University of Essex

School of Computer Science & Electronic Engineering

Hardware Accelerated Impairment Aware Control Plane

Yixuan Qin[1], Yvan Pointurier[2], Eduard Escalona[1],
Siamak Azodolmolky[2], Marianna Angelou[2], Ioannis Tomkos[2],
Kostas Ramantas[3], Kyriakos Vlachos[3], Reza Nejabati[1] and Dimitra Simeonidou[1]

[1]*School of CSEE, University of Essex, Wivenhoe Park, Colchester, CO4 3SQ, UK*
{yqin, eescal, rnejab, dsimeo}@essex.ac.uk
[2]*AIT P.O. Box 68, 0.8km Markopoulo Ave., 19002 Peania, Greece*
{yvan, sazo, mang, itom}@ait.edu.gr
[3]*CEI Department, University of Patras, 26500 Rio, Patras, Greece*
{ramantas,kvlachos}@ceid.upatras.gr

Supported by:

University of Essex

Talk Overview

School of Computer Science & Electronic Engineering

- Background
- Motivations of the hardware acceleration of IA-CP
- Hardware accelerated QoT estimation tool architecture
- Different Solutions for Integrating an Embedded processor with the FPGA Logic
- Experiment
- Results
- Conclusions

Background

- A dynamic impairment aware (IA) optical network is desired.
  - Physical layer impairments inject extra constraints on QoS of transmission services
- The dynamicity is provided by an extended IA control plane (CP) which enables real time candidate lightpath computation.
  - A proprietary dynamic network planning and operation tool (NPOT) has been developed.
  - It integrates advanced physical layer models with routing and wavelength assignment (RWA) algorithms to provide a mechanism for physical layer IA transport network provisioning.
- In CP, upon a connection request, the optical signal quality of transmission (QoT) is estimated in order to make the decision of accepting or rejecting candidate lightpaths' establishment.

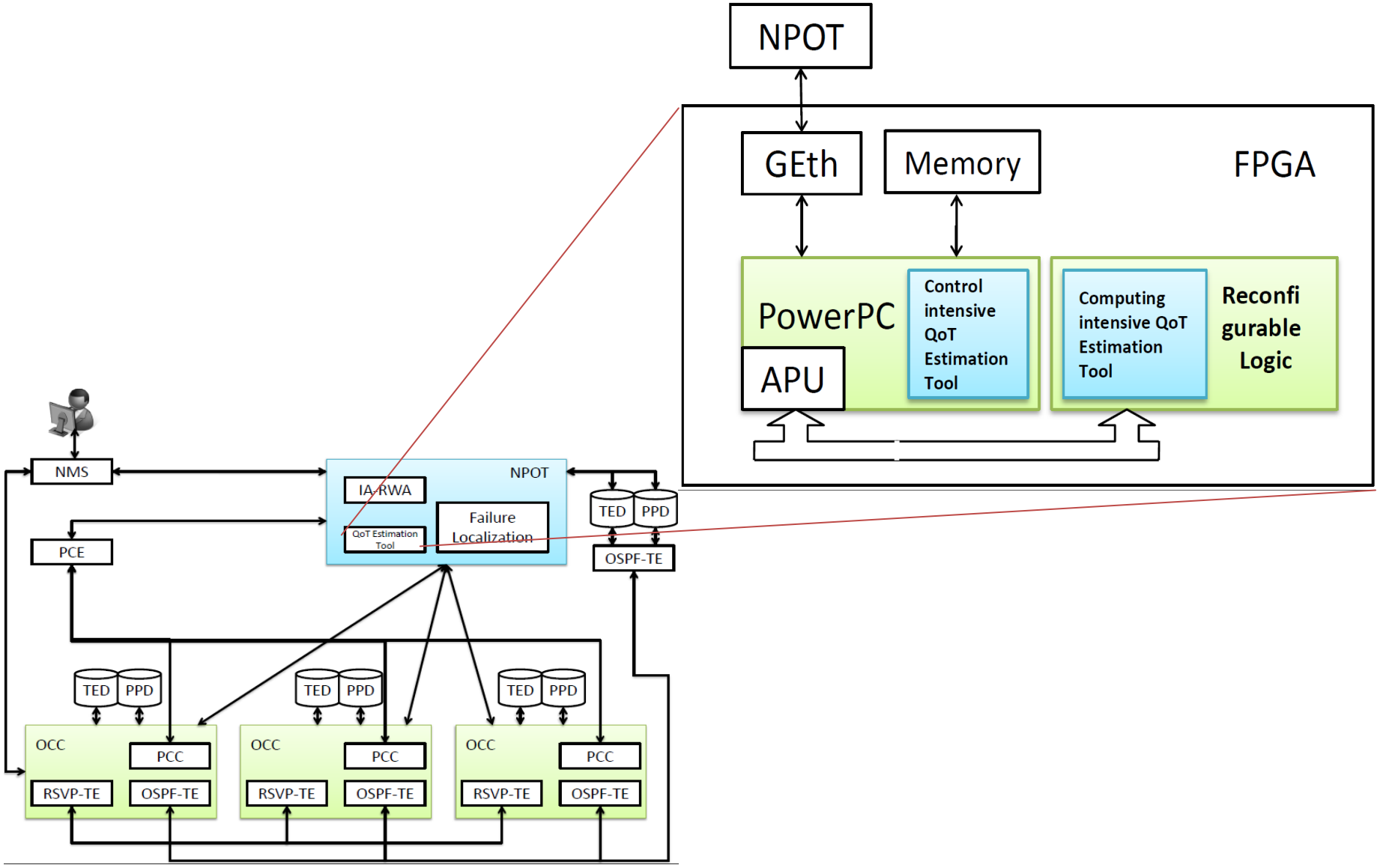
Motivation of Hardware Acceleration in IA-CP

