Machine Learning Assisted QoT Estimation and Planning with Low Margins

Kostas Christodoulopoulos¹, I. Sartzetakis², P. Soumplis², and E. Varvarigos²

¹Nokia Bell Labs, Stuttgart, Germany
²School of Electrical and Computer Engineering, National Technical University of Athens, Greece
Outline

• Motivation
• QoT estimation and margins
• Accurate QoT estimation using ML
  – Accuracy evaluation
• Provisioning with accurate QoT estimation
  – Incremental multiperiod planning
  – Case study
Motivation

✖ Optical networks are planned to be operated statically
  • Provision lightpaths, by estimating QoT at EOL (10 years)
    • Ageing, increased interference, inaccuracies – EOL Margins
✖ High margins lead to overprovisioning / high CAPEX & OPEX
  • Other overprovisioning factors: inefficient failure handling, accounting for future traffic demands
✖ Static network operation & overprovisioning will not work as traffic becomes more volatile – 5G, Edge Cloud
✔ Call for increased efficiency, lower overprovisioning, reduced margins
Evolution of margins over time

- Define acceptable performance after accounting for fast penalties (~1 dB), operator’s margin (~1 dB), uncertainties (~2 dB), unallocated (transponder-reach mismatch)
- Traditional approach: target to be acceptable at EoL
- Reduced margins: target to be acceptable at intermediate periods (while we also reduce uncertainties)
QoT estimation

• QoT estimation is used by a planning or online algorithm
• QoT estimator (Qtool): a Physical Layer Model (PLM)
  • Modelling inaccuracy
    – Inputs (databases)
      • Physical layer parameters: spans, fibers, amplifier parameters, etc.
      – Uncertainty: measuring errors, outdated measurements
    • Connections parameters: route, spectrum, baudrate, modulation, etc.
      – Output: lightpaths’ QoT (SNR, BER, etc.)
• Design margin: account for inaccuracies (model and input parameters)
• System margin: account for QoT deterioration until EOL (equipment ageing, increased interference, fiber cut reparation, etc.)
QoT estimation with Machine Learning

• Assume established connections / brownfield deployment
• Use monitoring and machine learning (ML)
  – Understand actual network conditions
    • Reduce design margin: improve accuracy of physical layer parameters
    • Reduce system margins: no need to target EOL
• Monitoring data
  – Power monitors
  – Rx (e.g. dispersion, SNR, BER), focus on SNR (used by developed models)
    but a Rx gives aggregated information → account for routes & spectrum
    • Lightpaths sharing common links share information
    • Lightpaths relative spectrum position includes information
    • Target per link and wavelength/interference QoT parameters
1st method: Machine Learning - Physical Layer Model (ML-PLM)

- Physical Layer Model (PLM)
  - Inputs
    - Connection parameters $P$
      routes, spectrum, TRx configuration (baudrate, modulation, etc.)
    - Physical layer parameters $b$
      spans, fiber attenuation, dispersion, nonlinear coefficients, amplifiers parameters, ROADM parameters
  - Output: lightpaths’ QoT (SNR) estimates $Q(b, P)$
- Parameters $b$ not accurately known, yield QoT estimation error
- Train PLM using monitoring $Y(P) \rightarrow$ ML-PLM

1st method: Machine Learning - Physical Layer Model (ML-PLM)

- ML training
  - Initialize physical layer parameters $b^0$ datasheets or (outdated) measurements
  - Fit (iteratively) parameters $b^i$ to min the error $Y(P) - Q(b^i, P)$
    - Fitting algo depends on the PLM model, and if we know $\partial Q/\partial b_j$
    - If $Q$ is nonlinear to some $b_j \rightarrow$ nonlinear fitting
  - Obtain fitted physical layer parameters $b^*$
- For a new connection, $w (P' = \{P U w\})$, use learned parameters $b^*$ to estimate $Q(b^*, P')$, when deciding how to establish it
- Implementation: Qtool = GN model, fitting = nonlinear regression (Levenberg-Marquardt nonlinear least squares algorithm)
2nd method: Machine Learning Model (ML-M)

• Without a Qtool
• ML-M: Machine Learning Model, functioning as Qtool
  – Input
    • Features $X = f(P)$, $X$: matrix with one row per lightpath
      For a lightpath its row - features represent QoT-related parameters
    • ML Model coefficients $\theta$
      depend on the particular ML model (linear regression, NN, SVM, etc.)
  – Output: lightpaths’ QoT estimates $\tilde{Y}(X, \theta)$
• Use monitoring $Y(P)$ to train the model and obtain $\theta^*$ that yield low estimation error
2nd method: Machine Learning Model (ML-M)

