Machine Learning Assisted QoT Estimation and Planning with Low Margins

Kostas Christodoulopoulos[1], I. Sartzetakis[2], P. Soumplis[2], and E. Varvarigos[2]

[1]Nokia Bell Labs, Stuttgart, Germany

[2] School 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

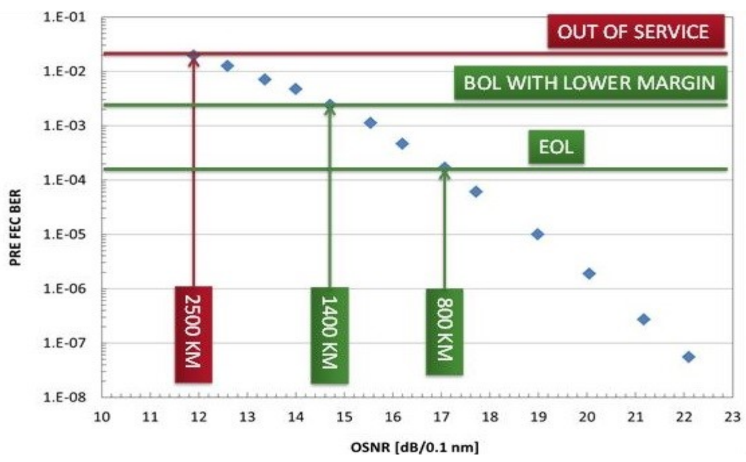
- Provision lightpaths, by estimating QoT at EOL (10 years)
  - Ageing, increased interference, inaccuracies – EOL Margins

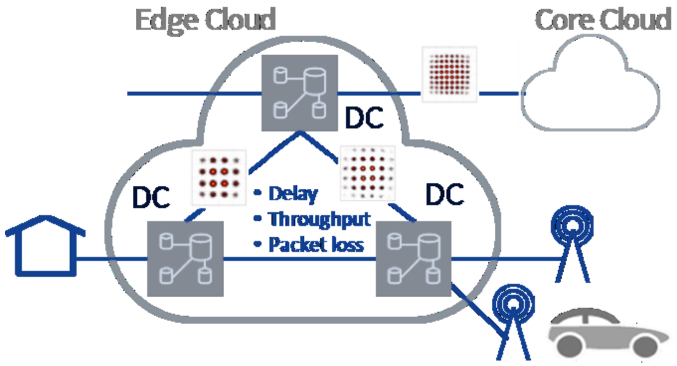
☒ High margins lead to overprovisioning / high CAPEX &OPEX

- 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

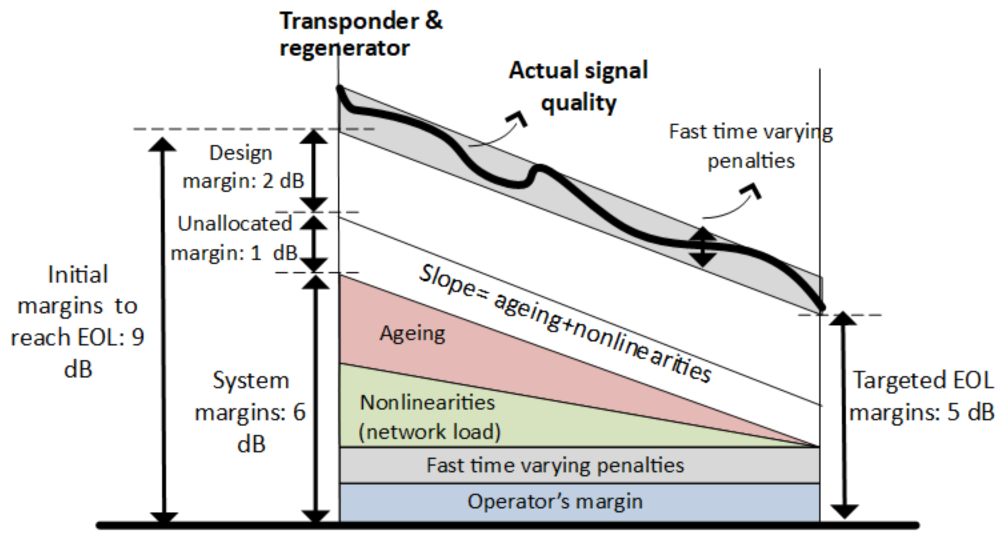
☑ Call for increased efficiency, lower overprovisioning, reduced margins

