

Master Thesis
Extended GN Model for L-C Band with final
experimental verifications



Double Degree Program

MSc Communications and Computer Networks Engineering

Máster Universitario en Ingeniería de Telecomunicación

Ángel Alonso Gómez

Supervisors:

Prof. Dr. Andrea Carena

Prof. Dr. Luis Salgado

Company Supervisor:

Dr. Antonio Napoli

Abstract

Nowadays, the huge data traffic growth has forced telecommunications companies and vendors to deploy ultra-wideband optical systems. To do so, they need of tools that allow them to estimate the system performance and to plan properly the optical network. On the physical layer, the metric used to characterize the system performance is the well-known Signal-to-Noise Ratio (SNR). This value determines the Quality of Transmission (QoT). Beyond of ASE noise impairments, the noise-like "non-linear interference" (NLI) power is also a critical issue that should be considered. The goal of this project is to evaluate the well-known Gaussian Noise Model (GN-Model) and the forefront version, the Generalized Gaussian Noise Model (GGN-Model)[1]. On the Introduction Chapter, the current state-of-the-art is briefly described and which are the cutting-edge tools to estimate system performance, as well as main optical fiber parameters involved on the non-linear interference generation. On the second Chapter, the metric used to defined the QoT is explained and through this value, the system performance is estimated. The metric is basically the SNR as it was introduced earlier. Following what was set forth above, the noise power is not only composed of ASE noise power, but also of noise-like NLI power. On the third Chapter, the GN-Model is thoroughly described as well as its physical meaning. Some important results came out after comparing how the noise-like NLI power is piled up varying system parameters (frequency spacing, number of spans etc). 50% of total noise-like NLI power is agglomerated between 3-7 channels (it depends on the number of spans and other system parameters). Furthermore, it was reported that the smaller the frequency spacing, the more incoherent is the non-linear effect accumulation. Incoherent accumulation symbolize that the noise-like NLI power generated on one span is summed up in power with the contributions of the rest of spans. On the fourth Chapter, the GGN-Model derivation is deeply explained. Main improvements with respect to GN-Model are considered as well. On the fifth Chapter, the model implementation and how all files interact each others is clarified. Finally on the sixth Chapter, all tests and simulations carried out through this thesis are illustrated. The most important one is the full C-Band (191.35-196 THz) + 15 Channels on L-Band (190.2-190.9 THz) without tilt compensation. The performance on a channel, placed on the lowest side of the operating bandwidth, is reported to be 7 dB worst than that obtained on a channel, placed on the highest side of the operating bandwidth in terms of frequency. Moreover, it has been demonstrated, how GN-Model reports mistaken estimates on L-Band for L+C system without Stimulated Raman Scattering (SRS) cross-talk compensation. Indeed, a difference of approximately 2000 km is reported in terms of maximum distance given a certain Bit Error Rate requirement (BER).

Key Words— GN-Model, GGN-Model, C+L systems, Matlab, Ultra wide-band Optical Networks, *OSNR*, *SNR*, *SRS* and *BER*

*Dedicado a
mi familia.*

Contents

1	Introduction and Coherent Optical Systems	7
1.1	Scenario under Analysis	8
1.2	Fiber Loss. Attenuation Coefficient	9
1.3	Chromatic Dispersion Parameter	10
1.4	Kerr Effect	11
1.5	Raman Scattering. Raman Cross-Talk	12
2	Gaussian Noise Model	15
2.1	GN-Model Reference Formula	15
2.2	Physical Interpretation of the GN-Model	16
2.3	FWM Efficiency Factor	17
2.4	Coherent Accumulation Factor	18
2.5	Hyperbolic Coordinates	18
2.6	GN-Model Approximated Formula	20
2.7	NLI Accumulation	21
3	Generalized Gaussian Noise Model	23
3.1	Generalized GN-Model Reference Formula	23
3.2	FWM Efficiency Factor	31
4	Metric to assess Quality of Transmission (QoT)	34
4.1	Metric definition and GN-Model implications	34
4.2	Maximum Length	36
5	Model implementation	38
5.1	Code Structure	38
5.1.1	Fiber Parameters	38
5.1.2	System-Signal Parameters	39
5.1.3	GN-Model Approximated Formula	40
5.1.4	GN-Model Cartesian Coordinates	40
5.1.5	GN-Model Hyperbolic Coordinates	41
5.1.6	GN-Model Hyperbolic Coordinates for $f=0$	41
5.1.7	GGN-Model Matrix Solution	41
5.1.8	GGN-Model Loop/Matrix Solution	42
5.1.9	GGN-Model Hybrid Solution	42

5.2	Code performance	43
5.2.1	Hyperbolic Coordinates vs Cartesian Coordinates	44
5.2.2	Generalized Gaussian Noise Model Matrix Solution	45
5.2.3	Generalized Gaussian Noise Model Loop/Matrix Solution	46
5.2.4	Generalized Gaussian Noise Model Hybrid Solution	47
6	Experimental verification and simulations	48
6.1	Telecom Italia Field Trial	48
6.2	Maximum Distance GN-Model	51
6.3	Experiment at Orange Laboratory	52
6.3.1	Laboratory Setup	52
6.3.2	Results and Conclusions	53
6.4	Experiment at Coriant Laboratory. Full C-Band + 15 L-Band	56
6.4.1	Laboratory Setup	56
6.4.2	Measurements and Conclusions	59
6.5	15 Channels on C-Band and 35 Channels on L-Band	63
7	Conclusions	65
	Appendices	66
A	Matlab Files	67

List of Figures

1.1	Multi-span Optical System Scenario [2]	8
1.2	Coherent Transmission Modulator [3]	9
1.3	Blue solid line represent both α and D for G.652A and orange solid line represents α for G.652D [4]	10
1.4	Blue solid line represents the Raman Gain and red solid one represents the triangular approximation	13
1.5	Blue solid line input WDM Signal PSD and red solid line tilted WDM Signal PSD due to SRS effect	14
2.1	Red area represents the integration domain for MCI, yellow one does the XCI one and the blue one does the SCI one.	17
2.2	Clipping off the integration beyond -30 dB makes the integration domain 75% smaller	19
2.3	$\chi(\nu_1, 0)$ and threshold equal to -30 dB	20
2.4	Red solid line 50 GHz frequency spacing, green solid line 38.4 GHz and yellow solid line 32 GHz	21
2.5	Blue solid line $N_s = 50$, green solid line $N_s = 20$ and red line $N_s = 1$	22
3.1	Blue solid line real part of the complex exponential and red dashed lines field profile of pumps frequencies, when $(f_1 - f)(f_2 - f)$ is low.	31
3.2	Blue solid line real part of the complex exponential and red dashed lines field profile of pumps frequencies, when $(f_1 - f)(f_2 - f)$ is high.	32
3.3	Blue solid line represents I_2^{GN} and red dashed line represents I_2^{GGN}	33
4.1	Reference Bandwidth for SNR (Rs) and for OSNR (Bn)	35
4.2	Random example of the I/Q diagram provided by the receiver	36
4.3	Green solid line represents the maximum length for a linear system given the input power and blue solid line does it for a non-linear system	37
5.1	Code Architecture	43
5.2	The graph shows the computing time of the GN-Model for different number of channels. Blue solid line represents Cartesian Coordinates approach and red solid line does Hyperbolic Coordinates one. Cartesian Coordinates approach is faster than Hyperbolic Coordinates one until 10 channels. Beyond 10 channels is the other way around.	44

LIST OF FIGURES

6.1	Field Trial Architecture. This figure has been taken from [5] (Fig. 1)	49
6.2	The optimal input power is 4 dBm, 3 dBm and 2 dBm for DP-4QAM, DP-8QAM and DP-16QAM respectively. Dashed line represent GN-Model Approximated Formula with incoherent accumulation. This model reports larger distances for both DP-4QAM and DP-8QAM, because for these mod. schemes the frequency spacing is higher, then NLI accumulation is more coherent. For DP-16QAM scheme the frequency spacing is lower.	51
6.3	Maximum length for different input powers. SMF-QPSK case	52
6.4	Red solid line represents the central channel, blue solid line left channel and yellow dashed line right one	53
6.5	Red solid line represents the central channel, blue solid line left channel and yellow solid line right one. It is observed that equalization stages are placed on 6th and 13th spans.	54
6.6	Red solid line represents the central channel, blue solid line left channel and yellow solid line right one	55
6.7	Blue solid line represents GN-Model and green solid line represents GGN-Model. GN-Model reports an over estimation at 193.5 THz with respect to GGN-Model and an under estimation at 194.5 THz with respect to GGN-Model.	55
6.8	Experiment Architecture	56
6.9	Input 96 Channels C-Band	57
6.10	Input 15 Channels L-Band	57
6.11	Output 96 Channels C-Band after 5 spans	59
6.12	Output 15 Channels L-Band after 5 spans	59
6.13	Back-to-back measurements	60
6.14	Span Block Diagram	61
6.15	Blue solid line represents Strategy 1 and red solid line represents Strategy 2. Yellow bars represent +/- 1 dB around the measured value. In other words, an estimation inside the interval $[x_{measured} - 1, x_{measured} + 1]$ is accepted.	62
6.16	Measure for channel at 193.75 THz. Square-like dots represents the measurements of η_{nl} and solid red line the simulated one. 0.5 dB maximum difference among measurements is reported.	63
6.17	Blue solid line GGN-Model at 190.3 THz, blue dashed line GN-Model at 190.3, green solid line GGN-Model at 191.8 THz and green dashed line GN-Model at 191.8 THz. Both models report similar values at 191.8 THz. On the other hand, GN-Model fails at 190.3 THz.	64

List of Tables

6.1	Main Experiment Parameters	49
6.2	Main Fiber Parameters	49
6.3	Back-to-back values in terms of OSNR and SNR	50
6.4	Main Fiber Parameters	57
6.5	System Parameters	58
6.6	Span Losses	58
6.7	BER-SNR	63

Chapter 1

Introduction and Coherent Optical Systems

The continuous growth of IP-traffic requires that telco companies increase the bandwidth capabilities in order to fulfill customer's requirements. To cope with this, system vendors evolved their products from direct-detection to coherent detection that allows to improve spectral efficiency and reaches. In coherent systems accumulated dispersion can be compensated through digital signal processing. This leads to deployment of uncompensated transmission (UT), which presents several benefits (e.g. better performance and no dispersion-managed links). This is one of the main reasons why current optical systems generation is based on this approach. One of the main parameters for telecommunication operators (to measure system performance) is the OSNR (Optical signal-to-noise-ratio), defined as the ratio between optical signal power and noise power, i.e. ASE noise, from the amplifiers. Besides the ASE, another parameter strongly influences the system performance, this is the noise-like power induced by nonlinear effects. These systems and their main effects can be efficiently modeled, and thus the network performance can be estimated. The most widely used model is the so-called Gaussian Noise Model, that effectively addresses the impact of non-linear propagation. The GN-Model helps the Physical Simulation Environment (PSE) [6], which under the Telecom Infra Project ,Inc. (TIP), intends to provide an Open-Source physical layer performance-predicting module or tool.

Historically, due to limitation in optical amplification, the common band for transmission has been C-Band (1528-1566 nm). Thus, the Gaussian Noise Model has been only designed and, fully tested in this spectral region. With the availability of new optical amplifiers delivering gain beyond the C-Band, there is a growing interest in deploying future system in the extended C+L band, spanning from 1530 nm to 1600 nm. The goal of this project is to extend GN-Model to GGN-Model for a reliable application over the C+L band.

1.1 Scenario under Analysis

The Multi-span and Coherent Transmission scenario is organized like a typical communication system, i.e. it is composed of a transmitter, a channel and a receiver. As multi-span approach is followed, the channel white box brings together many "channels". Figure 1.1 summarizes the scenario.

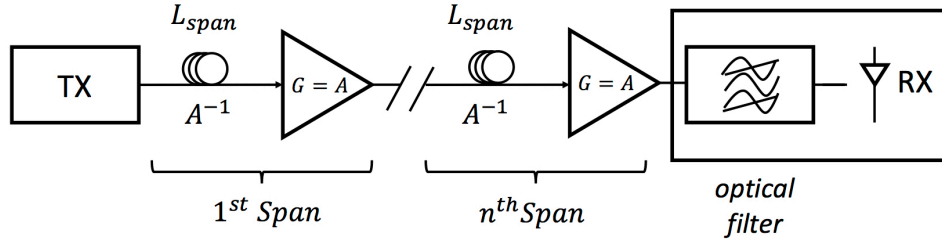


Figure 1.1: Multi-span Optical System Scenario [2]

Regarding the Coherent System, the modulator inside the transmitter is constituted by four branches, two in-phase components and two quadrature components. This is the consequence of the coherent system approach, where the "information" is not on the amplitude, but on the phase of the signal. One of the two in-phase components represents the x-component and the other one the y-component. For the quadrature components occurs the same. In other words, the information is transmitted both along x-axis and y-axis (double polarization). Therefore, PM-QPSK transmits four bits per symbol, meanwhile QPSK does two one per symbol. Figure 1.2 illustrates modulator scheme.

1.2. FIBER LOSS. ATTENUATION COEFFICIENT

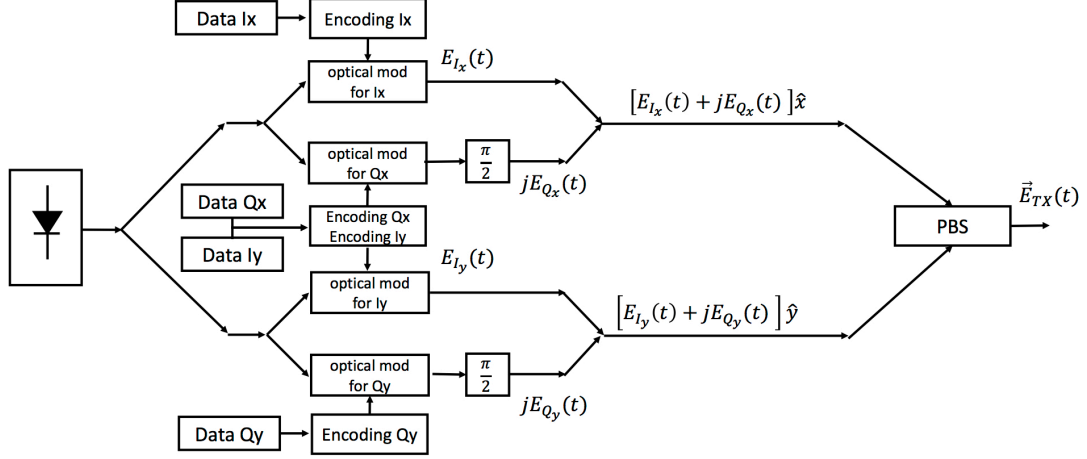


Figure 1.2: Coherent Transmission Modulator [3]

1.2 Fiber Loss. Attenuation Coefficient

The attenuation coefficient α includes all sources of attenuation. It is variable against the frequency, which will have further implications later. The evolution of the power along the optical fiber is explained by an exponential Law.

$$P_{out} = P_{in}e^{-\alpha L} \quad (1.1)$$

$$\alpha(dB/km) \approx 4.343\alpha \quad (1.2)$$

As it has been said before, the attenuation coefficient varies with the wavelength of the transmitted light. According to physical theory, the low-loss window is placed around 1550 nm, where $\alpha = 0.2$ dB/km. This is the so-called C-Band. Nowadays, this region is the widely used bandwidth. However, both the needs of increasing the bandwidth and the improve on fiber manufacturing are making the usage of other region also interesting. For instance, L-Band has become quite interesting. On this band the attenuation coefficient variation is more abrupt as in C-Band as it can be observed on Figure 1.3

