CHAPTER 6

REAL TIME PERFORMANCE ANALYSIS OF THE CDA-EPON ALGORITHM

The potential of any proposal depends on its commercial viability. To realize the CDA-EPON proposal in commercial switches, it has to be implemented in processor and checked for its compatibility in hardware architecture. Also, real time service assurance of LTE necessitates that the algorithm be tested in a processor before implementing it for commercial practice. For the above, the proposal is implemented and its performance is measured in a LTE switch using Intel’s IXP 2400 Network Processor. Performance is determined by the number of clock cycles consumed for the execution of the algorithm and hence, the number of clock cycles for executing the proposal is measured and compared with that of the conventional scheme. This chapter explains the various sections of the CDA-EPON algorithm along with their implementation in the processor.

6.1 IMPLEMENTATION

There is no functional difference in the operation of the ONUs between the conventional scheme and the CDA algorithm. Still, to record the performance of the system the ONU processing system is implemented in the network processor and the machine cycles required to execute the algorithm are estimated. Three microengines are required to construct an ONU hardware. The first microengine receives the packets, the second microengine performs the bandwidth estimation and the third microengine is used to transmit the packets. Description of the OLT’s functionality follows the ONU’s construction.
6.1.1  Microengine 0:0 – Reception

ME 0:0 is used to receive the packets for the ONU. The first microengine receives all the incoming packets which are then stored in memory for further processing. Figure 6.1 details the packet reception functionality in an ONU.

![Flowchart diagram](Figure 6.1 Packet Reception Functionality Algorithm)
6.1.2 Microengine 0:1 – ONU Processing

ME 0:1 reflects the functionality of the ONU. The ONU estimates the bandwidth requirement for the next cycle and generates the REPORT message filling the Report bitmap and Queue#n Report fields of the REPORT message. Then, the ONU sends the request message to the OLT. Figure 6.2 details the processes done in an ONU after receiving packets from the end users.

![Flowchart](image)

**Figure 6.2 ONU Processing Algorithm**

6.1.3 Microengine 0:2 – Transmission

ME 0:2 is used by ONU for packet transmission. Figure 6.3 details the sequence used for packet transmission
Figure 6.3 Packet Transmission Algorithm

Implementation of the conventional and proposed OLT models using network processor is now discussed. The implementation requires four microengines. ME 0:0 receives the packets from the ONU. ME 0:1 classifies the packet. ME 0:2 run the conventional and proposed OLT algorithms and forward the packets to the ME 0:3 for transmission. The receiving process carried out by ME 0:0 is similar to the ONU model as explained in the previous section and the transmission process carried out ME 0:3 is similar to the process handled by ME 0:2 in the ONU model.

6.1.4 Microengine 0:1 – Classification

In order to improve the CDA algorithm, the QoS mapping between EPON 802.1P priority queues and the LTE connections is done. The second
microengine ME 0:1 reflects the functionality of the QoS guarantor. EPON supports eight priority levels with the QoS for the service represented by the numbered priority queue in the 802.1P nomenclature. QoS in LTE is determined by QCI, Allocation and Retention Policy. The 9 QCI values define eight characteristics for IP packets ranging from VOIP call to email and chat. The mapping between the EPON priority queue and LTE flows provides equivalent QoS levels at both ends as detailed in Table 5.1. Figure 6.4 details the packet classification algorithm.

![QoS Mapping Algorithm](image)

**Figure 6.4 QoS Mapping Algorithm**

6.1.5 **Microengine 0:2 – OLT processing**

ME 0:2 runs the OLT processing functionality and the performance of the OLT processing is checked in both the conventional and the proposed scheme.

6.1.5.1 **Conventional OLT Processing**

ME 0:2 depict the functionality of the conventional OLT. OLT receives the REPORT message from all active ONUs. For each REPORT message extracts
the Report bitmap and Queue\#n Report fields from each REPORT message and calculates the ONU’s requested bandwidth. OLT calculates the available and allotted bandwidth to each ONU and then sends a GATE message to each ONU with these details. The steps are detailed in Figure 6.5.