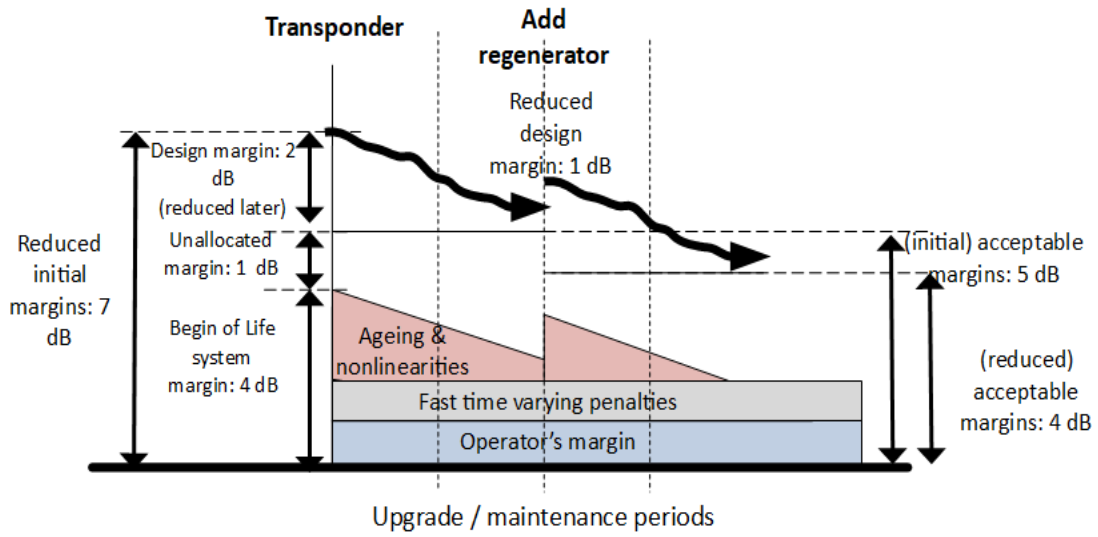




NOKIA Bell Labs

Evolution of margins over time



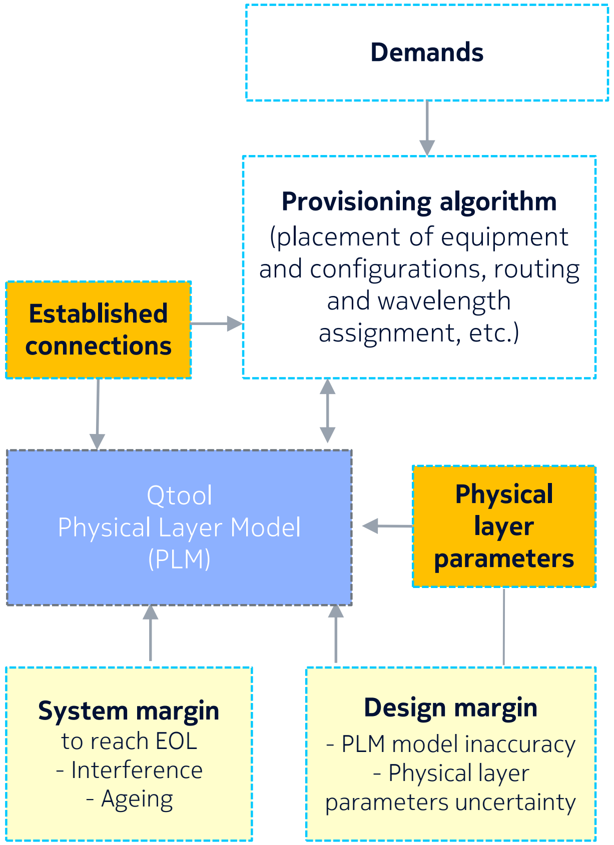


- Define acceptable performance after accounting for fast penalties (~1 dB), operator's margin (~1 dB), uncertainties (~2 dB), unallocated (transponder-reach mismatch)
- Traditional approach: target to be acceptable at EoL
- Reduced margins: target to be acceptable at intermediate periods (while we also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

NOKIA Bell Labs

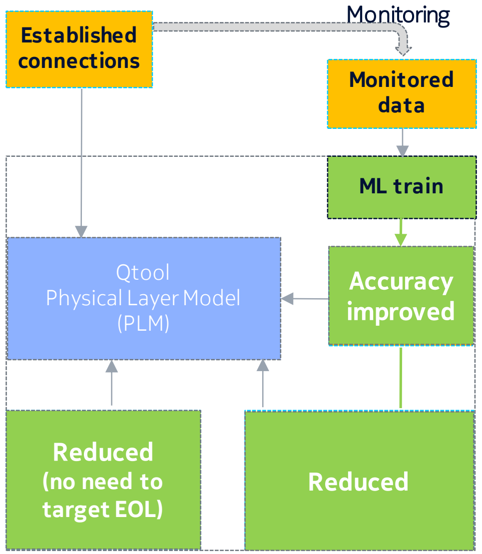
QoT estimation

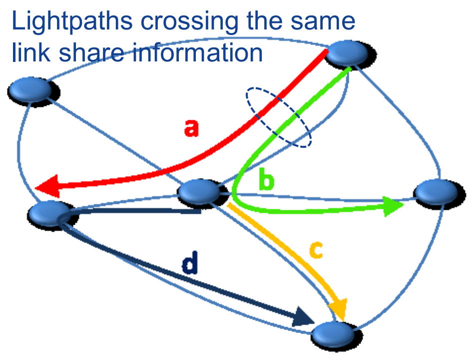
- QoT estimation is used by a planning or online algorithm
- QoT estimator (Qtool): a Physical Layer Model (PLM)
    - Modelling inaccuracy
  - Inputs (databases)
    - Physical layer parameters: spans, fibers, amplifier parameters, etc.
      - Uncertainty: measuring errors, outdated measurements
    - 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