1.3. CHROMATIC DISPERSION PARAMETER

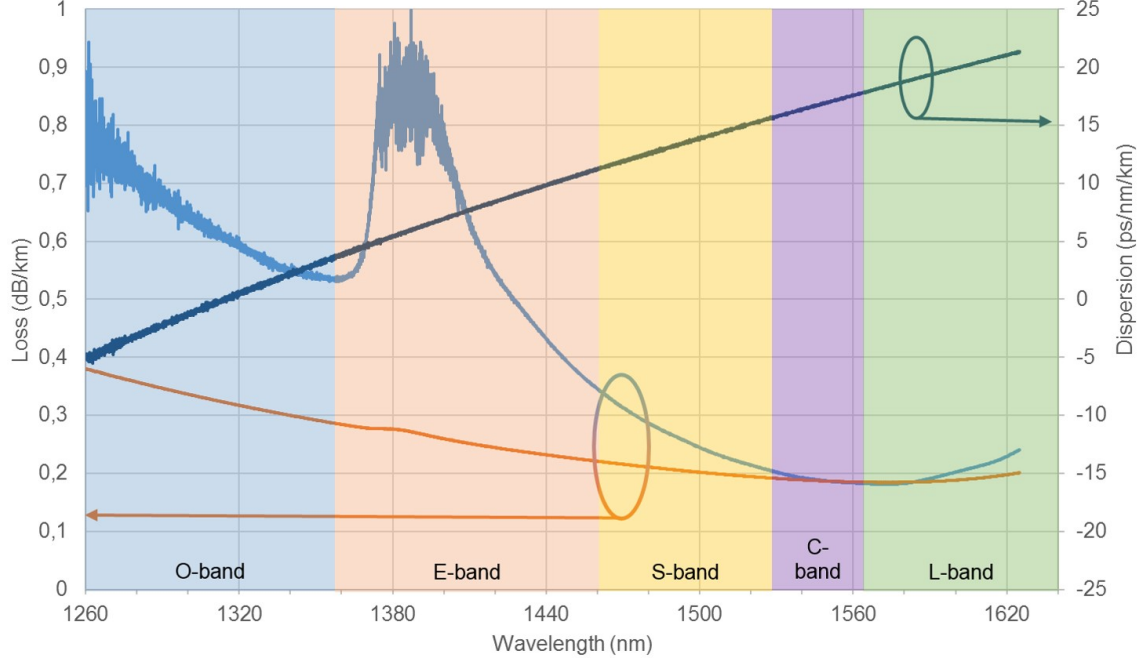


Figure 1.3: Blue solid line represent both α and D for G.652A and orange solid line represents α for G.652D [4]

1.3 Chromatic Dispersion Parameter

In single mode fibers different spectral components travel along the fiber with different propagation constant. This leads to pulse broadening or distortion. Recall that the propagation constant can be written as its Taylor expansion.

$$\begin{aligned}\beta(\omega) &= \beta_0 + \beta_1 \Delta\omega + \frac{1}{2}\beta_2 \Delta\omega^2 + \frac{1}{6}\beta_3 \Delta\omega^3 \\ \beta(f) &= \beta_0 + \beta_1 2\pi(f - f_0) + \frac{1}{2}\beta_2 (2\pi(f - f_0))^2 + \frac{1}{6}\beta_3 (2\pi(f - f_0))^3\end{aligned}\quad (1.3)$$

The terms β_0 , β_1 , β_2 are the coefficients of the Taylor series that models the $\beta(f)$. Note that the coefficients depends on central point where Taylor series is computed. Thus, this coefficients are frequency-dependent [7].

1. β_0 Frequency independent phase term: It is basically irrelevant for most TLC applications
2. β_1 Frequency dependent phase term: It is linearly frequency-dependent, then it does not distort the signal.
3. β_2 Frequency dependent phase term: It is non-linearly frequency-dependent, then it distorts the signal. It is the so-called Fiber Chromatic Dispersion.

1.4. KERR EFFECT

The group velocity is the velocity at which the envelope of a set of optical pulses travels through the fiber. The group velocity can be found using $v_g = \frac{1}{\beta_1}$. Furthermore, the β_2 parameter represent dispersion of the group velocity and is responsible for pulse broadening. The group delay is defined as the delay experienced by a spectral component f compared to those at f_0 . The group delay is defined as

$$\tau_g(f) = 2\pi\beta_2(f - f_0)L \quad (1.4)$$

However, β_2 is not the widely used parameter, but the dispersion parameter D . It is defined as follows:

$$D = \frac{d\beta_1}{d\lambda} \quad (1.5)$$

The dispersion parameter D has been defined in order to relate group delay ($\Delta\tau_g$) variation with wavelength (λ). Then, Equation 1.4 can be re-defined as:

$$\begin{aligned} \tau_g(\lambda) &= \tau_g(\lambda_0) + D(\lambda - \lambda_0)L \\ \Delta\tau_g &= D\Delta\lambda L \end{aligned} \quad (1.6)$$

And the dispersion parameter D is related with the group velocity dispersion (GVD) parameter β_2 :

$$D = -\frac{2\pi c}{\lambda^2}\beta_2 \quad (1.7)$$

Both terms vanish at a wavelength around 1300 [nm] for pure silica fibers. This wavelength is called zero-dispersion wavelength and is denoted as λ_D . Finally, the typical value is 16 [ps/nm/km] at 1550 [nm].

1.4 Kerr Effect

The Kerr effect causes a variation in the glass refractive index, which depends on the optical power $P(z)$. It is the most important non-linear effect on the fiber. The conventional refractive index is defined as n_L , n_2 is the non-linear index coefficient, $P(z,t)$ is the optical power and A_{eff} is the effective area. However, the real refractive index varies with time, as optical power does. Therefore an "instantaneous" refractive index must be defined [8].

$$n(z, t) = n_L + n_2 \frac{P(z, t)}{A_{eff}} \quad (1.8)$$

The effective area can be approximated to the area where the optical mode is confined in the fiber. Then, it can be expressed as:

$$A_{eff} \approx \pi r^2 \quad (1.9)$$

1.5. RAMAN SCATTERING. RAMAN CROSS-TALK

Where r is the core radius of the fiber.

Since the refractive index and the propagation constant are related, a variation of the refractive index leads to a variation of the propagation constant.

$$\begin{aligned}\Delta\beta &= \frac{2\pi}{\lambda}\Delta n \\ &= \frac{2\pi}{\lambda}n_2\frac{P(z)}{A_{eff}}\end{aligned}\tag{1.10}$$

Then, the non-linear coefficient is defined:

$$\gamma = \frac{2\pi n_2}{\lambda A_{eff}}\tag{1.11}$$

The Kerr Effect causes a phase shift induced by the signal power. This phase shift is expressed as [8]

$$\Phi_{NL}(z, t) = -\gamma|E(0, t)|^2 z_{eff}\tag{1.12}$$

Here arises an important parameter in optical communications: the well-known effective length. It is defined as the length where the field amplitude only suffers a phase shift by the non-linearity (1.12).

$$z_{eff} = \frac{1 - e^{-2\alpha z}}{2\alpha}\tag{1.13}$$

1.5 Raman Scattering. Raman Cross-Talk

Raman Scattering is a physical phenomena that occurs when a photon interacts with the propagation medium. The molecules, of which the guiding material is composed, can only oscillate at two energy levels, which are frequencies indeed. When these molecules are excited by the incoming photon, they jump up to a higher level. However, this energy level is prohibited. Therefore, they must jump down right after. As a result, if the final energy level is higher than the initial one, then less optical power is released together with the mechanical energy. On the other hand, if the final energy level is lower than the initial one, more mechanical energy is absorbed in order to release more optical energy as well [9].

There are two different kinds of Raman Scattering: The Spontaneous Raman Scattering and the Stimulated Raman Scattering (SRS). The former occurs in the absence of incident photon, so that, new photons are generated. The latter does as a consequence of an incoming photons flow. Therefore, energy from these photons is transferred to the propagating optical field. The most important effect of Raman Scattering are: Raman Cross-Talk, Raman Amplification and Raman Laser. Both Raman Amplification and Raman Laser are out of the scope of this thesis. We focus on Raman Cross-Talk effect.

1.5. RAMAN SCATTERING. RAMAN CROSS-TALK

Raman Cross-Talk is a negative effect of Raman Scattering that impairs the performance of a WDM system. Higher frequency components transfers power to lower ones. As a results, the spectra is tilted. The effect is governed by a set of discrete differential equations [10]. Theses equation are based on the triangular approximation to the Raman Gain spectrum. The triangular approximation assumes that Raman Gain varies linearly until a certain cut-off frequency (Typically 13.5 THz). Figure 1.4 illustrates it.

$$\frac{\partial P_n(z)}{\partial z} + \alpha P_n(z) + \left(\frac{g' \Delta f}{2A} \right) P_n(z) \sum_{m=1}^N (m - n) P_m(z) = 0 \quad (1.14)$$

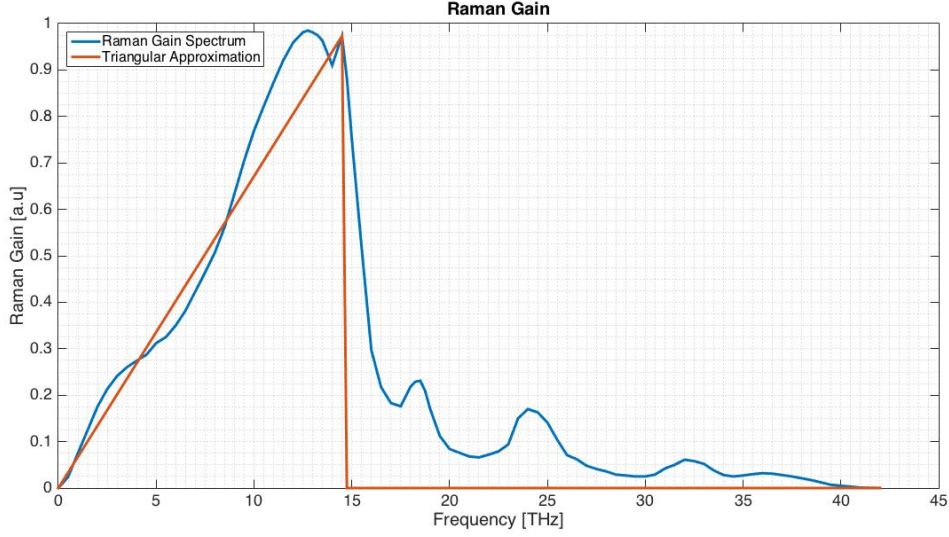


Figure 1.4: Blue solid line represents the Raman Gain and red solid one represents the triangular approximation

In Equation (1.14), $P_n(z)$ is the power of the n th channel as a function of the distance z , α is the power attenuation constant, g' is the slope of the triangular approximation to Raman gain, Δ_f is the interchannel frequency spacing, A is the effective area and $P_m(z)$ is the power of the m th channel as a function of the distance. The 2 in the denominator represents the polarization averaging. The system of differential equation exhibits the general solution:

$$P_n(z) = P_{no} J_o e^{-\alpha z} e^{G J_o (n-1) Z_e} \left[\sum_{m=1}^N P_{mo} e^{G J_o (m-1) Z_e} \right]^{-1} \quad (1.15)$$

In Equation (1.15), P_{no} is the input power of the n th channel, J_o is the total input power and is defined as $J_o = \sum_{m=1}^N P_{mo}$, G is the Raman gain $G = \frac{g' \Delta_f}{2A}$ and Z_e is the

1.5. RAMAN SCATTERING. RAMAN CROSS-TALK

effective length $Z_e = \frac{1-e^{-\alpha z}}{\alpha}$.

As a result, the input power spectrum on any span is not flat in frequency anymore after being propagated through previous spans (It is assumed that amplifiers compensate fiber loss). However, the input power the is tilted due to SRS cross-talk. In other words, the lower frequency channels receive power from the higher frequency channels (tilt shape).

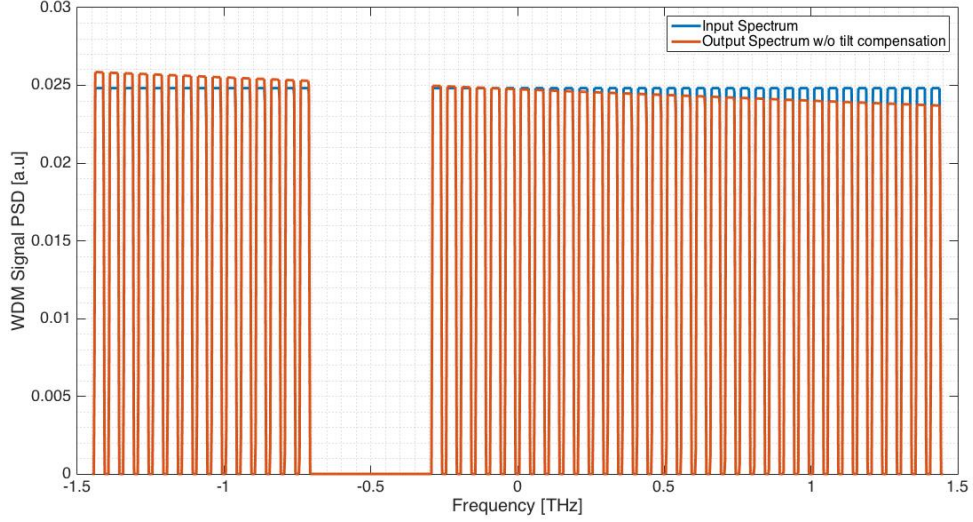


Figure 1.5: Blue solid line input WDM Signal PSD and red solid line tilted WDM Signal PSD due to SRS effect

Chapter 2

Gaussian Noise Model

The Gaussian Noise Model aims at assessing the non-linear propagation and providing a practical tool that allows to analyze system performance before deploying it. This model belongs to a family named as perturbation models [11]. The main assumption is simply to consider non-linearity as a perturbation. In other words, the non-linear disturbance is considered relatively small. This has been proved to work at common optimal system launch power. Moreover, a second assumption is that the transmitted signal behaves as a stationary Gaussian noise. Finally, the third assumption is that the signal disturbance produced by the non-linearity effect is assumed as Additive Gaussian Noise.

2.1 GN-Model Reference Formula

The Gaussian Noise Reference Formula, from now on GNFR Equation 1 [12], can be written as follows:

$$G_{NLI}(f) = \frac{16}{27} \gamma^2 L_{eff}^2 \psi(L_s, f)^2 \iint_{-\infty}^{\infty} G_{WDM}(f_1) G_{WDM}(f_2) G_{WDM}(f_1 + f_2 - f) \rho(f_1, f_2, f) \chi(f_1, f_2, f) df_1 df_2 \quad (2.1)$$

The symbols used are listed bellow:

1. γ : Kerr Effect Coefficient [$W^{-1}km^{-1}$]
2. L_{eff} : Effective Length [km]
3. $G_{WDM}(f)$: WDM Signal Power Spectral Density [W/Hz]
4. $\rho(f_1, f_2, f)$: FWM Efficiency Factor
5. $\chi(f_1, f_2, f)$: Coherent Accumulation Factor
6. α : Attenuation Constant [$1/km$]
7. β_2 : Group Velocity Dispersion Factor [ps^2/km]

2.2. PHYSICAL INTERPRETATION OF THE GN-MODEL

8. L_s : Span Length [km]

9. $\psi(L_s, f)^2$ Optical-Power Propagation function.

