidth to each flow (Flow-big and Flow-small). And when Flow-h enters the area of AR3, the allocation becomes $Flow-big:Flow-h:Flow-small = 2 : 1 : 1$ units.

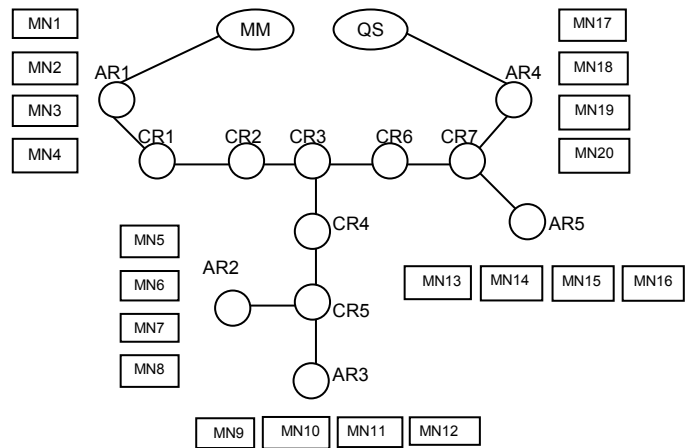


Figure 5. Simulated Network

B. Consideration

Through the simulation we have confirmed the following items.

- 1) We have confirmed that the allocation in the simulation is equal to the allocation calculated theoretically.
- 2) Users can obtain the more satisfied service. Normally when the bandwidth is fully utilized, the new service request is rejected and in case of handover, the flow is suspended. The proposed method can solve these issues in some degree.
- 3) By this method it is confirmed that the bandwidth can be utilized at maximum. Moreover if service price is linked to the value of the utility function, the user will pay more payment so that the operators can obtain more revenue and profit.

V. CONCLUSION

We have proposed a new adaptive End-to-End QoS management mechanism. As the next step, we can expand this method on the following items.

- 1) In the case that the incremental value of user's utility function is not high and a new flow, where it's incremental value of utility function is higher, is generated, it may be happened that the existing service will be suspended. However, the service can be continued if we modify the method that existing services should be guaranteed by some minimum bandwidth.
- 2) About the assignment of the utility function, we assume two cases; (a) mobile operators can define utility functions and save the information in QoS server, (b) mobile users can select a utility function for a flow and send this information to QoS server by signalling.

REFERENCES

- [1] 3GPP TS 29.212," 3rd Generation Partnership Project; Technical Specification Group Core Network and Terminals;Policy and Charging Control over Gx reference point(Relase 8)", 2008-09.
- [2] Y.Kaneda, "Policy-based End-to-End QoS Guarantee Using On-Path Signaling for Both QoS Requests and Feedback", ICOIN 2008, I-1, Jan.2008.

- [3] H.R. Varian, Microeconomic Analysis, W. W. Norton and Co., 1992.. [5] O. Gonzalez, F.Gotoh, and S. Gilbert, "Maximum Network Operator Revenue Resource Allocation, 2002 IEICE Society Conference, Sept. 2002.
- [4] N. Venkitaraman and J. Misore "A Core Stateless Utility based Rate Allocation Framework", Proceeding of INCP2001.