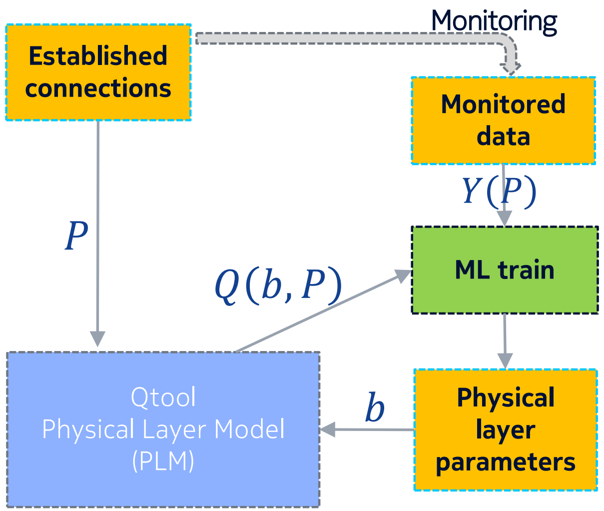




NOKIA Bell Labs

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

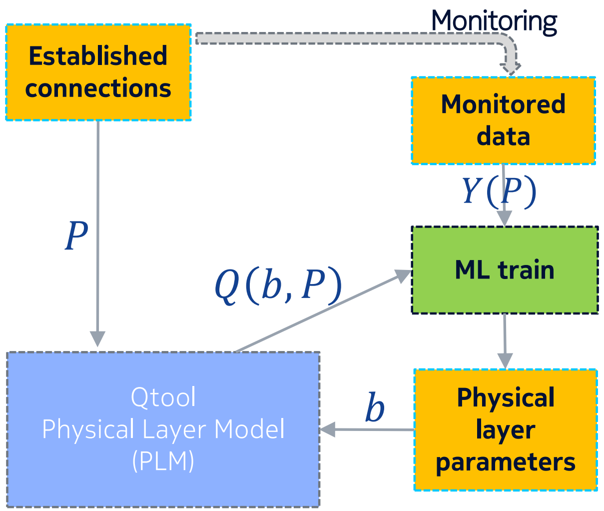
- Physical Layer Model (PLM)
  - Inputs
    - Connection parameters P
      routes, spectrum, TRx configuration (baudrate, modulation, etc.)
    - Physical layer parameters b
      spans, fiber attenuation, dispersion, nonlinear coefficients, amplifiers parameters, ROADM parameters
  - Output: lightpaths’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



[1] E. Seve, J. Pesic, C. Delezoide, S. Bigo, and Y. Pointurier, “Learning Process for Reducing Uncertainties on Network Parameters and Design Margins”, JOCN 2018

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

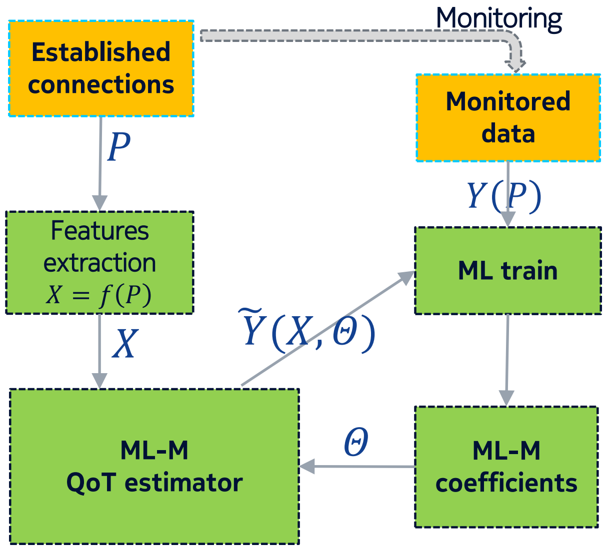
- ML training
  - Initialize physical layer parameters b^0 datasheets or (outdated) measurements
  - Fit (iteratively) parameters b^i to min the error $Y(P) - Q\left(b^i, P\right)$
    - Fitting algo depends on the PLM model, and if we know $\partial Q/\partial b_j$
    - If Q is nonlinear to some $b_j \rightarrow$ nonlinear fitting
  - Obtain fitted physical layer parameters b^*
- For a new connection, w ($P' = \{P \cup w\}$), 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)



NOKIA Bell Labs

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 Θ
      depend on the particular ML model (linear regression, NN, SVM, etc.)
  - Output: lightpaths' QoT estimates $\widetilde{Y}(X, \Theta)$
- Use monitoring $Y(P)$ to train the model and obtain Θ^* that yield low estimation error