$G_{NLI}(f)$ is the NLI noise PSD. On the scientific literature (e.g. [12]), $\psi(L_s, f)^2$ is removed. The reason comes from that it is the amount of power loss on the fiber and it is compensated by the amplifiers. Therefore, it depends on where $G_{NLI}(f)$ is computed, before or after the EDFA amplifier. The scenario of this chapter considers that the fiber parameters does not change with neither frequency nor space, so that, $\psi(L_s, f)^2 = e^{-2\alpha L_s}$.

2.2 Physical Interpretation of the GN-Model

Looking at the integral factor three actors are involved. Firstly, $G_{WDM}(f_1)G_{WDM}(f_2)G_{WDM}(f_1 + f_2 - f)$ represents the PSD of the three spectral components involved in the Four-Wave Mixing process, which is an inter-modulation phenomenon that takes place inside optical fibers. This process should be taken into account when the optical fiber is considered a non-linear system. Two or three spectral components interact each other and a new one is generated as a result. Secondly, $\rho(f_1, f_2, f)$ is the non-degenerate FWM efficiency of their beating. Finally, $\chi(f_1, f_2, f)$ is shown to be the coherent accumulated factor that takes multiple spans into account and how NLI disturbance is being accumulated span after span. From the physical point of view, the disturbance produced by the beating process can be classified in three kind of interference: SCI (Self-Channel Interference), XCI (Cross-Channel Interference) and MCI (Multiple-Channel Interference). The channel whose one of its frequency is being analyzed is named channel under analysis.

1. Self-Channel Interference (SCI) All three spectral components (f_1, f_2 and f_3) belongs to the channel under analysis.
2. Cross-Chanel Interference (XCI) It represents the interplay between the channel under analysis and another channel
3. Multiple-Channel Interference (MCI) Each triple involves at least two channel apart from the channel under analysis.

Figure 2.1 depicts the integration domain for $G_{NLI}(0)$ and the Nyquist Case, that is when $B_{ch} = \Delta f$.

2.3. FWM EFFECIENCY FACTOR

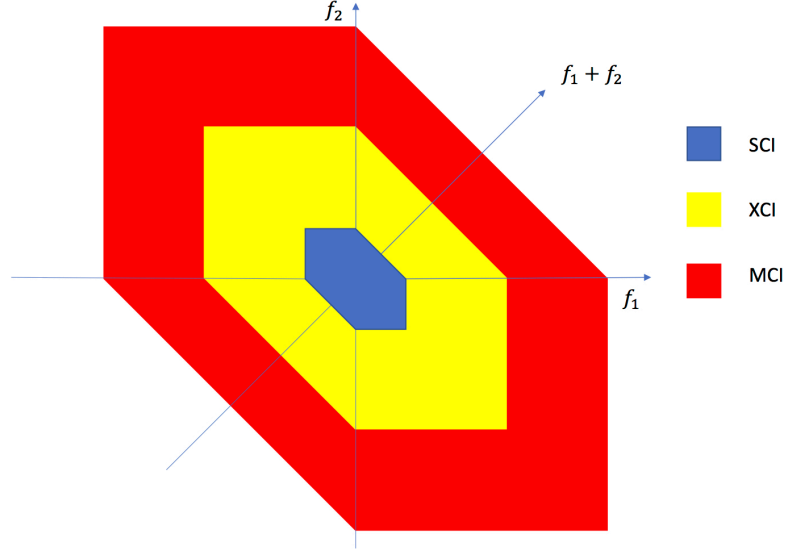


Figure 2.1: Red area represents the integration domain for MCI, yellow one does the XCI one and the blue one does the SCI one.

For no Nyquist-Case, i.e. $B_{ch} < \Delta f$, the integration domain is formed of many islands, so that, not all f_1 's that belongs to the interval $[-B_{WDM}/2, B_{WDM}/2]$ needs to be integrated (Chapter V [12]).

2.3 FWM Efficiency Factor

This factor represents the efficiency of the beating involved in FWM process. It is normalized with respect to L_{eff}^2 , then its maximum value is 1.

$$\rho(f_1, f_2, f) = \left| \frac{1 - e^{-2\alpha L_s} e^{j4\pi^2 \beta_2 L_s (f_1 - f)(f_2 - f)}}{2\alpha - j4\pi^2 \beta_2 (f_1 - f)(f_2 - f)} \right|^2 L_{eff}^{-2} \quad (2.2)$$

Next figure shows a contour plot of the FWM Efficiency Factor As it can be observed on previous picture, ρ is constant for the duple (f_1, f_2) that describes an hyperbola. Thus, it is very effective to convert coordinates system to hyperbolic coordinates, so that the numerical solution of the integral becomes faster. Furthermore, as it is observed before, the integration domain can be clipped off beyond a certain value of the duple (f_1, f_2) that satisfies a threshold previously predefined. Therefore, the time resolution speeds up. The threshold is set up after a trade-off process between integration speed and accuracy.

2.4 Coherent Accumulation Factor

The Coherent Accumulation Factor χ accounts for the coherent interference produced at each span and accumulated at the receiver side of the link. It is also called "phased-array factor" since it has a similar shape as the radiation pattern of a phased array antenna.

$$\chi(f_1, f_2, f) = \frac{\sin^2\left(2N_s\pi^2(f_1 - f)(f_2 - f)\beta_2 L_s\right)}{\sin^2\left(2\pi^2(f_1 - f)(f_2 - f)\beta_2 L_s\right)} \quad (2.3)$$

Next figure shows the shape of χ setting a single value of f_1 and f .

2.5 Hyperbolic Coordinates

As it has been said before, it is quite interesting to compute the integral on the frame of hyperbolic coordinates instead of Cartesian ones. The hyperbolic coordinates are defined as follows:

$$\begin{aligned} \nu_1 &= \sqrt{f_1 f_2} \\ \nu_2 &= -\frac{1}{2} \log_e \left(\frac{f_2}{f_1} \right) \\ f_1 &= \nu_1 e^{\nu_2} \\ f_2 &= \nu_1 e^{-\nu_2} \end{aligned}$$

Hence, the GNRF is rearranged substituting f_1 and f_2 by ν_1 and ν_2 .

$$\begin{aligned} G_{NLI}(f) &= \frac{32}{27} \gamma^2 L_{s,eff}^2 \int_0^{B_{WDM}} \rho(\nu_1, 0) \chi(\nu_1, 0) |2\nu_1| \\ &\quad \int_0^{\log_e(B_{WDM}/\nu_1)} \left[G_{WDM}(\nu_1 e^{\nu_2} + f) G_{WDM}(\nu_1 e^{-\nu_2} + f) \right. \\ &\quad G_{WDM}(2\nu_1 \cosh(\nu_2) + f) + \\ &\quad G_{WDM}(\nu_1 e^{\nu_2} + f) G_{WDM}(-\nu_1 e^{-\nu_2} + f) \\ &\quad G_{WDM}(2\nu_1 \sinh(\nu_2) + f) + \\ &\quad G_{WDM}(-\nu_1 e^{\nu_2} + f) G_{WDM}(\nu_1 e^{-\nu_2} + f) \\ &\quad G_{WDM}(-2\nu_1 \sinh(\nu_2) + f) + \\ &\quad G_{WDM}(-\nu_1 e^{\nu_2} + f) G_{WDM}(-\nu_1 e^{-\nu_2} + f) \\ &\quad \left. G_{WDM}(-2\nu_1 \cosh(\nu_2) + f) \right] d\nu_2 d\nu_1 \quad (2.4) \end{aligned}$$

Usually, we are only interested in the central channel since it is the worst one. In other words, the non-linearity effect is the greatest in this channel. Hence, the

2.5. HYPERBOLIC COORDINATES

GNRF at $f=0$, $G_{NLI}(0)$, is:

$$G_{NLI}(0) = \frac{64}{27} \gamma^2 N_s L_{s,eff}^2 \int_0^{B_{WDM}/2} \rho(\nu_1, 0) \chi(\nu_1, 0) |2\nu_1| \int_0^{\log_e(B_{WDM}/2\nu_1)} \left[G_{WDM}(\nu_1 e^{\nu_2}) G_{WDM}(\nu_1 e^{-\nu_2}) \right. \\ \left. \left[G_{WDM}(2\nu_1 \cosh(\nu_2)) + G_{WDM}(2\nu_1 \sinh(\nu_2)) \right] d\nu_2 d\nu_1 \quad (2.5)$$

where:

$$\rho(\nu_1, 0) = \left| \frac{1 - e^{-2\alpha L_s} e^{j4\pi^2 \beta_2 L_s \nu_1^2}}{2\alpha - j4\pi^2 \beta_2 \nu_1^2} \right|^2 L_{eff}^{-2} \quad (2.6)$$

$$\chi(\nu_1, 0) = \frac{\sin^2(2N_s \pi^2 \nu_1^2 \beta_2 L_s)}{\sin^2(2\pi^2 \nu_1^2 \beta_2 L_s)} \quad (2.7)$$

The hyperbolic transformation shows that most of the contribution of ρ is concentrated around $\nu_1 = 0$. Hence, the hyperbolic domain of integration can be clipped-off, so that the numerical solution is faster and the accuracy remains. Figure 2.2 illustrates this fact:

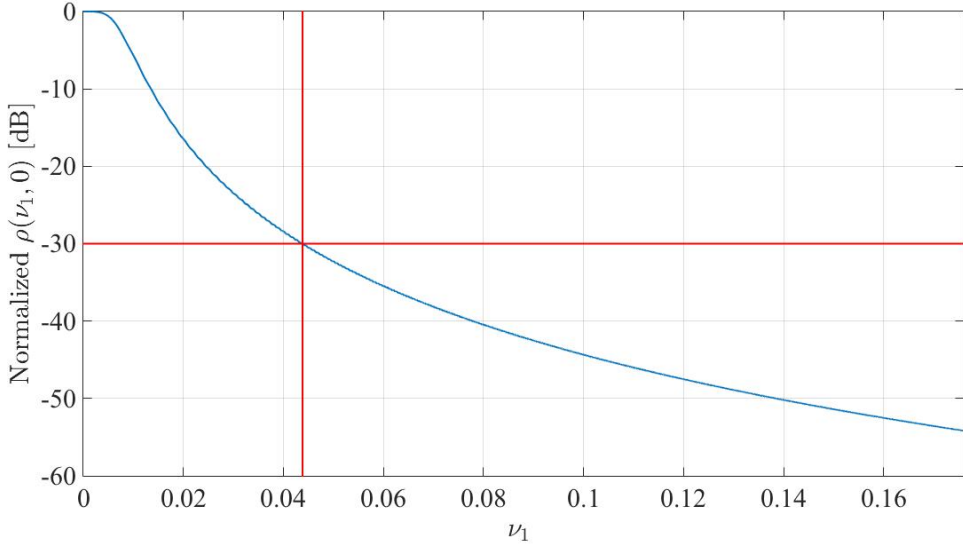


Figure 2.2: Clipping off the integration beyond -30 dB makes the integration domain 75% smaller

On the other hand, the distance between two consecutive peaks of χ increases as ν_1 increases. Then, a lot of samples of ν_1 are needed so that all peaks are taken into account. Therefore, another advantage of setting a threshold is that a reasonable

2.6. GN-MODEL APPROXIMATED FORMULA

number of samples are enough to represent χ properly. Figure 2.3 depicts χ when the domain of ν_1 is cut-off.

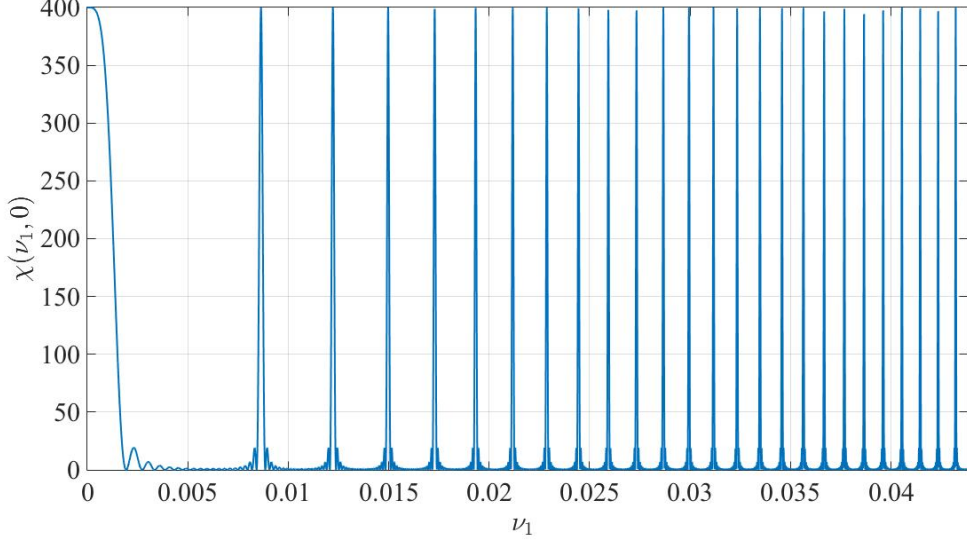


Figure 2.3: $\chi(\nu_1, 0)$ and threshold equal to -30 dB

2.6 GN-Model Approximated Formula

The GNRF(f) allows to compute the non-linear PSD at any frequency, in other words, the channel under test can be tuned easily. However, an approximated formula can be derived so that it substitutes the integral for $G_{NLI}(0)$ (Appendix F and G of [12]). At the central frequency the integration area is symmetric with respect to $f_1 + f_2 - f$ axis. Therefore, it is assumed to have a circular-like shape. This new integration domain helps to solve the integral analytically, however, it overestimates the non-linear PSD since the new domain is higher than the real domain. The closed-formula for the $f=0$ is:

$$G_{NLI} \approx \frac{8}{27} \frac{\gamma^2 G_{WDM}^3 L_{eff}^2}{\pi \beta_2 L_{eff,a}} \operatorname{asinh} \left(\frac{\pi^2}{2} \beta_2 L_{eff,a} B_{ch}^2 [N_{ch}^2]^{\frac{B_{ch}}{\Delta f}} \right) \quad (2.8)$$

The approximation works properly inside these boundaries:

1. $\beta_2 \geq 4 \left[\frac{ps^2}{km} \right]$
2. $R_s \geq 28 \text{ [GBaud/s]}$
3. $\frac{B_{ch}}{\Delta f} \geq 0.25$
4. $Spanloss \geq 7 \text{ [dB]}$

2.7 NLI Accumulation

This section shows some simulations that summarize how the evolution of NLI PSD changes vs either number of channels or number of spans. First of all, the evolution against the number of spans is analyzed. Function $g_{nli}(N_s)$ (2.9) considers the evolution of NLI PSD with respect to NLI PSD at central frequency after one span.