- The complex algorithmic calculations performed for QoT estimation is unsuitable for real time dynamic lighpath provisioning.
- Currently pure software based solution costs 10 to 1000 seconds.
  - Depends on the network load and RWA algorithms deployed.
- Hardware-accelerated QoT estimation tool sharply decreases the lightpath assessment time and accelerates lightpath provisioning.
  - Enable timing critical applications which need fast response and low delay.
  - Facilitates finer granularity of resource sharing and better resource utilization when the signalling protocol is able to set-up and tear-down connections more often.

University of Essex

Hardware Accelerated QoT Estimation Tool Architecture

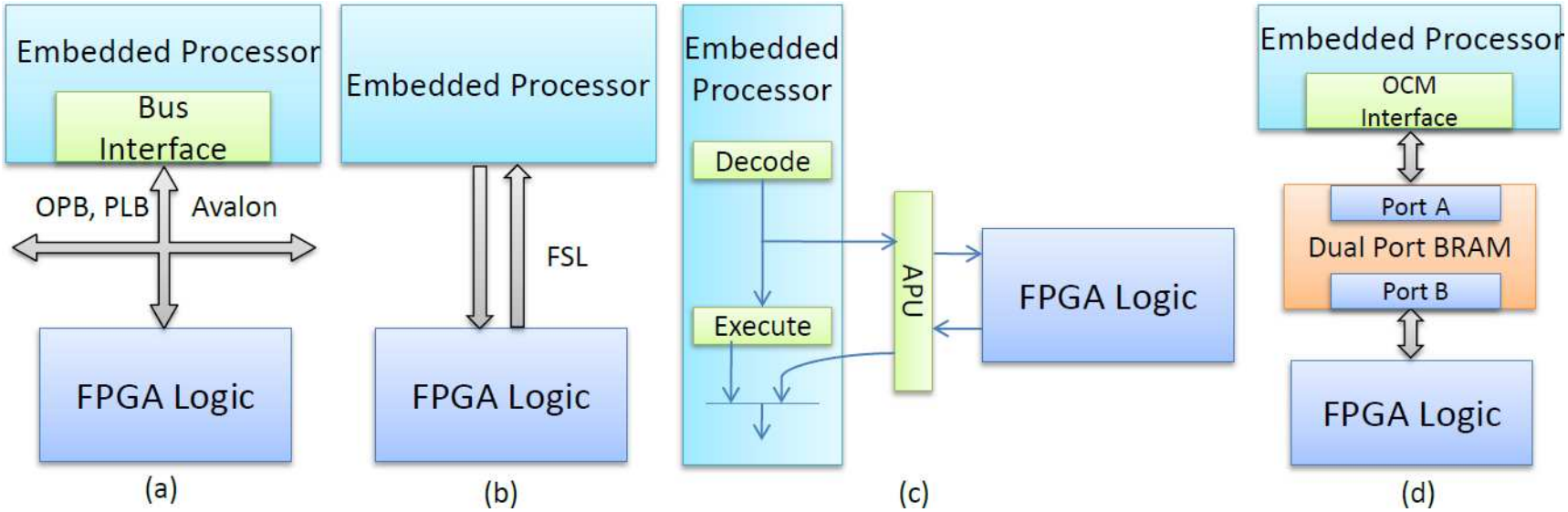
School of Computer Science & Electronic Engineering



Hardware Accelerated QoT Estimation Tool Architecture (cont.)