• Choosing appropriate features
  – Literature: end-to-end features (e.g. path length, #hops, #EDFAs)
    • Cannot cope with network heterogeneity
    → Per link features
  • Features matrix with link features: $X = [B\ A\ S\ W]_{P \times (1+3|L|)}$
    Bias+3 sets of link features for the 3 major impairment classes
    – $A$: ASE, $S$: SCI, $X$: XCI
    e.g. $S_{p,i} = PSD_p^3$ (power spectral density) of lightpath $p$ if it uses link $i$, else=0
    SCI noise contribution depends (linearly) on lightpath’s PSD$^3$
2nd method: Machine Learning Model (ML-M)

- Features designed so that the impairment noise contribution depends (close to) linearly on them
  - $X_{p,j}$ the $j$th impairment/link feature of lightpath $p$, the noise contribution of impairment on that link is approximated well with $n_j(X_{p,j}, \theta) = X_{p,j} \cdot \theta_j$
- Noise additivity assumption
  - The total noise of lightpath $p$ is $\sum_j n_j(X_{p,j}, \theta)$
- ML-Model: linear regression $\hat{Y}(X, \theta) = X \cdot \theta$, gradient decent
- Also tried NN and SVM, and obtained similar results
- NN, SVM would account better for nonlinear dependency of features and other more complicated features
ML QoT Estimation – accuracy evaluation

- Ground truth (create monitoring data and obtain estimation error): GN model
- 12 nodes Deutsche Telecom topology
- Physical layer parameters
  - Span parameters (length, fiber coefficients): ±0%, 10%, 20% around default values
  - Actual parameters assumed unknown → uncertainty: 0%, 10% and 20%
- Traffic
  - 4 traffic loads 100, 200, 300, 400 connections, 80% training, 20% testing
  - Uniform source-destination, uniform baudrate: 32, 43, 56 Gbaud
  - 500 instances per load
- ML-PLM, ML-M
Mean Square Error

- Both ML-PLM and ML-M achieve excellent MSE
- ML-PLM is better (note: the ground truth and the trained PLM are the same)
- ML-PLM’s error is higher for higher uncertainty
  - Starts from default / average parameters and learns
- ML-M’s accuracy is not sensitive to uncertainty since it does not assume any default parameters
Maximum Overestimation

- Similar findings for max overestimation
- Design margin = max overestimation
  - \( \text{SNR}_{\text{over}} = \text{SNR}_{\text{est}} - \text{SNR}_{\text{real}} \), for threshold \( \text{SNR}_{\text{thr}} \), it is safe if \( \text{SNR}_{\text{real}} > \text{SNR}_{\text{thr}} \) ➔ \( \text{SNR}_{\text{est}} - \text{SNR}_{\text{over}} > \text{SNR}_{\text{thr}} \)
- ML-PLM design margin: 0.05 dB, ML-M design margin: 0.2 dB for 200 lightpaths

(untrained) PLM
- max overestimation
  - 0 dB for 0% uncertainty
  - 0.9 dB for 10% uncertainty
  - 2 dB for 20% uncertainty
Quantifying savings of accurate QoT estimation

• Multi-period/incremental planning (period=several months to years)
  1. Traditional: provision with high margins to reach end-of-life (EOL) and account for inaccuracies
     • System margin: equipment ageing, interference increases, maintenance operations
     • Design margin: QoT estimation model inaccuracy
  2. With reduced margins / accurate QoT estimation
     • New connections: provision them with enough margins to reach next (or some targeted) period
     • Established connections: check their QoT and reconfigure or add regenerators to reach next (or targeted) period
Incremental planning algorithm

- **Input at the start of period** $\tau_i$
  - Traffic described by the remaining and new demands
  - TRx installed at previous periods / established lightpaths (up to $\tau_i$)
  - Equipment e.g. capabilities of Flex- (or fixed-) rate TRx
- **Interface with QoT estimator**
- **Objective**
  - Serve traffic
    - Cater for remaining lightpaths that run out of margins, serve new demands
  - Minimize added cost
- **Algorithm:** heuristic, examines previous and new connections 1-by-1 [1][2]

---

Incremental Planning and ML-PLM

**Greenfield**
- Initial demands
- Provisioning algorithm (placement of equipment and configurations, routing and wavelength assignment, etc.)
- Available equipment
- Qtool Physical Layer Model (PLM)
- Physical layer parameters
- Reduced (to reach next period)
- Design margin
  - PLM model inaccuracy
  - Physical layer parameters uncertainty

**Brownfield / Incremental planning**
- New demands
- Provisioning algorithm (placement of equipment and configurations, routing and wavelength assignment, etc.)
- Monitoring
- ML train
- Accuracy improved
- Reduced (to reach next period)
- Reduced