$$g_{nli}(N_s) = \frac{G_{nli}(0, N_s)}{G_{nli}(0, 1)} \quad (2.9)$$

The system under analysis consists on 21 channels at 32 GBaud/s. The optical fiber used was SMF, the roll-off factor was 0.3 and different frequency spacing values were compared. Figure 2.4 shows NLI PSD evolution for 50 GHz, 38.4 GHz, 32 GHz and incoherent accumulation. The smaller the frequency spacing, the more incoherent is the NLI accumulation, as reported on the figure below. The results are found through numerical integration of GNRF (2.1). Although others roll-off factors have been also analyzed for 50 GHz, 38.4 GHz and 32 GHz, no significant different results have been observed. The NLI accumulation against number of channel is studied as well. To do so, function $g_{nli}(N_{ch})$ (2.10) is defined. The scenario under analysis is SMF, 50 GHz spacing and 32 GBaud/s channels with 0.3 roll-off factor. Function $g_{nli}(N_{ch})$ is normalized to 95 channels (full C-Band).

$$g_{nli}(N_{ch}) = \frac{G_{nli}(0, N_s)}{G_{nli}(0, 95)} \quad (2.10)$$

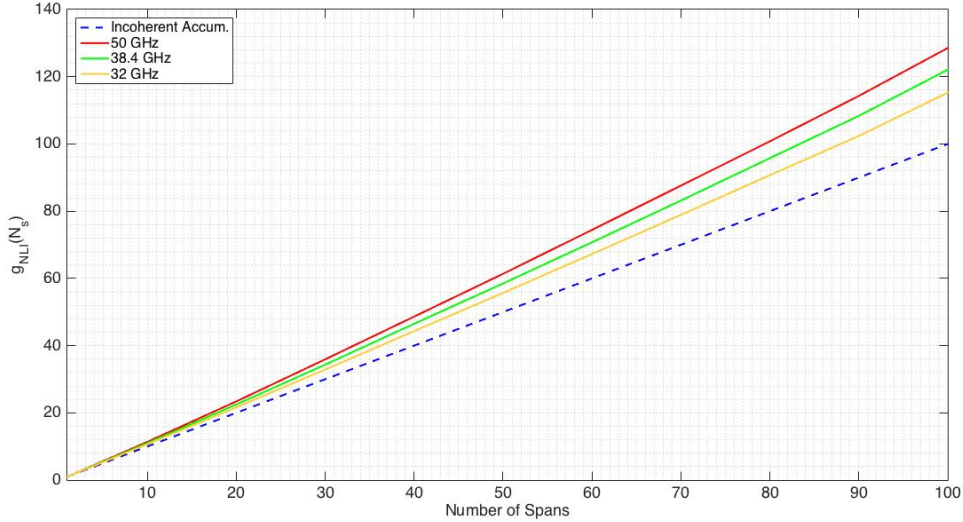


Figure 2.4: Red solid line 50 GHz frequency spacing, green solid line 38.4 GHz and yellow solid line 32 GHz

The evolution of NLI PSD grows following a logarithm law as the number of channels grows as well. The accumulation profile has been analyzed after one span,

2.7. NLI ACCUMULATION

twenty spans and fifty spans. The greater the number of spans, the faster the accumulation function grows. In other words, if the system is large enough, the receiver just need an optical bandwidth 3-4 channels wide to reduce 50% of the total NLI generated. On next figure 2.5, it is observed this important behavior.

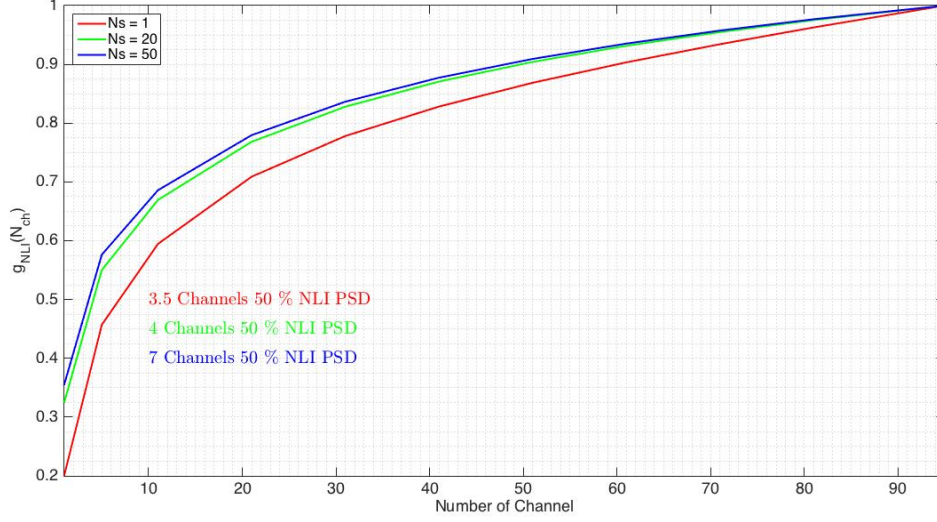


Figure 2.5: Blue solid line $N_s = 50$, green solid line $N_s = 20$ and red line $N_s = 1$

The experimental results available [12] states that the NLI noise accumulation follows a logarithm law (2.11).

$$G_{NLI}(f, N_s) = G_{NLI}(f, 1)N_s^{1+\epsilon} \quad (2.11)$$

The ϵ factor depends on many parameters and should be calculated for each system setup.

Chapter 3

Generalized Gaussian Noise Model

In order to assess properly the performance of ultra-wideband optical networks (i.e. C+L systems), we need to consider spatial/frequency fiber parameters variations. To do so, GN-Model is remodeled assuming this dependence leading to the so-called Generalized Gaussian Model. All computations and assumptions are based on the GN-Model derivation [13]. However, some changes should be taken into account as loss/gain variation is not constant against frequency.

3.1 Generalized GN-Model Reference Formula

Here it is shown the mathematical derivation of the NLI PSD $G_{NLI}(f)$ when the propagation and attenuation parameter depended on frequency, $\beta(f)$ and $\alpha(f)$. Then, the Generalized Gaussian Noise Model (GGN-Model) Reference Formula is depicted (3.18). This formula was derived on [1]. On this case, not only frequency variation, but also spatial/frequency variation of loss/gain is analyzed. The main difference is that, on the mathematical derivation shown below, the spatial integral is solved as $\alpha(f)$ and $\beta(f)$ does not change with z . However, on GGN-Model this integral cannot be solved. Following [13], the model is re-cast starting from the NLSE.

$$\frac{\partial}{\partial z} E(z, f) = [g(z, f) - j\beta(f)]E(z, f) + Q_{NLI}(z, f) \quad (3.1)$$

where:

1. $g(z, f)$: Gain profile
2. $\beta(f)$: Propagation constant
3. $E(z, f)$ Electrical field
4. Q_{NLI} Kerr Effect term

Notice that when lumped-amplification is considered, $g(z, f) = -\alpha(f)$. Then, we introduce it into (3.1).

$$\frac{\partial}{\partial z} E(z, f) = [-\alpha(f) - j\beta(f)]E(z, f) + Q_{NLI}(z, f) \quad (3.2)$$

3.1. GENERALIZED GN-MODEL REFERENCE FORMULA

Where,

$$Q_{NLI}(z, f) = -j\gamma E(z, f) * E^*(z, -f) * E(z, f)$$

Since $Q_{NLI}(z, f)$ depends on $E(z, f)$ the equation can not be solved applying differential equation theory. We analyze the nature of $Q_{NLI}(z, f)$ at the input of the fiber.

$$\begin{aligned} Q_{NLI}(0, f) &= -j\gamma E(0, f) * E^*(0, -f) * E(0, f) \\ &= -j\gamma \int_{-\infty}^{\infty} E(0, f_1) E^*(0, -(f - f_1)) df_1 * E(0, f) \\ &= -j\gamma \iint_{-\infty}^{\infty} E(0, f_1) E^*(0, f_1 - f_2) E(0, f - f_2) df_1 df_2 \end{aligned} \quad (3.3)$$

Recall that the model defines the field as follows:

$$E(f) = \sqrt{f_o G_{TX}(f)} \sum_{n=-\infty}^{\infty} \xi_n \delta(f - n f_o) \quad (3.4)$$

We introduce (3.4) into $Q_{NLI}(0, f)$ definition

$$\begin{aligned} Q_{NLI}(0, f) &= -j\gamma \iint_{-\infty}^{\infty} \sqrt{f_o G_{TX}(f_1)} \sum_{n=-\infty}^{\infty} \xi_n \delta(f_1 - n f_o) \\ &\quad \sqrt{f_o G_{TX}(f_1 - f_2)} \sum_{k=-\infty}^{\infty} \xi_k^* \delta(f_1 - f_2 - k f_o) \\ &\quad \sqrt{f_o G_{TX}(f - f_2)} \sum_{m=-\infty}^{\infty} \xi_m \delta(f - f_2 - m f_o) \\ &= -j\gamma \sum_{n=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} \sqrt{f_o}^3 \sqrt{G_{TX}(n f_o) G_{TX}(k f_o) G_{TX}(m f_o)} \\ &\quad \xi_n \xi_k^* \xi_m \delta(f_1 - n f_o) \delta(f_1 - f_2 - k f_o) \delta(f - f_2 - m f_o) \\ &= -j\gamma \sum_{n=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} \sqrt{f_o}^3 \sqrt{G_{TX}(n f_o) G_{TX}(k f_o) G_{TX}(m f_o)} \\ &\quad \xi_n \xi_k^* \xi_m \delta(-f_2 - (k - n) f_o) \delta(f - f_2 - m f_o) \\ &= -j\gamma \sum_{n=-\infty}^{\infty} \sum_{k=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} \sqrt{f_o}^3 \sqrt{G_{TX}(n f_o) G_{TX}(k f_o) G_{TX}(m f_o)} \\ &\quad \xi_n \xi_k^* \xi_m \delta(f - (n - k + m) f_o) \end{aligned}$$

We define the set A_i as the tuple (n, k, m) such that $n - k + m = i$. Furthermore, the subset X_i such that $n - k + m = i$ and $n = k$ or $m = k$. Therefore, $A_i = \tilde{A}_i + X_i$.

$$Q_{NLI}(0, f) = -j\gamma f_o^{3/2} \sum_{i=-\infty}^{\infty} \delta(f - i f_o) \sum_{n, k, m \in A_i} \xi_n \xi_k^* \xi_m \sqrt{G_{TX}(n f_o) G_{TX}(k f_o) G_{TX}(m f_o)}$$

3.1. GENERALIZED GN-MODEL REFERENCE FORMULA

$$Q_{NLI}(0, f) = Q_{NLI, \tilde{A}_i}(0, f) + Q_{NLI, X_i}(0, f)$$

$$Q_{NLI, \tilde{A}_i}(0, f) = -j\gamma f o^{3/2} \sum_{i=-\infty}^{\infty} \delta(f - i f_o) \sum_{n, k, m \in \tilde{A}_i} \xi_n \xi_k^* \xi_m \sqrt{G_{TX}(n f_o) G_{TX}(k f_o) G_{TX}(m f_o)}$$

$$Q_{NLI, X_i}(0, f) = -j\gamma f o^{3/2} \sum_{i=-\infty}^{\infty} \delta(f - i f_o) \sum_{n, k, m \in X_i} \xi_n \xi_k^* \xi_m \sqrt{G_{TX}(n f_o) G_{TX}(k f_o) G_{TX}(m f_o)}$$

We analyze the subset X_i :

1. $n-k+m=i$ and $n=k \rightarrow n-n+m=i \rightarrow m=i$
2. $n-k+m=i$ and $m=k \rightarrow n-m+m=i \rightarrow n=i$

$$\begin{aligned} Q_{NLI, X_i}(0, f) &= -j\gamma f o^{3/2} \sum_{i=-\infty}^{\infty} \delta(f - i f_o) \\ &\quad \left(\sum_k \sum_k \sum_i \xi_k \xi_k^* \xi_i \sqrt{G_{TX}(k f_o) G_{TX}(k f_o) G_{TX}(i f_o)} \right. \\ &\quad \left. + \sum_i \sum_k \sum_k \xi_i \xi_k^* \xi_k \sqrt{G_{TX}(i f_o) G_{TX}(k f_o) G_{TX}(k f_o)} \right) \\ &= -j\gamma f o^{3/2} \sum_{i=-\infty}^{\infty} \delta(f - i f_o) \\ &\quad \left(2 \sum_k \sum_i |\xi_k|^2 \xi_i G_{TX}(k f_o) \sqrt{G_{TX}(i f_o)} \right) \end{aligned} \quad (3.5)$$

The power of a single realization of the random process is $\tilde{P}_E = f_o \sum_{n=-\infty}^{\infty} G_{TX}(n f_o) |\xi_n|^2$ and it is also demonstrated that $\tilde{P}_E = P_{TX}$. We rearrange the expression:

$$\begin{aligned} Q_{NLI, X_i}(0, f) &= -j2\gamma P_{TX} \sqrt{f_o} \sum_{i=-\infty}^{\infty} \delta(f - i f_o) \sqrt{G_{TX}(i f_o)} \\ &= -j2\gamma P_{TX} E(0, f) \end{aligned} \quad (3.6)$$

We assume that this expression is still valid for any value of z .

$$Q_{NLI, X_i}(z, f) = -j2\gamma P_{TX} e^{-2\alpha(f)} \sqrt{f_o} \sum_{i=-\infty}^{\infty} \delta(f - i f_o) \sqrt{G_{TX}(i f_o)}$$

Recall that:

$$E(f) = \sqrt{f_o G_{TX}(f)} \sum_{n=-\infty}^{\infty} \xi_n \delta(f - n f_o)$$

3.1. GENERALIZED GN-MODEL REFERENCE FORMULA

Then,

$$Q_{NLI}(z, f) = -j2\gamma P_{TX} e^{-2\alpha(f)z} E(f) + Q_{NLI, \tilde{A}_i}$$

Therefore, the NLSE becomes:

$$\frac{\partial}{\partial z} E(z, f) = [-\alpha(f) - j\beta(f) - j2\gamma P_{TX} e^{-2\alpha(f)z}] E(z, f) + Q_{NLI, \tilde{A}_i}(z, f) \quad (3.7)$$

Now, the model makes an assumption that allows to obtain a closed solution of the Eq. $Q_{NLI, \tilde{A}_i}(z, f)$ is considered independent from $E(z, f)$. In other words, the non-linearity interference is generated from a different source. Recalling the theory for first order linear differential equation:

$$\frac{dy}{dx} + p(x)y = q(x) \quad (3.8)$$

The solution is:

$$y(x) = e^{-\int_{x_0}^x p(s)ds} \left[y_0 + \int_{x_0}^x q(s) e^{\int_{x_0}^s p(t)dt} ds \right] \quad (3.9)$$

Where:

$$y_0 = y(0) \quad ; \quad x_0 \rightarrow x = 0$$

Now we rearrange (3.7) to compare with (3.8).

$$\frac{\partial}{\partial z} E(z, f) + [\alpha(f) + j\beta(f) + j2\gamma P_{TX} e^{-2\alpha(f)z}] E(z, f) = Q_{NLI, \tilde{A}_i}(z, f)$$

We define function $\Gamma(z, f)$ as:

$$\begin{aligned} \Gamma(z, f) &= - \int_0^z [\alpha(f) + j\beta(f) + j2\gamma P_{TX} e^{-2\alpha(f)z'}] dz' \\ &= -\alpha(f)z - j\beta(f)z - j2\gamma P_{TX} z_{eff}(z, f) \end{aligned}$$