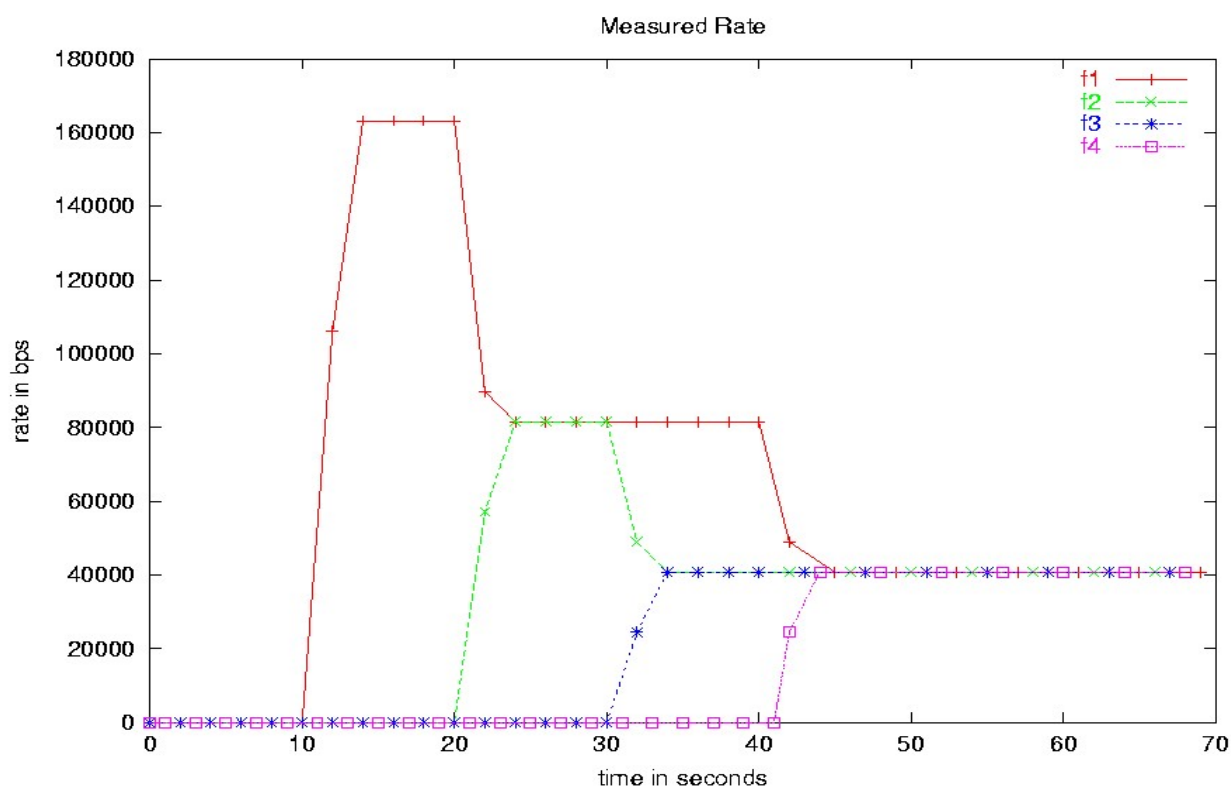


Figure 6. Simple Flows Evaluation : Non-Hand Over

Table-1. Utility Values for Flows

Flow	1u	2u	3u	4u
Flow1	0.9	0.3	0.15	0.1
Flow2	0.8	0.25	0.14	0.1
Flow3	0.7	0.2	0.2	0.1
Flow4	0.4	0.4	0.1	0.1

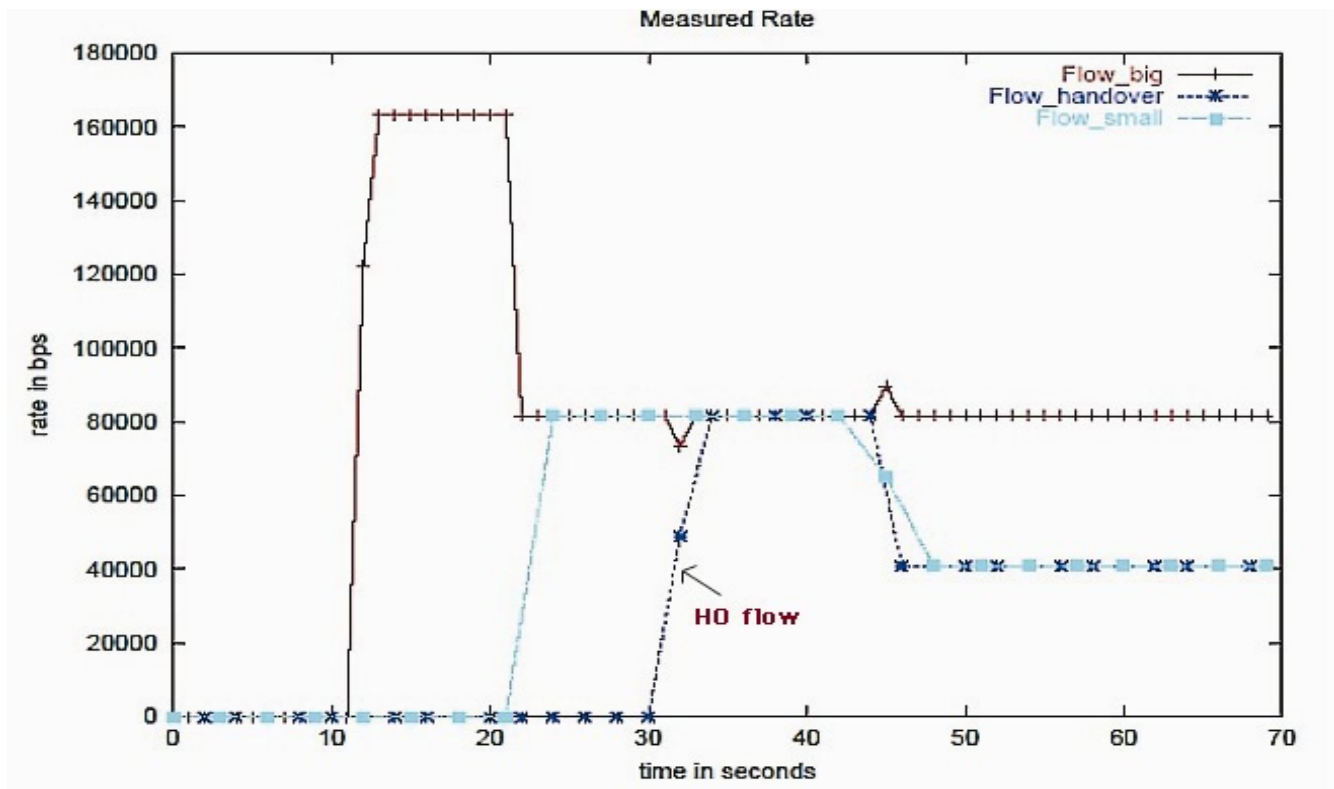


Figure 7. Simple Flows Evaluation including Hand Over

Table-2. Utility Values for Flows

Flow	1u	2u	3u	4u
Flow-big	0.9	0.3	0.15	0.1
Flow-h	0.8	0.25	0.14	0.1
Flow-small	0.7	0.2	0.2	0.1