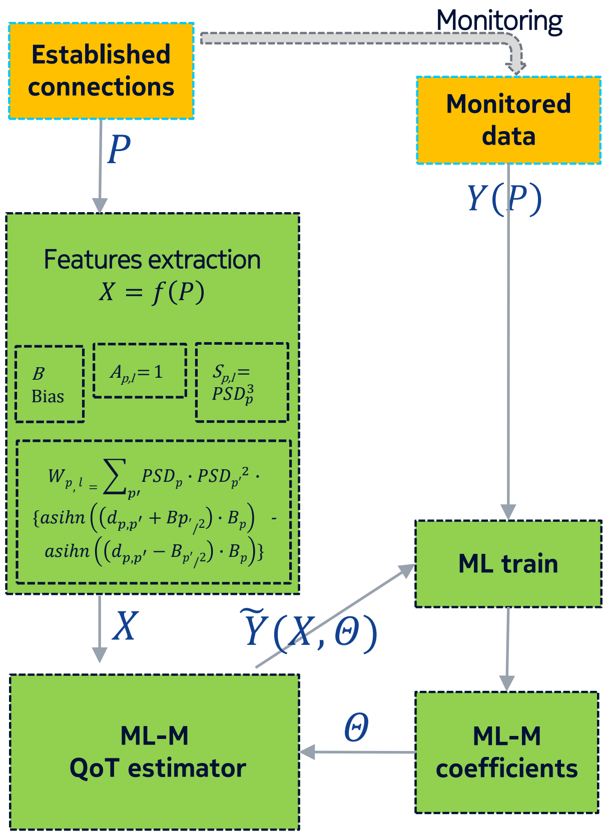
NOKIA Bell Labs

2^{nd} 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]$_{|P| \mathrm{x}(1+3|L|)}$
  Bias+3 sets of link features for the 3 major impairment classes
  - A: ASE, S: SCI, X: XCI

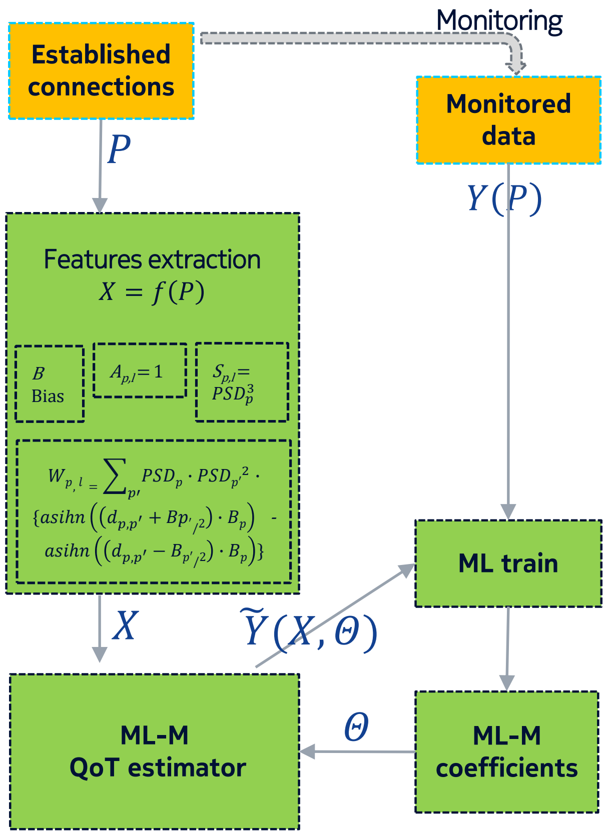
  e.g. $S_{p,i}=PSD_p^3$ (power spectral density) of lightpath p if it uses link i, else=0
  SCI noise contribution depends (linearly) on lightpath's PSD^3



NOKIA Bell Labs

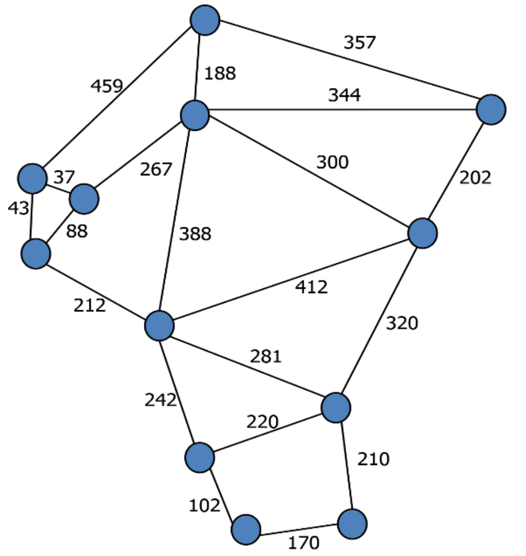
2nd method: Machine Learning Model (ML-M)

- Features designed so that the impairment noise contribution depends (close to) linearly on them
  - $X_{p,j}$ the jth 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 $\tilde{Y}(X, \Theta) = X \cdot \Theta$, 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