Where the effective length varies with both frequency and length as it is defined:

$$z_{eff}(z, f) = \frac{1 - e^{-2\alpha(f)z}}{2\alpha(f)} \quad (3.10)$$

$$E(z, f) = e^{\Gamma(z, f)} \left[E(0, f) + \int_0^z Q_{NLI, \tilde{A}_i}(z', f) e^{-\Gamma(z', f)} dz' \right]$$

$$E(z, f) \approx E_{LIN}(z, f) + E_{NLI}(z, f)$$

3.1. GENERALIZED GN-MODEL REFERENCE FORMULA

$$E_{LIN}(z, f) = e^{\Gamma(z, f)} E(0, f)$$

$$E_{NLI} = e^{\Gamma(z, f)} \int_0^z Q_{NLI, \tilde{A}_i}(z', f) e^{-\Gamma(z', f)} dz'$$

The model considers the non-linearity noise-like signal as perturbation. In other words, it is much lower than the transmitted signal. Then, $E(z, f) \approx E_{LIN}(z, f)$. Therefore,

$$Q_{NLI}(z, f) \approx -j\gamma E_{LIN}(z, f) * E_{LIN}^*(z, -f) * E_{LIN}(z, f)$$

$$E_{LIN}(z, f) = e^{\Gamma(z, f)} E(0, f)$$

$$Q_{NLI}(z, f) \approx -j\gamma \left[\int_{-\infty}^{\infty} e^{\Gamma(z, f_1)} E(0, f_1) e^{\Gamma^*(z, f_1 - f)} E^*(0, f_1 - f) df_1 \right] * E_{LIN}(z, f)$$

$$\begin{aligned} Q_{NLI}(z, f) &\approx -j\gamma \left[\int_{-\infty}^{\infty} e^{[-j\beta(f_1)z - \alpha(f_1)z - j2\gamma P_{TX} z_{eff}(z, f_1)]} \right. \\ &\quad \left. E(0, f_1) e^{[j\beta(f_1 - f)z - \alpha(f_1 - f)z + j2\gamma P_{TX} z_{eff}(z, f_1 - f)]} E^*(0, f_1 - f) df_1 \right] * E_{LIN}(z, f) \\ &= -j\gamma \left[\iint_{-\infty}^{\infty} e^{[-j\beta(f_1)z - \alpha(f_1)z - j2\gamma P_{TX} z_{eff}(z, f_1)]} \right. \\ &\quad \left. E(0, f_1) e^{[j\beta(f_1 - f_2)z - \alpha(f_1 - f_2)z + j2\gamma P_{TX} z_{eff}(z, f_1 - f_2)]} E^*(0, f_1 - f_2) \right. \\ &\quad \left. e^{-j\beta(f - f_2)z - \alpha(f - f_2)z - j2\gamma P_{TX} z_{eff}(z, f - f_2)} E(0, f - f_2) df_1 df_2 \right] \end{aligned} \quad (3.11)$$

We define a function $A(z, f_1, f_2, f)$

$$\begin{aligned} A(z, f_1, f_2, f) &= -j(\beta(f_1) - \beta(f_1 - f_2) + \beta(f - f_2))z \\ &\quad -(\alpha(f_1) + \alpha(f_1 - f_2) + \alpha(f - f_2))z \\ &\quad -j2\gamma P_{TX}(z_{eff}(z, f_1) - z_{eff}(z, f_1 - f_2) + z_{eff}(z, f - f_2)) \end{aligned}$$

$$Q_{NLI}(z, f) \approx -j\gamma \iint_{-\infty}^{\infty} e^{A(z, f_1, f_2, f)} E(0, f_1) E^*(0, f_1 - f_2) E(0, f - f_2) df_1 df_2$$

Introducing (3.4) into previous expression:

$$\begin{aligned} Q_{NLI}(z, f) &\approx -j\gamma \iint_{-\infty}^{\infty} e^{A(z, f_1, f_2, f)} f_o^{3/2} \sqrt{G_{TX}(f_1) G_{TX}(f_1 - f_2) G_{TX}(f - f_2)} \\ &\quad \sum_n \sum_k \sum_m \xi_n \xi_k^* \xi_m \delta(f_1 - n f_o) \delta(f_1 - f_2 - k f_o) \delta(f - f_2 - m f_o) df_1 df_2 \end{aligned}$$

3.1. GENERALIZED GN-MODEL REFERENCE FORMULA

Integral solution is straightforward.

$$Q_{NLI}(z, f) \approx -j\gamma f_o^{3/2} \sum_n \sum_k \sum_m \xi_n \xi_k^* \xi_m e^{A(z, nf_o, kf_o, mf_o)} \sqrt{G_{TX}(nf_o)G_{TX}(kf_o)G_{TX}(mf_o)} \delta(f - [n - k + m]f_o)$$

Recalling that the set A_i is composed of $n-k+m=i$ and \tilde{A}_i is that set such that $\tilde{A}_i = A_i - X_i$

$$Q_{NLI, \tilde{A}_i}(z, f) \approx -j\gamma f_o^{3/2} \sum_i \delta(f - if_o) \sum_{n, k, m \in \tilde{A}_i} \xi_n \xi_k^* \xi_m e^{A(z, nf_o, kf_o, mf_o)} \sqrt{G_{TX}(nf_o)G_{TX}(kf_o)G_{TX}(mf_o)} \quad (3.12)$$

Now we focus on $E_{NLI}(z, f)$ so that we can compute the PSD of the non-linearity.

$$E_{NLI}(z, f) = e^{\Gamma(z, f)} \int_0^z Q_{NLI, \tilde{A}_i}(z', f) e^{-\Gamma(z', f)} dz' \quad (3.13)$$

(3.12) is introduced into (3.13):

$$E_{NLI} = -j\gamma f_o^{3/2} e^{-j\beta(f)z} e^{-\alpha(f)z} e^{-j2\gamma P_{TX} z_{eff}(z, f)} \sum_i \delta(f - if_o) \sum_{n, k, m \in \tilde{A}_i} \sqrt{G_{TX}(nf_o)G_{TX}(kf_o)G_{TX}(mf_o)} \int_0^z e^{j\beta(f)z'} e^{\alpha(f)z'} e^{j2\gamma P_{TX} z_{eff}(z', f)} e^{A(z', f_1, f_2, f)} dz' \quad (3.14)$$

The integral is quite tedious because of $z_{eff}(z, f)$ changes with frequency since it depends on $\alpha(f)$. Then, it is more convenient to use an equivalent expression of Eq (26) where we do not split the set A_i into \tilde{A}_i and X_i . The equivalent expression of the non-linear field is:

$$E_{NLI}(z, f) \approx \sum_i \delta(f - if_o) \left[-j\gamma f_o^{3/2} e^{\left[-j\beta(f) - \alpha(f)\right]z} \sum_{m, n, k \in A_i} \xi_n \xi_k^* \xi_m \sqrt{G_{TX}(nf_o)G_{TX}(kf_o)G_{TX}(mf_o)} \int_0^z e^{j\Delta\beta z'} e^{\Delta\alpha z'} dz' \right]$$

Where,

$$\begin{aligned} \Delta\beta &= \beta([n - k + m]f_o) - \beta(nf_o) + \beta(kf_o) - \beta(mf_o) \\ \Delta\alpha &= \alpha([n - k + m]f_o) - \alpha(nf_o) - \alpha(kf_o) - \alpha(mf_o) \end{aligned}$$

The expression is the same, but it is quite easier to integrate. Previously, the factor z_{eff} appeared on $\Gamma(z, f)$ and only the subset \tilde{A}_i affected E_{NLI} since the subset

3.1. GENERALIZED GN-MODEL REFERENCE FORMULA

X_i only added a term proportional to $E(z, f)$. This division is a good choice for GN-Model because it considers constant parameters, so that z_{eff} is constant for all frequency components and it is canceled inside Q_{NLI, \tilde{A}_i} , not increasing the integral complexity. We solve the integral,

$$\begin{aligned} \int_0^z e^{(j\Delta\beta + \Delta\alpha)z'} dz' &= \frac{1}{j\Delta\beta + \Delta\alpha} \int_0^z (j\Delta\beta + \Delta\alpha) e^{(j\Delta\beta + \Delta\alpha)z'} dz' \\ &= \frac{1}{j\Delta\beta + \Delta\alpha} \left[e^{(j\Delta\beta + \Delta\alpha)z} - 1 \right] \\ &= \frac{1 - e^{(j\Delta\beta + \Delta\alpha)z}}{-\Delta\alpha - j\Delta\beta} \end{aligned}$$

$$E_{NLI}(z, f) = \sum_i \mu_i \delta(f - if_o) \quad (3.15)$$

The NLI disturbance is a set of deltas. The power spectral density of a given instance of such process would be:

$$\Theta_{E_{NLI}}(f) = \sum_i |\mu_i|^2 \delta(f - if_o) \quad (3.16)$$

We use the statistical expectation operator to compute the average PSD.

$$\begin{aligned} G_{E_{NLI}}(f) &= E\{\Theta_{E_{NLI}}(f)\} \\ &= \sum_i E\{|\mu_i|^2\} \delta(f - if_o) \end{aligned}$$

We focus on $|\mu_i|^2 = \mu_i \mu_i^*$

$$\begin{aligned} |\mu_i|^2 &= \left[-j\gamma f_o^{3/2} e^{[-j\beta(f) - \alpha(f)]z} \sum_{n,k,m \in A_i} \xi_n \xi_k^* \xi_m \right. \\ &\quad \left. \frac{\sqrt{G_{TX}(nf_o)G_{TX}(kf_o)G_{TX}(mf_o)}}{-\Delta\alpha - j\Delta\beta} \right] \left[j\gamma f_o^{3/2} e^{[j\beta(f) - \alpha(f)]z} \right. \\ &\quad \left. \sum_{n',k',m' \in A_i} \xi_n^* \xi_k \xi_m^* \frac{\sqrt{G_{TX}(n'f_o)G_{TX}(k'f_o)G_{TX}(m'f_o)}}{-\Delta\alpha + j\Delta\beta} \right] \end{aligned}$$

$$\begin{aligned} |\mu_i|^2 &= \gamma^2 f_o^3 e^{-2\alpha(f)z} \sum_{n,k,m \in A_i} \sum_{n',k',m' \in A_i} \xi_n \xi_n^* \xi_k^* \xi_k' \xi_m \xi_m^* \\ &\quad \frac{\sqrt{G_{TX}(nf_o)G_{TX}(kf_o)G_{TX}(mf_o)} \sqrt{G_{TX}(n'f_o)G_{TX}(k'f_o)G_{TX}(m'f_o)}}{\left| \frac{1 - e^{(j\Delta\beta + \Delta\alpha)z}}{-\Delta\alpha - j\Delta\beta} \right|^2} \end{aligned}$$

3.1. GENERALIZED GN-MODEL REFERENCE FORMULA

$$\begin{aligned}
E\{|\mu_i|^2\} &= \gamma^2 f_o^3 e^{-2\alpha(f)z} \sum_{n,k,m \in A_i} \sum_{n',k',m' \in A_i} E\{\xi_n \xi_n^{I*} \xi_k^* \xi_k' \xi_m \xi_m^{I*}\} \\
&\quad \sqrt{G_{TX}(nf_o)G_{TX}(kf_o)G_{TX}(mf_o)} \sqrt{G_{TX}(n'f_o)G_{TX}(k'f_o)G_{TX}(m'f_o)} \\
&\quad \left| \frac{1 - e^{(j\Delta\beta + \Delta\alpha)z}}{-\Delta\alpha - j\Delta\beta} \right|^2 \quad (3.17)
\end{aligned}$$

It is demonstrated [13] that:

$$E\{\xi_n \xi_n^{I*} \xi_k^* \xi_k' \xi_m \xi_m^{I*}\} = E\{|\xi_n|^2\} E\{|\xi_k|^2\} E\{|\xi_m|^2\} = 1$$

Following the procedure in [13], an approximation of (3.17) is given:

$$E\{|\mu_i|^2\} \approx 2\gamma^2 f_o^3 e^{-2\alpha(f)z} \sum_n \sum_m G_{TX}(nf_o) G_{TX}(mf_o) G_{TX}([n+m-i]f_o)$$

$$\left| \frac{1 - e^{\{\alpha(if_o) - \alpha(nf_o) - \alpha([n+m-i]f_o) - \alpha(mf_o)\}z} e^{j\{\beta(if_o) - \beta(nf_o) + \beta([n+m-i]f_o) - \beta(mf_o)\}z}}{-\alpha(if_o) + \alpha(nf_o) + \alpha([n+m-i]f_o) + \alpha(mf_o) - j\{\beta(if_o) - \beta(nf_o) + \beta([n+m-i]f_o) - \beta(mf_o)\}} \right|$$

Therefore, the PSD of the non-linearity interference for the Single Polarization case.

$$G_{ENLI}(f) = 2\gamma^2 f_o^3 e^{-2\alpha(f)z} \sum_i \delta(f - if_o) \sum_n \sum_m G_{TX}(nf_o) G_{TX}(mf_o) G_{TX}([n+m-i]f_o)$$

$$\left| \frac{1 - e^{\{\alpha(if_o) - \alpha(nf_o) - \alpha([n+m-i]f_o) - \alpha(mf_o)\}z} e^{j\{\beta(if_o) - \beta(nf_o) + \beta([n+m-i]f_o) - \beta(mf_o)\}z}}{-\alpha(if_o) + \alpha(nf_o) + \alpha([n+m-i]f_o) + \alpha(mf_o) - j\{\beta(if_o) - \beta(nf_o) + \beta([n+m-i]f_o) - \beta(mf_o)\}} \right|$$

The expansion for Dual-Polarization case is explained in [13].

$$G_{ENLI}(f) = \frac{16}{27} \gamma^2 f_o^3 e^{-2\alpha(f)z} \sum_i \delta(f - if_o) \sum_n \sum_m G_{TX}(nf_o) G_{TX}(mf_o) G_{TX}([n+m-i]f_o)$$

$$\left| \frac{1 - e^{\{\alpha(if_o) - \alpha(nf_o) - \alpha([n+m-i]f_o) - \alpha(mf_o)\}z} e^{j\{\beta(if_o) - \beta(nf_o) + \beta([n+m-i]f_o) - \beta(mf_o)\}z}}{-\alpha(if_o) + \alpha(nf_o) + \alpha([n+m-i]f_o) + \alpha(mf_o) - j\{\beta(if_o) - \beta(nf_o) + \beta([n+m-i]f_o) - \beta(mf_o)\}} \right|$$

The transition to a frequency-continuous domain:

$$\begin{aligned}
G_{ENLI}(f) &= \frac{16}{27} \gamma^2 e^{-2\alpha(f)z} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} G_{TX}(f_1) G_{TX}(f_2) G_{TX}(f_1 + f_2 - f) \\
&\quad \left| \frac{1 - e^{\{\alpha(f) - \alpha(f_1) - \alpha(f_1 + f_2 - f) - \alpha(f_2)\}z} e^{j\{\beta(f) - \beta(f_1) + \beta(f_1 + f_2 - f) - \beta(f_2)\}z}}{-\alpha(f) + \alpha(f_1) + \alpha(f_1 + f_2 - f) + \alpha(f_2) - j\{\beta(f) - \beta(f_1) + \beta(f_1 + f_2 - f) - \beta(f_2)\}} \right|
\end{aligned}$$

The GNRF of GGN-Model is depicted above [1].

$$\begin{aligned}
G_{NLI}(f) &= \frac{16}{27} \gamma^2 \rho(z, f)^2 \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} G_{TX}(f_1) G_{TX}(f_2) G_{TX}(f_1 + f_2 - f) \\
&\quad \left| \int_0^z e^{j4\pi^2(f_1 - f)(f_2 - f)\beta_2 \zeta} \frac{\rho(\zeta, f_1) \rho(\zeta, f_3) \rho(\zeta, f_2)}{\rho(\zeta, f)} d\zeta \right|^2 df_1 df_2 \quad (3.18)
\end{aligned}$$

3.2 FWM Efficiency Factor

One of the difference between GN-Model and GGN-Model is how the FWM Efficiency Factor is addressed. On the GN-Model, the spatial integral is easily solved as the parameters remains constant along both space and frequency. However, it changes when the parameters are considered variable. Next expression shows the FWM Efficiency factor for GGN-Model.

$$\rho(z, f_1, f_2, f) = \left| \int_0^z e^{j4\pi^2(f_1-f)(f_2-f)\beta_2\zeta} \frac{\rho(\zeta, f_1)\rho(\zeta, f_3)\rho(\zeta, f_2)}{\rho(\zeta, f)} d\zeta \right|^2 \quad (3.19)$$

Recall that this factor for GN-Model follows the expression depicted in 2.2.

$$\rho(f_1, f_2, f) = \left| \frac{1 - e^{-2\alpha L_s} e^{j4\pi^2\beta_2 L_s (f_1-f)(f_2-f)}}{2\alpha - j4\pi^2\beta_2 (f_1-f)(f_2-f)} \right|^2 L_{eff}^{-2}$$

Clearly, the function shown above does not depend on z , as the integral was solved analytically. On the other hand, the integral 3.19 should be solved numerically. First of all, the function to be integrated is analyzed. It is a complex exponential damped by the field profile of all pumps frequencies. The factor $(f_1 - f)(f_2 - f)$ determines the slope of the complex exponential. In other words, the frequency of the sinusoidal functions derived from the complex exponential. This features determines the number of samples on z needed to solve the integral properly. Furthermore, this is a critical issue, as it determines the simulation time. For example, when f_1 and f_2 are close to the frequency under analysis f , then the sinusoidal functions are slow (Figure 3.1). Therefore, few nS are needed. However, the minimum nS increases as both f_1 and f_2 move away from f (Figure 3.2).