![Figure 6.5 Conventional OLT Processing Algorithm](image)

After evaluating the performance of the conventional algorithm, the CDA algorithm is then implemented.

### 6.1.5.2 CDA-EPON OLT Processing

ME 0:2 reflect the functionality of the proposed OLT. The OLT is designed to be configured with the SLA values of the ONUs. OLT on receiving a request from the ONU, grants immediately either its SLA or requested bandwidth based on whether the request is higher or lower than the SLA value without waiting for request messages from rest of the active ONUs. For ONUs with request higher than its SLA value, the first set of GATE message do not generate a REPORT message in response by resetting the ‘Force Report’ flag in the GATE message. Whereas, for ONUs with request lower than or equal
to its SLA value, a single GATE message is sent and the allocation meets its request. This GATE message generates a REPORT message in response to continue with the transmission process for the next cycle.

If more bandwidth is available, the OLT then sends another GATE message to all the high bandwidth requesting ONUs with its excess request of bandwidth in a round-robin fashion. The second GATE message unlike the first informs the ONU to respond back with a REPORT message to continue with the transmission process for the next cycle. The steps involved in the proposed OLT model are detailed in Figure 6.6.

![Figure 6.6 CDA-EPON OLT Processing Algorithm](image)

After implementing the algorithm, the performance of the same in the microengine is evaluated.
6.2 RESULTS AND DISCUSSION

The conventional and proposed architecture are implemented in IXP2400 network processor which is configured to work as a LTE switch and studied for their performance.

Since there is no difference in configuration for the ONU system between the conventional and proposed CDA schemes, the performance study of the microengines in the ONU configuration is studied for better understanding of the system. Reception implementation is done in ME 0:0 for both ONU and OLT configuration. Initially, ME 0:0 is free but the execution of the microengine increases as the packets are received and transferred to the next microengine for further operation as detailed in Figure 6.7.

![Machine clock cycles for ONU algorithm](image)

**Figure 6.7 Machine clock cycles for ONU algorithm**

When there are no more packets to receive, this microengine becomes completely free. ME 0:1 implements the functionality of the ONU. Speed of execution is initially low but gets increased as packets are received and the REPORT messages are generated containing the bandwidth request details for each request. When there are no more packets to receive, the consumption
becomes gradually lower. ME 0:2 transmit the packets from the ONU to the OLT. Initially, the speed of execution for ME 0:2 is very low because packets reach ME 0:2 only after they are received and processed by ME 0:0 and ME 0:1. The speed of execution increases after this initial slowness when more packets reach ME0:2 scheduled for transmission after ME 0:1 finishes processing of the packet.

Performance of the OLT configuration is the critical factor in deciding the realization of the algorithm. Performance study of the microengines in the OLT configuration is depicted in Figure 6.8. Since ME 0:0 is similar in both ONU and OLT model and ME 0:3 in OLT is similar to the process handled by ME 0:2 in the ONU model, the focus is on the functionalities of ME 0:1 and ME 0:2. Packet classification functionality is implemented in ME 0:1 which classifies the packets based on the Type of Service. ME 0:1 is initially free but when the packets are transferred from the ME 0:0 for classification it increases gradually. Conventional and proposed OLT functionality are implemented in ME 0:2. Machine clock cycle consumption by ME 0:1 in the conventional mode gradually increases when the packets are processed after reception and when packets from all ONUs are received it becomes high, since the bandwidth allocation calculation is done only after receiving from all the ONUs. In the proposed mode, the utilization of the machine clock cycle is higher than the conventional mode in the start since the OLT now sends a GATE message to every request immediately. But the utilization remains stable and lower than the conventional mode after receiving request from all ONUs since the OLT now does the second round of bandwidth allocation which involves lesser processing than conventional mode. The observation from the clock cycles indicates that the processor requires around 17000 cycles to execute the conventional OLT algorithm at the peak and 14000 cycles to execute the proposed OLT algorithm at the peak. The peak utilization of the machine cycles in the CDA algorithm is
lower than the conventional mode and hence becomes a proof for the suitability of the algorithm for commercial deployment in a LTE switch.

Figure 6.8 Machine clock cycles for Conventional (top) and CDA-EPON (bottom) OLT algorithm