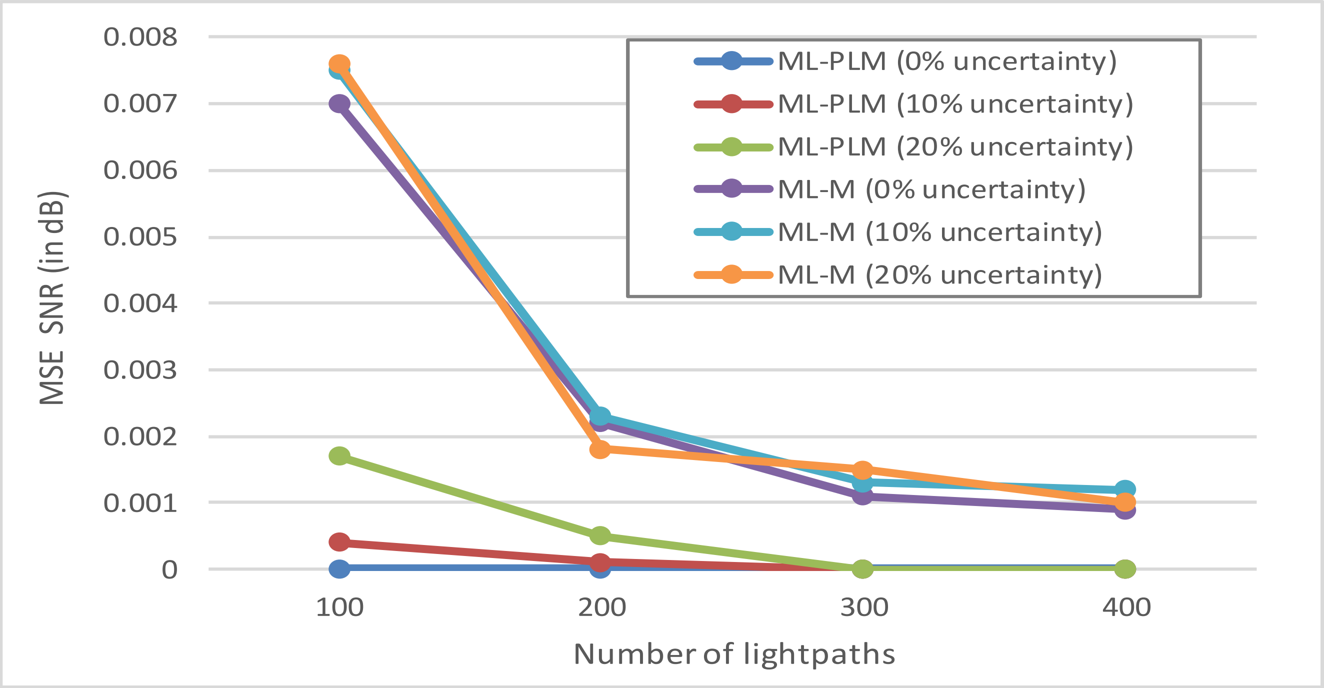
ML QoT Estimation – accuracy evaluation

- 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



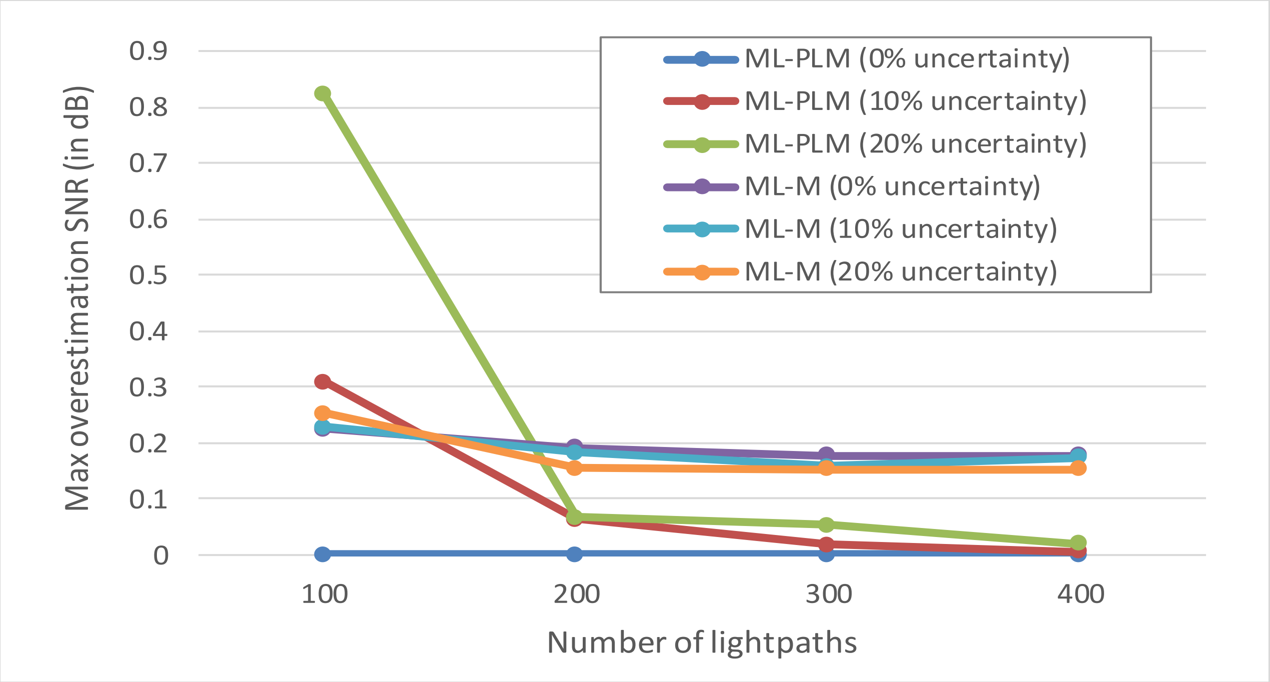
NOKIA Bell Labs

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



(untrained) PLM
max overestimation
0dB for 0% uncertainty
0.9dB for 10% uncertainty
2dB for 20% uncertainty