- An extended GMPLS CP is deployed.
  - OSFP-TE is extended to disseminate impairment information.
  - Standard RSVP-TE is deployed to setup the lightpath.
- New path request needs to be verified based on QoT estimation.
  - Hardware acceleration happens in QoT estimation.
  - If the path reservation fails or the QoT estimation tool gives negative Q factor for alternative paths, the Network Management System is notified accordingly.
- Both embedded processor (PowerPC) and FPGA logics are used for the QoT estimation.
  - Linked by auxiliary processor unit (APU)
  - Control intensive and computation intensive operations are separated in PowerPC and FPGA logics.
- Communicate with the CP by network

Different Solutions for Integrating an Embedded Processor with the FPGA Logic



- Embedded processor bus connected model (a)
  - A single data transaction can require many processor cycles and need bus arbitration.
  - Slow communication.
- I/O based model (b)
  - A dedicated point to point communication channel to exchange data between the FPGA logic and embedded processor without any arbitration overhead.

- I/O based model (cont.)
  - The reduced control complexity enables lower latency and higher rate data movement .
  - One channel only supports one direction data movement.
- Extended instruction set model (c)
  - PowerPC instruction set is extended to execute more functions natively.
  - Instructions are simultaneously presented to the embedded processor decoder and the APU controller.
  - If the embedded processor recognizes the instruction, it is executed, otherwise, the APU acknowledges the instruction then sends to the FPGA logic to execute it.
- Shared memory model (d)
  - An on chip dual port block RAM (DPBRAM) is used between the FPGA logic and embedded processor.

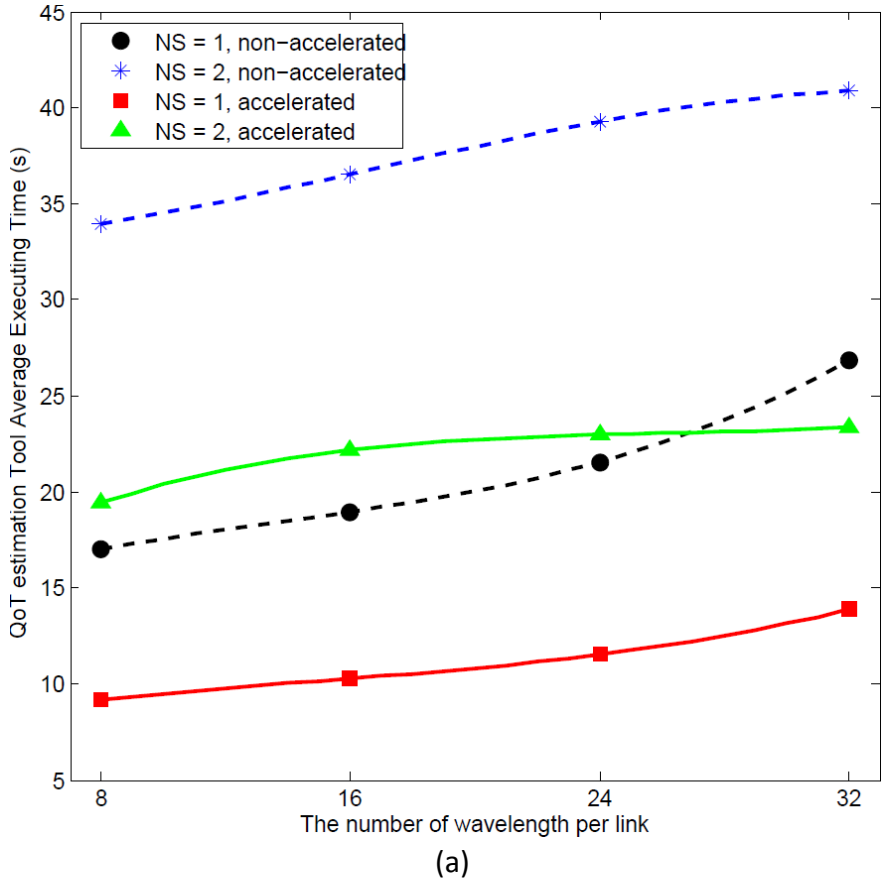
Experiment

Table : Comparison between Non-Accelerated and Accelerated QoT Estimation Tool

Non-accelerated QoT estimation tool		Accelerated QoT estimation tool	
CPU	Memory	"CPU"	Memory
3.2GHz Intel Quad Core Extreme	4GB DDR3	100 MHz Xilinx Virtex 4 + 300 MHz IBM PowerPC 405	1 GB DDR2