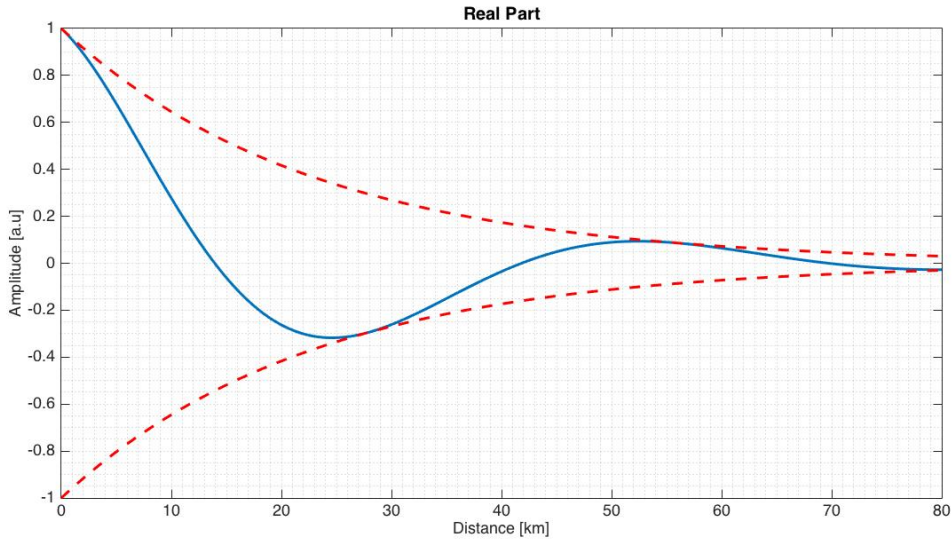


Figure 3.1: Blue solid line real part of the complex exponential and red dashed lines field profile of pumps frequencies, when $(f_1 - f)(f_2 - f)$ is low.

3.2. FWM EFFICIENCY FACTOR

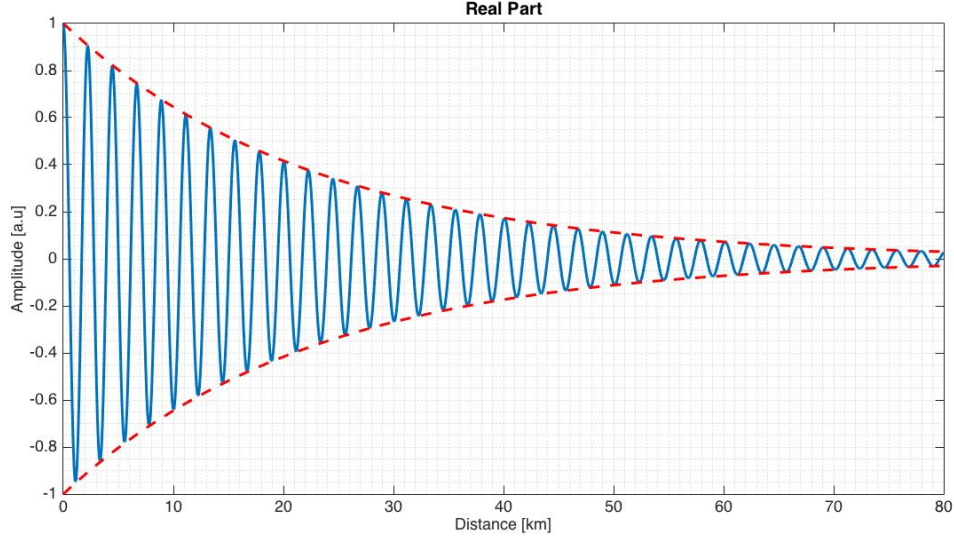


Figure 3.2: Blue solid line real part of the complex exponential and red dashed lines field profile of pumps frequencies, when $(f_1 - f)(f_2 - f)$ is high.

The reader can wonder what is the consequence of not determine properly the number of samples. The answer is clearly that the model provides an over/under estimate. Function $I_2^{GN}(f_1, f)$ is defined as:

$$I_2^{GN}(f_1, f) = G_{TX}(f_1) \int_{-B_w/2}^{B_w/2} G_{TX}(f_2) G_{TX}(f_1 + f_2 - f) \left| \int_0^z e^{j4\pi^2(f_1-f)(f_2-f)\beta_2\zeta} \frac{\rho(\zeta, f_1)\rho(\zeta, f)\rho(\zeta, f_2)}{\rho(\zeta, f_3)} d\zeta \right|^2 df_2 \quad (3.20)$$

Function $I_2^{GN}(f_1, f)$ is defined as:

$$I_2^{GN}(f_1, f) = G_{TX}(f_1) \int_{-B_w/2}^{B_w/2} G_{TX}(f_2) G_{TX}(f_1 + f_2 - f) \left| \frac{1 - e^{-2\alpha L_s e^{j4\pi^2\beta_2 L_s(f_1-f)(f_2-f)}}}{2\alpha - j4\pi^2\beta_2(f_1-f)(f_2-f)} \right|^2 df_2 \quad (3.21)$$

Later on, function $I_2(f_1, f)$ is integrated with respect to f_1 . Both functions should match for the case of constant parameters if only if nS is properly set. Figures 3.3 show the misleading result when nS is not correct.

3.2. FWM EFFICIENCY FACTOR

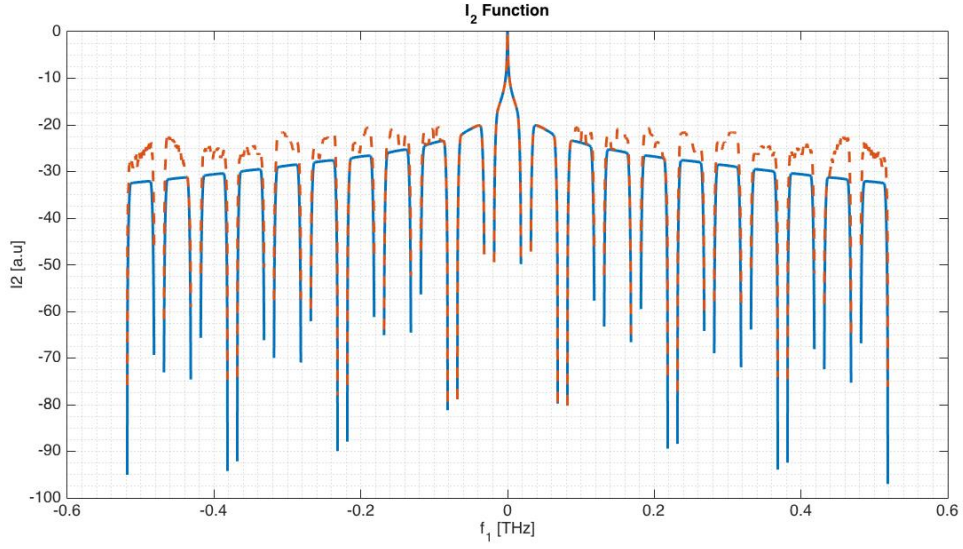


Figure 3.3: Blue solid line represents I_2^{GN} and red dashed line represents I_2^{GGN}

At first sight, it seems that the mismatching does not affect the results so much as it is 20 dB lower than the main lobe. However, it does as the number of channels increases. To sum up, it can be state that optimal number of sample is variable and depends on what pumps frequency are being analyzing.

Chapter 4

Metric to assess Quality of Transmission (QoT)

4.1 Metric definition and GN-Model implications

From the physical layer point of view, the most common metric to evaluate the QoT is the Signal-to-Noise Ratio (SNR) (4.1). In optics, however, the widely used parameter is the so called Optical-Signal-to-Noise Ratio (OSNR) (4.2). The difference is the reference bandwidth (Figure 4.1). In optical communications the reference bandwidth to measure the noise is typically around 12.48 GHz (0.1 nm). Thus, (4.1) is obtained after the post-processing stage and (4.2) is the "physical" value obtained by the Optical Spectrum Analyzer. Next expressions are taken from [14].

$$SNR = \frac{P_{ch}}{P_{ASE}} \quad (4.1)$$

$$OSNR = \frac{R_s}{B_n} SNR \quad (4.2)$$

Furthermore, under Linear Propagation and Gaussian Noise scenario the relation between Bit Error Rate (BER) and SNR is [14]:

$$BER = \frac{1}{2} \operatorname{erfc}\left(\sqrt{SNR/2}\right) \quad (4.3)$$

P_{ch} is the received channel power and P_{ASE} is the noise power produced by EDFA amplifiers.

$$P_{ch} = P_{RX} R_s^{-1} \int_{-\infty}^{\infty} |H_{RX}(f)|^2 df \quad (4.4)$$

$$P_{ASE} = \int_{-\infty}^{\infty} G_{ASE} |H_{RX}(f)|^2 df \quad (4.5)$$

4.1. METRIC DEFINITION AND GN-MODEL IMPLICATIONS

G_{ASE} is the bilateral power spectral density down-converted to baseband of dual polarization ASE noise impinging onto the Rx. At this point, one assumption of which the GN-Model was based on, is used. It considers the NLI effect as an additive Gaussian noise and independent of ASE noise. This is the most important implication of the GN-Model since it allows to use a more accurate SNR definition than Equation (4.1) as the metric that evaluates QoT. According to this, the SNR expression (4.1) is re-cast and (4.1) is renamed as SNR_{ASE} :

$$SNR = \frac{P_{ch}}{P_{ASE} + P_{NLI}} \quad (4.6)$$

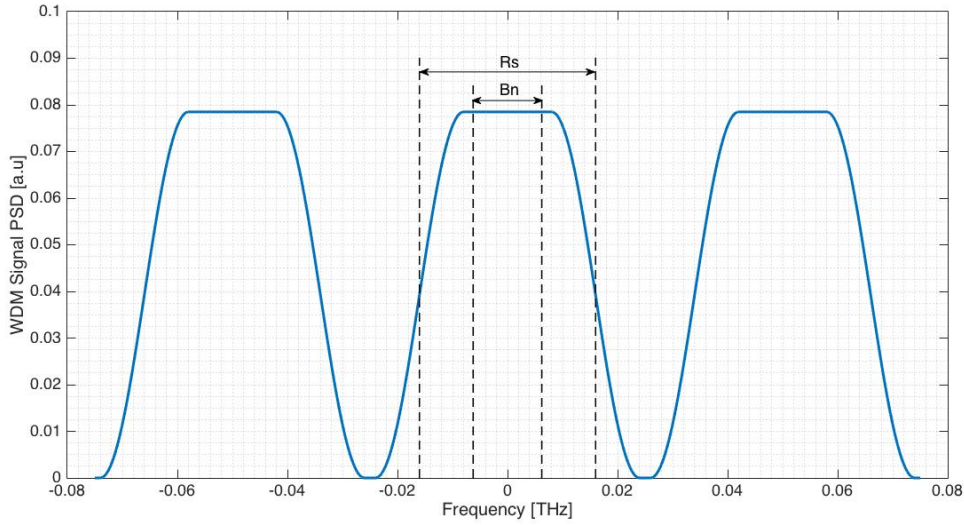


Figure 4.1: Reference Bandwidth for SNR (R_s) and for OSNR (B_n)

P_{NLI} is defined as follows:

$$P_{NLI} = \int_{-\infty}^{\infty} G_{NLI}(f) |H_{RX}(f)|^2 df \quad (4.7)$$

$G_{NLI}(f)$ is the bilateral down-converted power spectral density of the NLI perturbation. The objective of the GN-Model and all its updates is just to determine $G_{NLI}(f)$. As it was set forth above, the SNR parameter is a general ratio commonly used for all communications systems. However, the parameter available on the Optical Spectrum Analyzer (OSA) is the well-known OSNR. In other words, OSA only measures the "physical" noise, so that it cannot measure the noise-like NLI power as it is an in-band effect. Furthermore, (4.6) can be written as:

$$\frac{1}{SNR} = \frac{1}{OSNR} \frac{R_s}{B_n} + \frac{1}{SNR_{NLI}} \quad (4.8)$$

Where SNR_{NLI} is defined as :

$$SNR_{NLI} = \frac{P_{ch}}{P_{NLI}} \quad (4.9)$$

SNR_{NLI} should be measured in order to validate GN-Model. To do so, Error Vector Magnitude (EVM) is measured at the received constellation [6]. The difference between the theoretical point on the constellation and the received one is the error vector. In other words, how far are the received point to its ideal location. Figure 4.2 summarizes it. The variance of the EVM represents the noise power and the signal power is the signal vector distance to the square. Then, a relation between signal power and noise power can be established.