- Similar findings for max overestimation
- Design margin = max overestimation
  - $SNR_{over} = SNR_{est} - SNR_{real}$, for threshold SNR_{thr}, it is safe if $SNR_{real} > SNR_{thr} \Rightarrow SNR_{est} - SNR_{over} > SNR_{thr}$
- ML-PLM design margin: 0.05 dB, ML-M design margin: 0.2 dB for 200 lightpaths

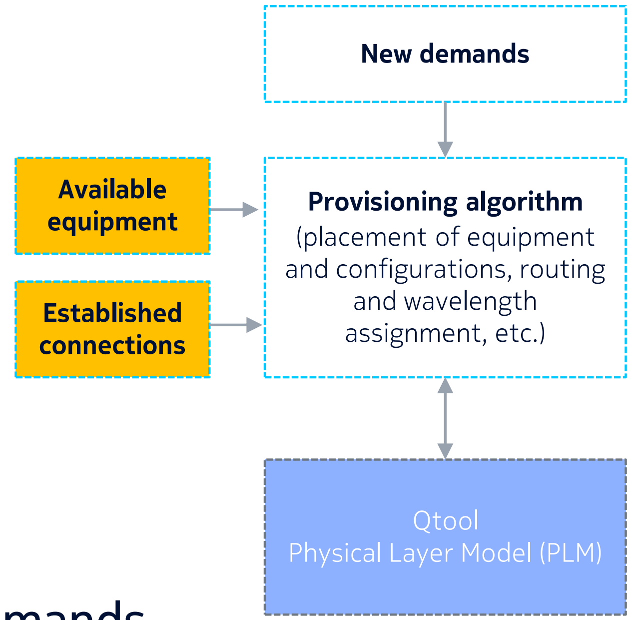
NOKIA Bell Labs

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

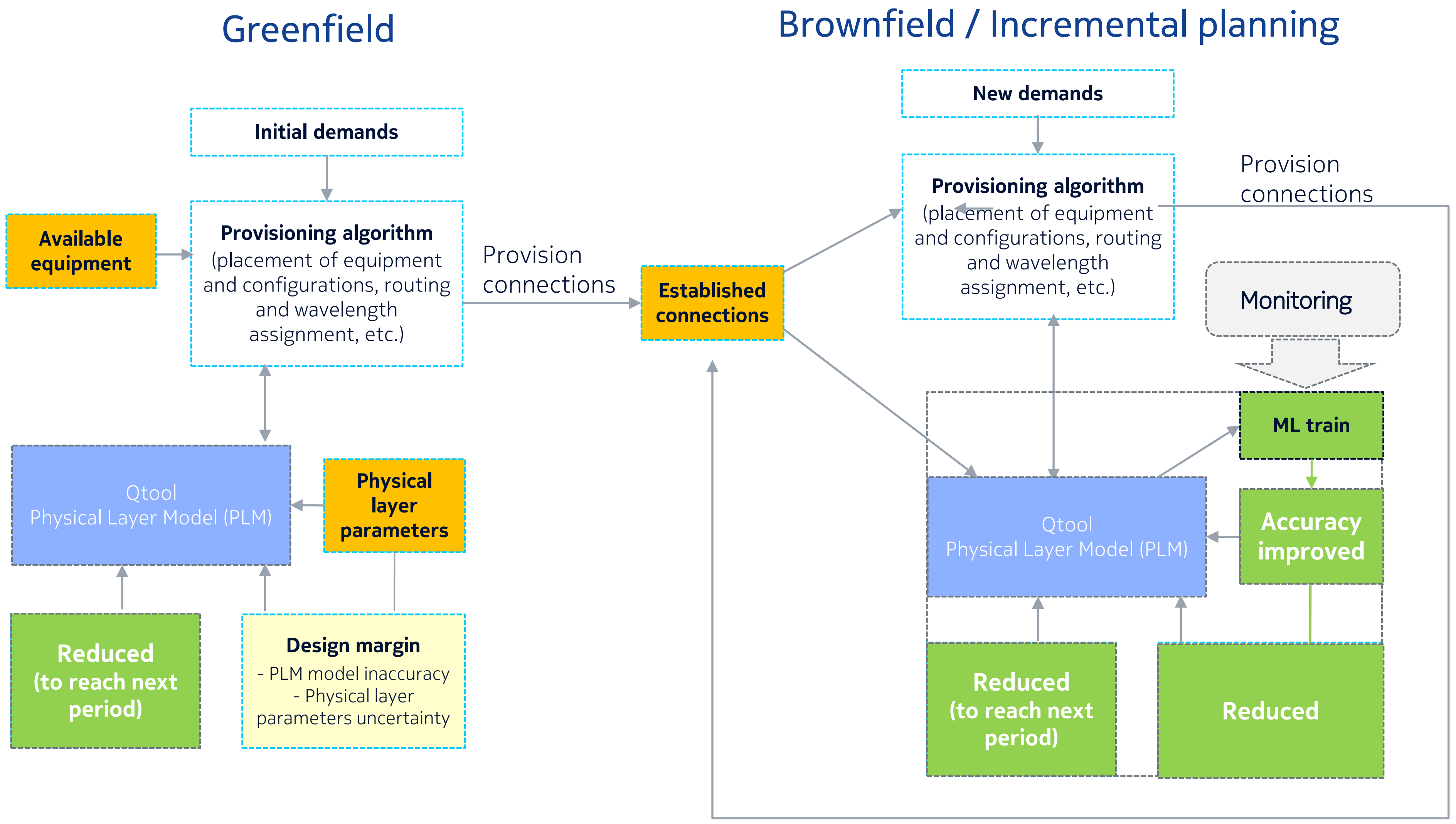
Incremental planning algorithm

- Input at the start of period τ_i
  - Traffic described by the remaining and new demands
  - TRx installed at previous periods / established lightpaths (up to τ_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]