---

2019
Case study – Topology, traffic, TRx

- DT network topology
- 11 periods, 1 period ≈ 1 year, incremental planning every 1 year
- Initial traffic: 200 initial connections, uniform src-dst, uniform [100-200] Gbps
- Traffic increases by 20% per period
- 2 types of TRx: TRx1 available at period 0 ($\tau_0$), TRx2 available at period 5 ($\tau_5$)
  - TRx1: 32 Gbaud, DP-QPSK to DP-16QAM, $\text{SNR}_{\text{thr}}=0.01\text{dB}$, cost= 1, at period 0 ($\tau_0$)
  - TRx2: 64 Gbaud, DP-QPSK to DP-32QAM, $\text{SNR}_{\text{thr}}=0.01\text{dB}$, cost= 1, at period 5 ($\tau_5$)
- Cost reduction 10% per period

<table>
<thead>
<tr>
<th>Data Rate (Gbps)</th>
<th>Baud Rate (Gbaud)</th>
<th>Mod Format</th>
<th>BOL ageing &amp; BOL interf. &amp; High design</th>
<th>EOL ageing &amp; EOL interf. &amp; Low design</th>
<th>EOL ageing &amp; EOL interf. &amp; High design</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>32</td>
<td>DP-QPSK</td>
<td>4720</td>
<td>3600</td>
<td>2280</td>
</tr>
<tr>
<td>150</td>
<td>32</td>
<td>DP-8QAM</td>
<td>2080</td>
<td>1600</td>
<td>1280</td>
</tr>
<tr>
<td>200</td>
<td>32</td>
<td>DP-16QAM</td>
<td>1040</td>
<td>800</td>
<td>560</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Data Rate (Gbps)</th>
<th>Baud Rate (Gbaud)</th>
<th>Mod Format</th>
<th>BOL ageing &amp; BOL interf. &amp; High design</th>
<th>EOL ageing &amp; EOL interf. &amp; Low design</th>
<th>EOL ageing &amp; EOL interf. &amp; High design</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>64</td>
<td>DP-QPSK</td>
<td>4160</td>
<td>2800</td>
<td>2240</td>
</tr>
<tr>
<td>300</td>
<td>64</td>
<td>DP-8QAM</td>
<td>2720</td>
<td>1840</td>
<td>1440</td>
</tr>
<tr>
<td>400</td>
<td>64</td>
<td>DP-16QAM</td>
<td>1840</td>
<td>1280</td>
<td>960</td>
</tr>
<tr>
<td>500</td>
<td>64</td>
<td>DP-16QAM</td>
<td>1280</td>
<td>880</td>
<td>640</td>
</tr>
</tbody>
</table>
Case study – Physical layer evolution & margins

• Initialize with heterogeneous spans and uncertainty
  – Attenuation, dispersion, nonlinear coefficients uniformly around default values ±10%
  – Unknown to QoT estimator, requires ~1 dB margin
• Ageing: increase per period according to table
• 10 instances (load & physical layer), average results
• Planning with high margins
  – EOL system margin (EOL ageing & full load interference), BOL design margin (2 dB)
• Planning with reduced margins - ML-M (or ML-PLM) and incremental planning algorithm
  • Initial period: design = 2 dB, system = 2 periods ageing & current interference
  • Each period, train ML-M and obtain new design (=1dB+0.2dB+training max overest.), system = 2 periods ageing & current interference

<table>
<thead>
<tr>
<th>Physical layer parameters evolution</th>
<th>Increase per period</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transponder margin (dB)</td>
<td>0.05</td>
</tr>
<tr>
<td>Attenuation (dB/km)</td>
<td>0.0015</td>
</tr>
<tr>
<td>EDFA noise figure (dB)</td>
<td>0.1</td>
</tr>
<tr>
<td>OXC loss (dB)</td>
<td>0.3</td>
</tr>
<tr>
<td>Interference</td>
<td>According to load</td>
</tr>
</tbody>
</table>
Case study - basic comparison

- The reduction of the system margin postpones the purchase of equipment
- The reduction of the design margin (ML – learning) avoids the purchase, after the first period
- ~20% savings at the end of 10 periods
  - Could be even higher if we optimize the power
Conclusions

- Traditionally lightpaths are provisioned using a QoT estimator (PLM) and EOL margins
- Developed 2 ML QoT estimators (with a PLM and without)
  - Use monitoring data, understand physical conditions and ageing, reduce system margins
  - Very good accuracy, design margin reduced to 0.2 dB with few 100s lightpaths
- Quantified savings of accurate QoT estimation
  - Integrated ML-M with incremental planning algorithm
  - Multi-period planning case study
    ~20% savings with accurate QoT estimation/planning with reduced margins as opposed to EOL margins
Copyright and confidentiality

The contents of this document are proprietary and confidential property of Nokia. This document is provided subject to confidentiality obligations of the applicable agreement(s).