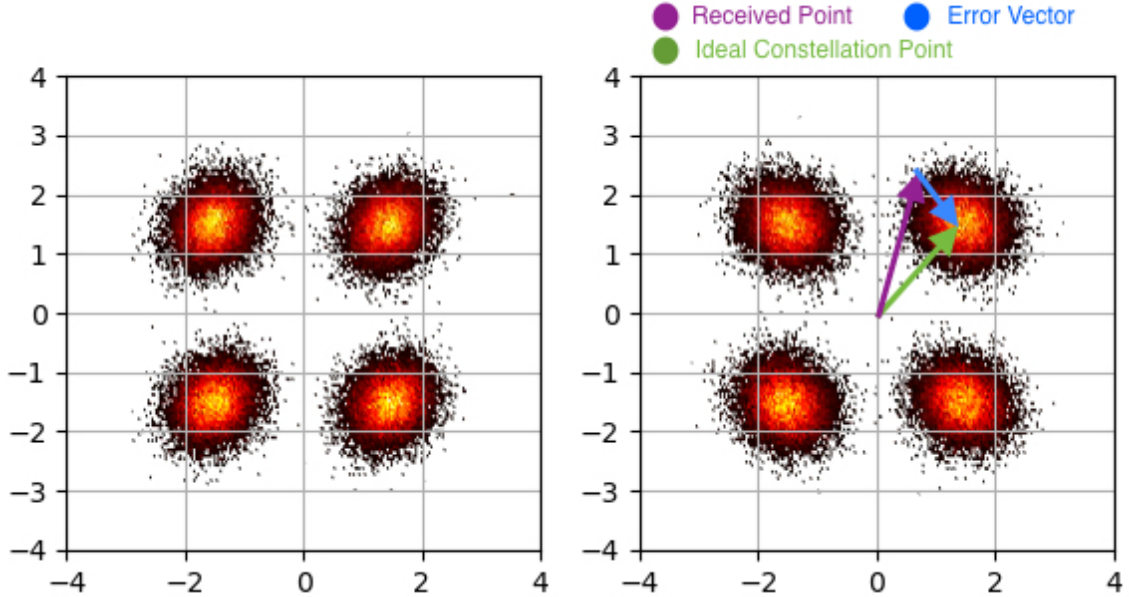


Figure 4.2: Random example of the I/Q diagram provided by the receiver

4.2 Maximum Length

One the most important application of the metric defined above is to estimate the maximum length, so that the system satisfies a QoT. Typically, a system requirement is the maximum BER accepted. This depends on a lot of issues: the kind of technology (3G, 4G, Passive Optical Network (PON), etc), the kind of data (Audiovisual data, voice data or text data), required quality etc. Therefore, the given BER determines a minimum SNR value at the receiver side (4.3). This value is named SNR_{sens} . The implications of the GN-Model let non-linear effects be taken into account in SNR_{sens} . Let compare linear and non-linear scenarios.

M^{lin} and M^{nli} are the maximum number of spans for the linear scenario and for the

4.2. MAXIMUM LENGTH

non-linear one, respectively.

$$\begin{aligned} SNR_{sens} &= \frac{P_{ch}}{M^{lin} P_{ase}} \\ M^{lin} &= \frac{1}{SNR_{sens}} \frac{P_{ch}}{P_{ase}} \end{aligned} \quad (4.10)$$

SNR_{sens} is in linear scale, P_{ch} is the input channel power and P_{ase} is the noise power induced by the amplifiers. Note that P_{ase} does not depend on P_{ch} . Therefore, the maximum length is infinite since M^{lin} grows as P_{ch} does. In other words, M^{lin} is a linear function of P_{ch} . On the other hand, for the non-linear scenario case the SNR definition changes as explained on previous section (4.6).

$$\begin{aligned} SNR_{sens} &= \frac{P_{ch}}{M^{nli} P_{ase} + M^{nli} P_{nli}} \\ SNR_{sens} &= \frac{P_{ch}}{M^{nli} P_{ase} + M^{nli} \eta^{(1)} P_{ch}} \\ M^{nli} &= \frac{1}{SNR_{sens}} \frac{P_{ch}}{P_{ase} + \eta^{(1)} P_{ch}^3} \end{aligned} \quad (4.11)$$

$\eta^{(1)}$ is a constant that relates P_{ch} and P_{nli} on the first span. In this case NLI accumulation is considered incoherent, then P_{nli} after N spans is $P_{nli} = N \eta^{(1)} P_{ch}^3$. Figure 4.3 illustrates the evolution of the maximum length for linear and non-linear scenario. The number of channels is 58 and the fiber is a common SMF with $\alpha = 0.22 \text{ dB/km}$.

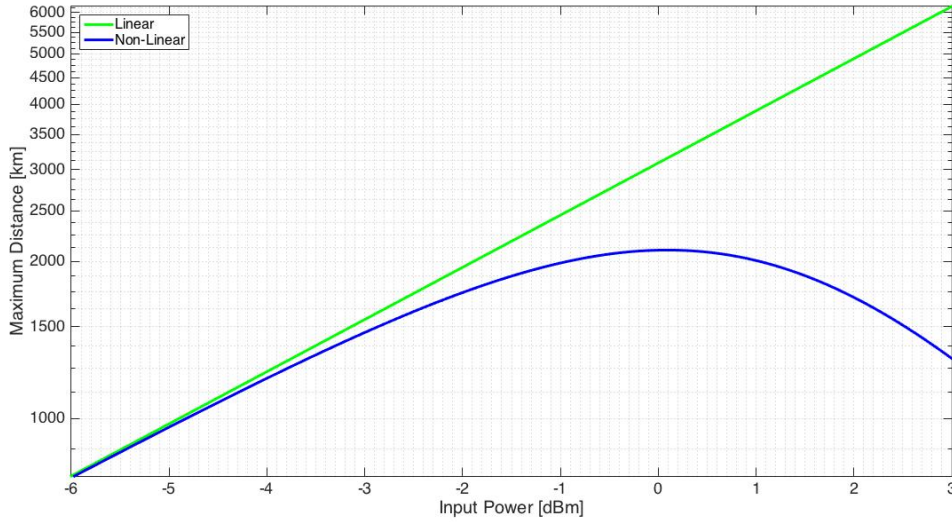


Figure 4.3: Green solid line represents the maximum length for a linear system given the input power and blue solid line does it for a non-linear system

Chapter 5

Model implementation

Both models have been implemented in order to carry out simulations that allow to estimate the system performance evaluation. Thus, the developed tool is useful to plan optical networks affected by non-linear effect. Moreover, the software tool implemented on this Master Thesis is also flexible as different models can be chosen depending on the accuracy level desired. For instance, if only preliminary analysis of NLI effect at central frequency is expected, it is highly recommendable to use GN-Model approximated formula (2.8) as the algorithm is executed very fast. If a more detailed analysis is expected and the number of channels is not large enough so that the fiber parameters change a lot (in C-Band attenuation frequency dependence is soft) GN-Model (2.1) can be used. On the other hand, if we are dealing with a ultra-wide-band optical system (involving C+L Bands) GGN-Model is the proper model to estimate NLI at any frequency. The code has been divided as follows: fiber parameters are defined on an independent Matlab file, system parameters are defined on another independent Matlab file and the GN-Model approximated formula (2.8), the GN-Model (2.1) and the GGN-Model (3.18) are also developed on an independent Matlab file each one.

5.1 Code Structure

On this section, the code structure is deeply explained so that others can use the tool. As it is said above, the main three parts are: fiber parameters definition, system parameters definition and model implementation.

5.1.1 Fiber Parameters

The fiber parameters are defined on the Matlab file named as "*fiberParam.m*". This file is an ad-hoc defined function. It uses two facilities available on the Matlab environment: struct data type and switch-case expression. The struct data type allows to group related parameters using data containers called *fields*. For our purpose, this way to organize the code is significantly useful since all fiber parameters are grouped together and all of them are related to a type of fiber. On the other

hand, the switch-case expression allows to select a fiber and hence all its parameters. The function input and output variables are listed above:

- Input
 - **fiberType** String Data Type. Used by switch-case structure to select the fiber.
- Output
 - **fiberParams** Struct Data Type. It contains fiber parameters.

The fields inside *fiberParams* struct are listed and explained below:

- **GVD** Group Velocity Dispersion $[\frac{ps^2}{km}]$.
- **alpha** Attenuation Coefficient. Electric field definition $[\frac{1}{km}]$.
- **gamma** Kerr Effect Coefficient. $[W^{-1}km^{-1}]$.

The procedure to define a new type of fiber is basically to open "*fiberParam.m*", to define a new case with the fiber type name and to fill its fields with the fiber parameters values. For example, next figure shows a SMF with D = to be filled, alpha = to be filled and gamma = to be filled.

5.1.2 System-Signal Parameters

Both signals and system parameters are defined in the same Matlab file named as "*systemParams.m*". This file is an ad-hoc defined function. It follows the same strategy as the one used for "*fiberParam.m*". The input and the variables of the function are listed below:

- Input
 - **delta_f** Double Data Type. Spacing frequency in $[THz]$.
 - **article** String Data Type. System name reference.
 - **nCh** Double Data Type. Number of Channels.
 - **power** Double Data Type. Channel Power in $[dBm]$.
- Output
 - **signal** Struct Data Type. It contains signal parameters.
 - **sys** Struct Data Type. It contains system parameters.

The fields inside both *signal* and *sys* structs are listed:

- **signal**
 - **roll_off** Double Data Type. Roll-off factor.
 - **Rs** Double Data Type. Symbol Rate in $[GBauds]$.
 - **power** Double Data Type. Channel Power in $[dBm]$.
 - **BwCh** Double Data Type. Channel Bandwidth in $[THz]$.

- **sys**
 - **Flin** Double Data Type. Figure Noise in linear scale.
 - **GLin** Double Data Type. EDFA Gain in linear scale.
 - **Ls** Double Data Type. Span length in $[km]$.
 - **nCh** Double Data Type. Number of Channels.
 - **delta_f** Double Data Type. Spacing frequency in $[THz]$.
 - **Bw** Double Data Type. Total Bandwidth in $[THz]$.
 - **Ns** Double Data Type. Number of Spans.
 - **fCh** Double Data Type. Central frequency of each channel in $[THz]$.
 - **fo** Double Data Type. Central frequency in $[THz]$

The procedure to define a new set of parameters in order to simulate a system is basically to open the file *"systemParams.m"*. Next figure a case of a system definition.

5.1.3 GN-Model Approximated Formula

The file name as *"GNLI_aprox.m"* computes the GN-Model based on the approximated formula defined on (2.8). The input and output variables are listed below:

- Input
 - **fiberParams** Struct Data Type. It contains fiber parameters.
 - **signal** Struct Data Type. It contains signal parameters.
 - **sys** Struct Data Type. It contains system parameters.
- Output
 - **gnli** Double Data Type. NLI PSD in $[\frac{W}{Hz}]$.

5.1.4 GN-Model Cartesian Coordinates

The file named as *"GNLI_f.m"* computes the GN-Model through numerical integration (2.1). The integral is solved on Cartesian coordinates (f_1, f_2) . The input and output variables are listed below:

- Input
 - **fiberParams** Struct Data Type. It contains fiber parameters.
 - **signal** Struct Data Type. It contains signal parameters.
 - **sys** Struct Data Type. It contains system parameters.
 - **f** Double Data Type. Frequency under analysis on base-band in $[THz]$.
 - **gtx** Double Array Data Type. WDM-Signal PSD in $[\frac{W}{Hz}]$.

- Output
 - **GNLI** Double Data Type. NLI PSD in $[\frac{W}{Hz}]$.

5.1.5 GN-Model Hyperbolic Coordinates

The file named as "*GNLI_f_hyp.m*" computes the GN-Model through numerical integration (2.4). The integral is solved on Hyperbolic coordinates (ν_1, ν_2) . The input and output variables are listed below:

- Input
 - **fiberParams** Struct Data Type. It contains fiber parameters.
 - **signal** Struct Data Type. It contains signal parameters.
 - **sys** Struct Data Type. It contains system parameters.
 - **f** Double Data Type. Frequency under analysis on base-band in $[THz]$.
 - **gtx** Double Array Data Type. WDM-Signal PSD in $[\frac{W}{Hz}]$.
- Output
 - **GNLI** Double Data Type. NLI PSD in $[\frac{W}{Hz}]$.

5.1.6 GN-Model Hyperbolic Coordinates for $f=0$

The file named as "*GNLI_0_hyp.m*" computes the GN-Model through numerical integration (2.5). The integral is solved on Hyperbolic coordinates (ν_1, ν_2) . The input and output variables are listed below:

- Input
 - **fiberParams** Struct Data Type. It contains fiber parameters.
 - **signal** Struct Data Type. It contains signal parameters.
 - **sys** Struct Data Type. It contains system parameters.
- Output
 - **GNLI** Double Data Type. NLI PSD in $[\frac{W}{Hz}]$.

5.1.7 GGN-Model Matrix Solution

The file named as "*GGNLI_f.m*" computes the GGN-Model through numerical integration (3.18). The Matrix Solution is explained on 5.2.2. The input and output variables are listed below:

- Input
 - **fiberParams** Struct Data Type. It contains fiber parameters.
 - **signal** Struct Data Type. It contains signal parameters.

- **sys** Struct Data Type. It contains system parameters.
- **f** Double Data Type. Frequency under analysis on base-band in $[THz]$.
- **fibProf** Double Matrix Type $[nchxzeta_samples]$. Power Evolution Profile.
- **sigPSD** Double Array Data Type. WDM-Signal PSD in $[\frac{W}{Hz}]$.
- **zeta_samples** Double Type. Number of samples for z. It should be the minimum to solve the worst case.

- Output

- **GGf** Double Array Data Type. NLI PSD in $[\frac{W}{Hz}]$.

5.1.8 GGN-Model Loop/Matrix Solution

The file named as "*GGNLI_f_2.m*" computes the GGN-Model through numerical integration (3.18). This function is almost equal than the previous one. The loop/-Matrix solution refers to how the spatial integral of GGN-Model (3.18) is solved. The difference with respect to Matrix Solution is just that f2 enters into a for loop instead of being computed on one go. The input and output variables are listed below:

- Input

- **fiberParams** Struct Data Type. It contains fiber parameters.
- **signal** Struct Data Type. It contains signal parameters.
- **sys** Struct Data Type. It contains system parameters.
- **f** Double Data Type. Frequency under analysis on base-band in $[THz]$.
- **fibProf** Double Matrix Type $[nchxzeta_samples]$. Power Evolution Profile.
- **sigPSD** Double Array Data Type. WDM-Signal PSD in $[\frac{W}{Hz}]$.
- **zeta_samples** Double Type. Number of samples for z. It should be the minimum to solve the worst case.

- Output

- **GGf** Double Data Type. NLI PSD in $[\frac{W}{Hz}]$.

5.1.9 GGN-Model Hybrid Solution

The file named as "*GGNLI_f_3.m*" computes the GGN-Model through numerical integration (3.18). This function computes just the channel under test and its neighbors. The tails of (3.20) is filled by (3.21) The input and output variables are listed below:

- Input
 - **fiberParams** Struct Data Type. It contains fiber parameters.
 - **signal** Struct Data Type. It contains signal parameters.
 - **sys** Struct Data Type. It contains system parameters.
 - **f** Double Data Type. Frequency under analysis on base-band in $[THz]$.
 - **fibProf** Double Matrix Type $[nch \times zeta_samples]$. Power Evolution Profile.
 - **sigPSD** Double Array Data Type. WDM-Signal PSD in $[\frac{W}{Hz}]$.
 - **zeta_samples** Double Type. Number of samples for z. It should be the minimum to solve the worst case.
 - **I2_gn** Double Array Data Type. Function (3.20)
 - **channel** Double Type. Channel under analysis.
- Output
 - **GGf** Double Data Type. NLI PSD in $[\frac{W}{Hz}]$.

Figure 5.1 provides to the reader a general view of the code and how different fractions of it interact each other.