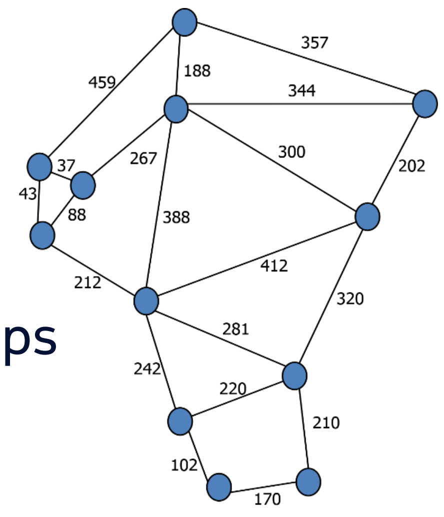
[1] P. Soumplis, K. Christodoulopoulos, M. Quagliotti, A. Pagano, E. Varvarigos, "Network Planning with Actual Margins", Journal of Lightwave Technology (JLT), 2017
[2] P. Soumplis, K. Christodoulopoulos, M. Quagliotti, A. Pagano, E. Varvarigos, "Multi-Period Planning With Actual Physical and Traffic Conditions", Journal of Optical Communications & Networking (JOCN), 2018

Incremental Planning and ML-PLM



NOKIA Bell Labs

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 (τ_0), TRx2 available at period 5 (τ_5)
  - TRx1: 32 Gbaud, DP-QPSK to DP-16QAM, SNR_{thr}=0.01dB, cost= 1, at period 0 (τ_0)
  - TRx2: 64 Gbaud, DP-QPSK to DP-32QAM, SNR_{thr}=0.01dB, cost= 1, at period 5 (τ_5)
- Cost reduction 10% per period

Data Rate (Gbps)	Baud Rate (Gbaud)	Mod Format	BOL ageing & BOL interf. & High design	EOL ageing & EOL interf. & Low design	EOL ageing & EOL interf. & High design
100	32	DP-QPSK	4720	3600	2280
150	32	DP-8QAM	2080	1600	1280
200	32	DP-16QAM	1040	800	560

Data Rate (Gbps)	Baud Rate (Gbaud)	Mod Format	BOL ageing & BOL interf. & High design	EOL ageing & EOL interf. & Low design	EOL ageing & EOL interf. & High design
200	64	DP-QPSK	4160	2800	2240
300	64	DP-8QAM	2720	1840	1440
400	64	DP-16QAM	1840	1280	960
500	64	DP-16QAM	1280	880	640

NOKIA Bell Labs

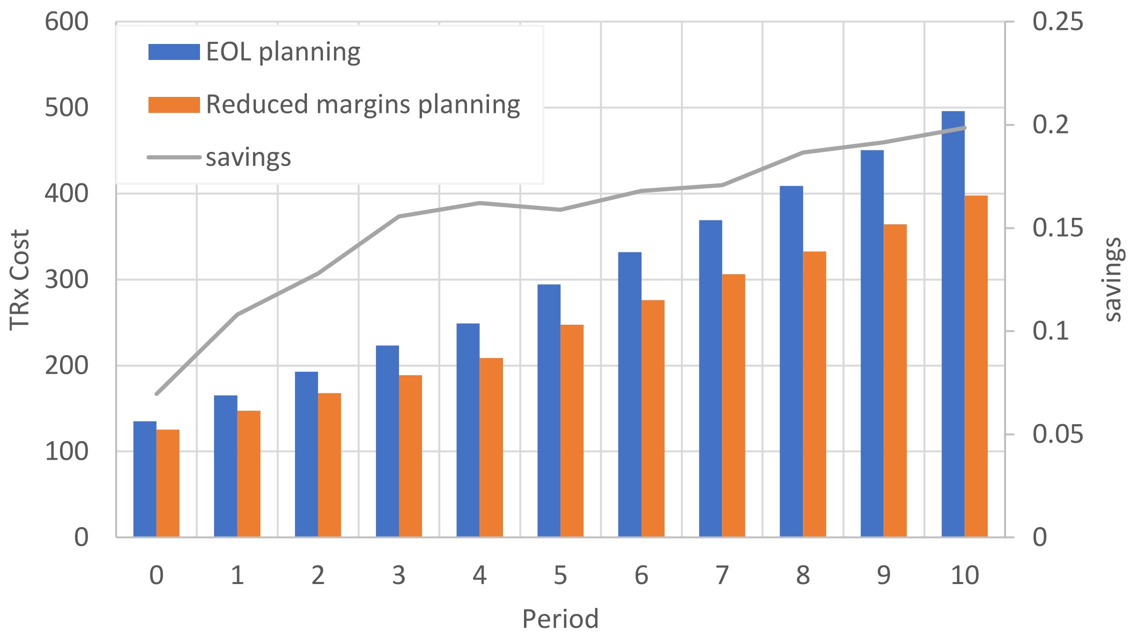
Case study – Physical layer evolution & margins