This document is intended for use of Nokia’s customers and collaborators only for the purpose for which this document is submitted by Nokia. No part of this document may be reproduced or made available to the public or to any third party in any form or means without the prior written permission of Nokia. This document is to be used by properly trained professional personnel. Any use of the contents in this document is limited strictly to the use(s) specifically created in the applicable agreement(s) under which the document is submitted. The user of this document may voluntarily provide suggestions, comments or other feedback to Nokia in respect of the contents of this document (“Feedback”).

Such Feedback may be used in Nokia products and related specifications or other documentation. Accordingly, if the user of this document gives Nokia Feedback on the contents of this document, Nokia may freely use, disclose, reproduce, license, distribute and otherwise commercialize the feedback in any Nokia product, technology, service, specification or other documentation.

Nokia operates a policy of ongoing development. Nokia reserves the right to make changes and improvements to any of the products and/or services described in this document or withdraw this document at any time without prior notice.

The contents of this document are provided "as is". Except as required by applicable law, no warranties of any kind, either express or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose, are made in relation to the accuracy, reliability or contents of this document. NOKIA SHALL NOT BE RESPONSIBLE IN ANY EVENT FOR ERRORS IN THIS DOCUMENT or for any loss of data or income or any special, incidental, consequential, indirect or direct damages howsoever caused, that might arise from the use of this document or any contents of this document.

This document and the product(s) it describes are protected by copyright according to the applicable laws.

Nokia is a registered trademark of Nokia Corporation. Other product and company names mentioned herein may be trademarks or trade names of their respective owners.
QoT estimation – state of the art

<table>
<thead>
<tr>
<th>Paper</th>
<th>Qtool</th>
<th>ML method</th>
<th>Features</th>
<th>Simulations / Experiments</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>[1]</td>
<td>without Qtool</td>
<td>Regression</td>
<td>Interference aware links (1/SNR per link, different links according to adjacent lightpaths)</td>
<td>Simulations, GN model as ‘ground truth’</td>
<td>Homogeneous’ spans</td>
</tr>
<tr>
<td>[2]</td>
<td>without Qtool</td>
<td>Classification</td>
<td>- #hops, path length, longest link, modulation format, network traffic volume, …</td>
<td>Simulations, GN model with worst case interference as ‘ground truth’</td>
<td>Worst case interference ‘Homogeneous’ spans</td>
</tr>
<tr>
<td>[3]</td>
<td>with Qtool (inhouse, GN model)</td>
<td>Regression</td>
<td>Input parameters of Qtool (power at nodes, X_K)</td>
<td>Simulations, GN model and inhouse Qtool as ‘ground truth’, worst case interference</td>
<td>Worst case interference</td>
</tr>
<tr>
<td>[4]</td>
<td>with Qtool (inhouse close to GN)</td>
<td>Calculate the derivatives, similar to gradient decent</td>
<td>length-dependent loss and nonlinear intensity (IRI) noise based on the GN model [15], computing in each fiber span the SPM-like and XPM-like noise contributions due to nonlinear effects in fiber based on frequency spacing between optical signals, their signal power levels, and the fiber nonlinear coefficient [2]. In this work we used only a single-mode fiber (SMF).</td>
<td>Experiments, 6 nodes !</td>
<td></td>
</tr>
<tr>
<td>[5]</td>
<td>With Qtool (GN model)</td>
<td>Regression</td>
<td>Maximum likelihood / extended Kalman filter</td>
<td>Not clearly described, for sure SNR and wavelength</td>
<td>Experiments (6 nodes, VDAs to emulate different link lengths), and simulations (to evaluate benefits)</td>
</tr>
<tr>
<td>[6]</td>
<td>Without Qtool</td>
<td>Ridge regression</td>
<td>26 input features for each wavelength or data sample. These features include data rate, fiber type, frequency, length of path, margin, measured fiber loss, measurement date, number of amplifiers in the path, number of passes through ROADMs, optical return loss (ORL), end-of-path optical signal-to-noise ratio (OSNR), and polarization mode dispersion (PMD). We estimate the OSNR of each fiber section based on launch power, amplifier noise, and measured span loss. We then combine these fiber section estimates to estimate the end-to-end path OSNR. In cases where regeneration is needed, we treat the sections between regeneration points as separate wavelengths.</td>
<td>Experiments</td>
<td></td>
</tr>
</tbody>
</table>