- Experiment scenarios are listed in the table above
- Pure software based QoT estimation tool is executed on a modern fast PC
- Accelerated QoT estimation tool is executed on FPGA and PowerPC hardware platform
- Reference network topology is provided by Deutsche Telekom (DT)
- QoT estimation tool performance is evaluated in terms of the number of available wavelengths per link, varying from 8 to 32 with a step of 8, and for a varying number of lightpaths established in the network.

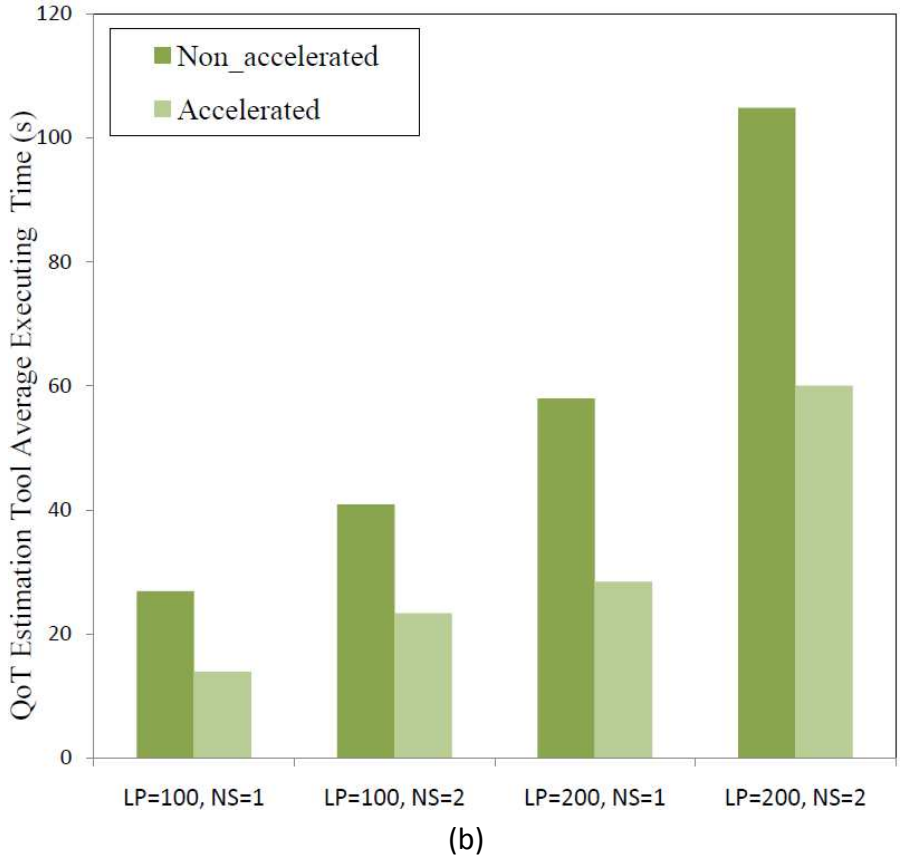
Results



(a)

- The QoT estimation tool performance for 100 existing lightpaths
- The average executing time is against the number of wavelength per link for the two different network sizes (NS={1,2}).
- For both network sizes, the accelerated QoT estimation tool execution time (solid line) is nearly halved than the non-accelerated (dotted line).

Results



- The QoT estimation tool performance for combinations of two numbers of lightpath (LP = {100, 200}) and two network sizes (NS = {1, 2}) with fixed number of wavelengths per link (32).
- The accelerated QoT estimation tool outperforms the non-accelerated one by almost 100% for each combination of LP and N.
- It scales smoothly and mitigates the impact of increasing load.

Conclusions

- A HW/SW co-design accelerated QoT estimation tool is demonstrated.
- A considerable performance improvement is achieved.
- Better performance can be achieved by deploying shared memory model and more FPGA logics.

University of Essex

School of Computer Science & Electronic Engineering

Thank you

Yixuan Qin
yqin@essex.ac.uk