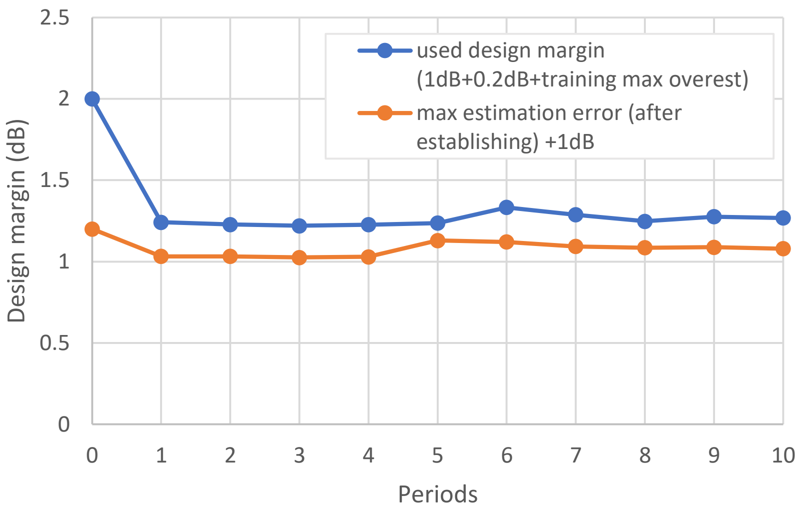
- Initialize with heterogeneous spans and uncertainty
  - Attenuation, dispersion, nonlinear coefficients uniformly around default values ±10%
  - Unknown to QoT estimator, requires ~1 dB margin
- Ageing: increase per period according to table
- 10 instances (load & physical layer), average results
- 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

Physical layer parameters evolution			Increase per period
system margin	Ageing	Transponder margin (dB)	0.05
		Attenuation (dB/km)	0.0015
		EDFA noise figure (dB)	0.1
		OXC loss (dB)	0.3
		Interference	According to load

NOKIA Bell Labs

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
  - Multiperiod planning case study

    ~20% savings with accurate QoT estimation/planning with reduced margins as opposed to EOL margins

Copyright and confidentiality

QoT estimation – state of the art

Paper	Qtool	ML method	Features	Simulations /Experiments	Comment
[1]	without Qtool	Regression Kriging (linear correlation)	Interference aware links (1/SNR per link, different links according to adjacent lightpaths)	Simulations, GN model as ‘ground truth’	Homogeneous’ spans
[2]	without Qtool	Classification K-nearest neighbors, Random Forest	#hops, path length, longest link, modulation format, network traffic volume, …	Simulations, GN model with worst case interference as ‘ground truth’	Worst case interference ‘Homogeneous’ spans
[3]	with Qtool (inhouse, GN model)	Regression Gradient decent	Input parameters of Qtool (power at nodes, XXX)	Simulations, GN model and inhouse Qtool as ‘ground truth’, worst case interference	Worst case interreference
[4]	with Qtool (inhouse close to GN)	Calculate the derivatives, similar to gradient decent	length-dependent loss and nonlinear intensity (NLI) 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).	Experiments, 6 nodes !	
[5]	With Qtool (GN model)	Regression Maximum likehood / extended Kalman filter	Not clearly described, for sure SNR and wavelength	Experiments (6 nodes, VOAs to emulate different link lengths), and simulations (to evaluate benefits)	Account for non-homogeneous amplification
[6]	Without Qtool	Ridge regression LASSO regression LASSO with quadratic features Multilayer perceptron Gaussian process regression Gradient boosted regression trees Random forest regression trees	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 passthrough 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.	Experiments	

[1] I. Sartzetakis, K. Christodoulopoulos, C. Tsekrekos, D. Syvridis, E. Varvarigos, "Quality of transmission estimation in WDM and elastic optical networks accounting for space–spectrum dependencies", JOCN 2016
[2] C. Rottondi, L. Barletta, A. Giusti, M. Tornatore, “Machine learning method for quality of transmission prediction of unestalished lightpaths”, JOCN 2018
[3] E. Seve et. al., “Learning Process for Reducing Uncertainties on Network Parameters and Design Margins,” JOCN, 2018.
[4] M. Bouda, et. al. “Accurate Prediction of Quality of Transmission Based on a Dynamically Configurable Optical Impairment Model”, Journal of Optical Communications and Networking, 2018. (Fujitsu)
[5] P. Samadi et. al., “Quality of Transmission Prediction with Machine Learning for Dynamic Operation of Optical WDM Networks,” ECOC 2017. (Bergman)
[6] G Choudhury, et. al., ”Two use cases of machine learning for SDN-Enabled IP/Optical Networks: traffic matrix prediction and optical path perfomrnace prediction”, JOCN 2018