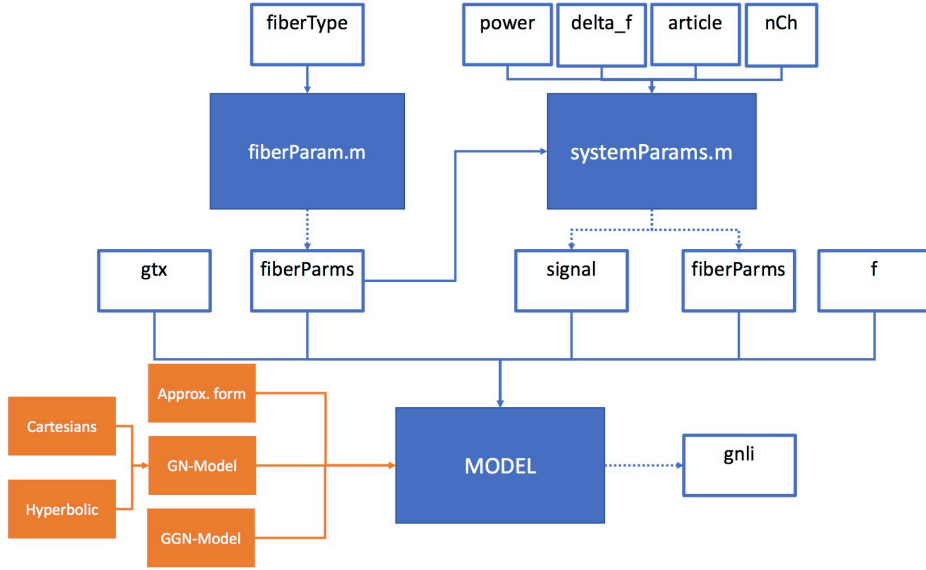


Figure 5.1: Code Architecture

5.2 Code performance

The main limitation of both the GN-Model and the GGN-Model is its computing time. As the number of channels increases, then the total bandwidth grows as

well, more number of samples are required in order to solve properly the integral. Therefore, this limitation determine the maximum number of channels, that can be simulated depending on the hardware resources. Besides all, this imperfection becomes more critical when GGN-Model is computed, as the spatial integral is introduced to consider the spatial dependence as well. All this issues are briefly analyzed on this section.

5.2.1 Hyperbolic Coordinates vs Cartesian Coordinates

The hyperbolic reference system, derived on Appendix E ("Details of the derivation of the GNR in hyperbolic Coordinates") of [12], reduces significantly the computing time with respect to the one of the Cartesian Coordinates approach. The computing time has been measured using the Matlab command called *"tic-toc"*.

A WDM system along SMF fiber has been simulated. The signals are 32 GBaud/s and 50 GHz frequency spaced. The number of channels is varied from 3 to 101 channels. Both Cartesian based model and Hyperbolic Coordinates based one computing times have been measured through *"tic-toc"* command as explained above. Figure 5.2 shows the evolution of the consuming time for both cases.

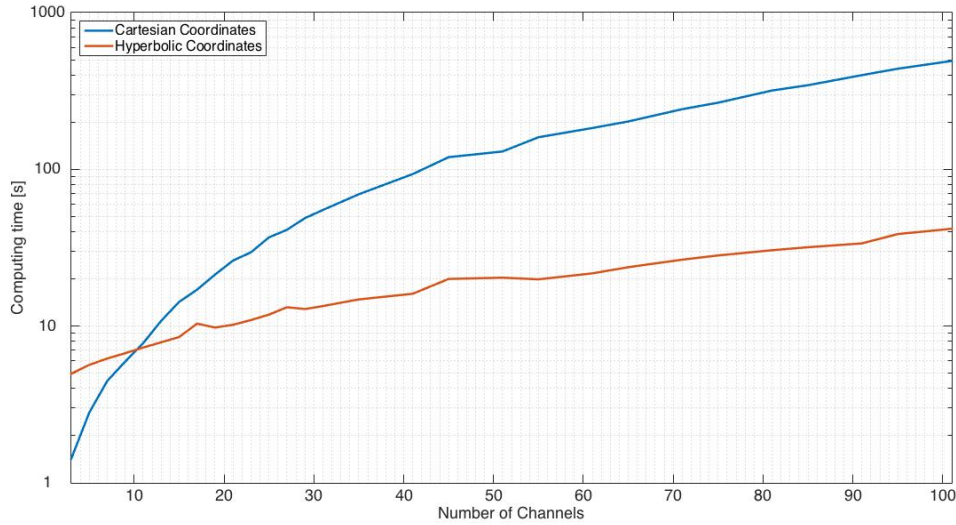


Figure 5.2: The graph shows the computing time of the GN-Model for different number of channels. Blue solid line represents Cartesian Coordinates approach and red solid line does Hyperbolic Coordinates one. Cartesian Coordinates approach is faster than Hyperbolic Coordinates one until 10 channels. Beyond 10 channels is the other way around.

The Hyperbolic Coordinates approach is slower until 10 channels, however it is much faster than the Cartesian Coordinates one. The reasons comes from the fact through the Hyperbolic Coordinates approach many samples are avoided as most

of the NLI energy is accumulated as ν_1 goes to zero, decreasing exponentially as ν_1 increases (Figure 2.2).

5.2.2 Generalized Gaussian Noise Model Matrix Solution

One of the critical issues of the GGN-Model implementation is how the integrals are solved, specially, for ultra wide-band system. Thus, it is fundamental to implement it as optimal as possible. Matlab environment works much faster with vector-based implementations rather than with loop-based implementations. According to this, the model has been implemented as follows:

Recall the GNRF (3.18):

$$G_{NLI}(z, f) = \frac{16}{27} \gamma^2 \rho(z, f)^2 \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} G_{TX}(f_1) G_{TX}(f_2) G_{TX}(f_1 + f_2 - f) \\ \left| \int_0^z e^{j4\pi^2(f_1-f)(f_2-f)\beta_2\zeta} \frac{\rho(\zeta, f_1)\rho(\zeta, f_3)\rho(\zeta, f_2)}{\rho(\zeta, f)} d\zeta \right|^2 df_1 df_2$$

For the sake of clearness, the function is renamed.

$$G(f_1, f_2, f) = G_{TX}(f_1) G_{TX}(f_2) G_{TX}(f_1 + f_2 - f) \quad (5.1)$$

$$Z(f_1, f_2, f, \zeta) = e^{j4\pi^2(f_1-f)(f_2-f)\beta_2\zeta} \frac{\rho(\zeta, f_1)\rho(\zeta, f_3)\rho(\zeta, f_2)}{\rho(\zeta, f)} \quad (5.2)$$

(5.1) and (5.2) are introduced into the GNRF (3.18).

$$G_{NLI}(z, f) = \frac{16}{27} \gamma^2 \rho(z, f)^2 \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} G(f_1, f_2, f) \\ \left| \int_0^z Z(f_1, f_2, f, \zeta) d\zeta \right|^2 df_1 df_2$$

The variable f is the frequency under analysis, then it is a single double value on the code. f_1 and f_2 are double vectors spanning from $-B_w/2$ until $B_w/2$. ζ is also a double vector from 0 until L_s (Span Length) and z is the spatial position where $G_{NLI}(z, f)$ is computed, it is usually at the end of the span (L_s). The integral are solved as follows: A loop is built up. The loop length is equivalent to f_1 one. Thus, on i^{th} iteration f_1 is set to a single value, i.e. $f_1 = f_1(i)$.

$$G(f_1(i), f_2, f) = G_{TX}(f_1(i)) G_{TX}(f_2) G_{TX}(f_1(i) + f_2 - f) \quad (5.3)$$

$$Z(f_1(i), f_2, f, \zeta) = e^{j4\pi^2(f_1(i)-f)(f_2-f)\beta_2\zeta} \frac{\rho(\zeta, f_1(i))\rho(\zeta, f_3)\rho(\zeta, f_2)}{\rho(\zeta, f)} \quad (5.4)$$

5.2. CODE PERFORMANCE

At this point, a matrix can represents the integration function of ζ -integral as there are only two variables (f_2 and ζ). In other words, $Z(f_1, f_2, f, \zeta)$ is implemented as a matrix where each column represents one sample of ζ and all samples of f_2 , meanwhile, each row represents one sample of f_2 and all samples of ζ . Thus, Z is the integration function on a matrix-like shape.

$$\begin{aligned} Z &= \begin{bmatrix} z_{11} & z_{12} & z_{13} & \dots & z_{1n} \\ z_{21} & z_{22} & z_{23} & \dots & z_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ z_{m1} & z_{m2} & z_{m3} & \dots & z_{mn} \end{bmatrix} \\ f_2 &\in \left[-\frac{Bw}{2}, \frac{Bw}{2} \right] \\ \zeta &\in [0, L_s] \end{aligned} \tag{5.5}$$

For instance the element z_{13} represents the value for $f_2(1)$ and $\zeta(3)$. Later on, the integral is solved numerically row by row, as the variable to be integrated is ζ . However, as the number of channels grows, the number of samples for f_2 grows as well as for ζ (see Figures 3.1 and 3.2). At a certain number of channels, the matrix dimensions become so huge that our hardware resources are not able to manage the matrix. To cope with this, another method to solve the triple integral has been designed : Loop/Matrix Solution 5.2.3. The numerical solution of the integral of the matrix gives a vector as a result. This vector is named I_2 and represents next function:

$$I(f_2, f_1(i), f) = \left| \int_0^z Z(f_1, f_2, f, \zeta) d\zeta \right|^2 \tag{5.6}$$

The I_2 length is equal to f_2 one. Then, f_2 -integral is solved numerically. The results is stored on vector I_1 . The length of I_1 is equal to I_1 one. At the end, of the loop vector I_1 has been filled up for all samples of f_1 .

5.2.3 Generalized Gaussian Noise Model Loop/Matrix Solution

As it has been set forth above, for ultra wide-band system the matrix solution becomes unaffordable for the computing capacity of common hardware resources. Then, loop/matrix solution has been designed. On this approach a new loop is added. The length of the new loop is equal to the number of channels. At each iteration the matrix solution is applied. However, number of rows is reduced as f_2 interval is not $f_2 \in [-Bw/2, Bw/2]$ but $f_2 \in [-Bw_{Ch}/2, Bw_{Ch}/2]$. Adding a new

loop is less efficient, but on the other hand matrix dimensions are smaller.

$$\begin{aligned}
 Z &= \begin{bmatrix} z_{11} & z_{12} & z_{13} & \dots & z_{1n} \\ z_{21} & z_{22} & z_{23} & \dots & z_{2n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ z_{k1} & z_{m2} & z_{m3} & \dots & z_{kn} \end{bmatrix} \\
 f_2 &\in \left[-\frac{Bw_{Ch}}{2}, \frac{Bw_{Ch}}{2} \right] \\
 \zeta &\in [0, L_s]
 \end{aligned} \tag{5.7}$$

Eventually, if our hardware resources does not allow us to manage matrices so big, the second approach is required, even though the computation is slower.

5.2.4 Generalized Gaussian Noise Model Hybrid Solution

On section 3.2 the FWM Efficiency Factor is addressed and how it affects the computing time. First of all, a fixed number of samples was considered. In this case, the worst case should be analyzed. The reason is that the integral should be solved properly when $(f_1 - f)(f_2 - f)$ is both big and small. Clearly, if the number of samples chosen allows to compute the integral when the sinusoidal functions are fast, it does when the sinusoidal functions are slow as well. However, this is not optimal at all since more resources than enough are used to solve the integral when the sinusoidal is slow. The hybrid method is to compute I_2^{GGN} (3.20) only for the channel under analysis and its neighbour channels. For the rest of channels I_2^{GN} (3.21) is computed. The reason is that most of the information is allocated on the channel under analysis and its neighbours as it is seen on 3.3.

Chapter 6

Experimental verification and simulations

In this section all modeling and experimental verification carried out are explained. First of all, both models have been implemented in Matlab. Then, a set of papers, where already verified results are shown, has been compared with the results provided by both GN-Model and GGN-Model implementation. Finally, some ad hoc experiment have been carried out in order to prove the performance of the Generalized Gaussian Noise Model.

6.1 Telecom Italia Field Trial

Field Trial Setup

The Field Trial [5] carried on Telecom Italy link has been simulated in order to verify that the GN-Model has been properly implemented. The system under test was a solution for medium and long haul applications. Both the transmitter and the receiver were located in Torino, however the signal traveled through the link till Chivasso (38km) in a loop back configuration. Figure 6.8 shows Field Trial architecture. On Figure 7.8.a, the Transmitter (Torino) is depicted as well as the channel under test and its neighboring channel setup. On 7.8.b, the link connecting Torino TEx1 and Chivasso in a loop back configuration at TEx2. On 7.8.c, the OTDR trace of 80 km link. Finally, on 7.8.d the receiver in Torino.

6.1. TELECOM ITALIA FIELD TRIAL

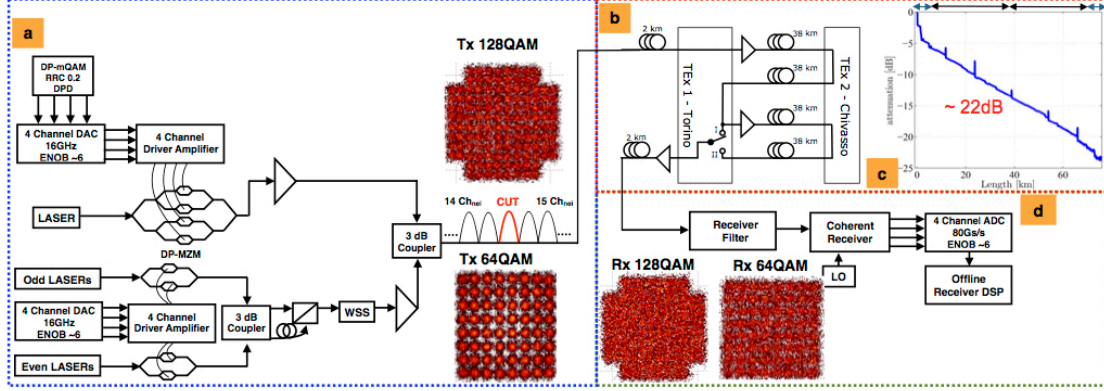


Figure 6.1: Field Trial Architecture. This figure has been taken from [5] (Fig. 1)

Table 6.1 summarizes the experiment parameters.

Mod. Format	Baud Rate (GBaud)	Net Bit Rate (Gbps)
DP-4QAM	57.24	200
DP-8QAM	42	200
DP-16QAM	33.01	200
Mod. Format	FEC Threshold	Spacing Used (GHz)
DP-4QAM	1.0E-02	68.75
DP-8QAM	2.4E-02	50
DP-16QAM	3.4E-02	37.5
Mod. Format	Sp. Efficiency (bit/s/Hz)	FEC Overhead
DP-4QAM	2.9	7%
DP-8QAM	3.9	15%
DP-16QAM	5.5	20%

Table 6.1: Main Experiment Parameters

Fiber parameters are listed on Table 6.2

Fiber Parameters		
Att. Constant α [$\frac{dB}{km}$]	GVD β_2 [$\frac{ps^2}{km}$]	NLI coeff. γ [$\frac{1}{Wkm}$]
0.3	-21	1.21

Table 6.2: Main Fiber Parameters

Results and Conclusions

According to the results reported by [5] for 193 THz channel the optimal launch power is > 5 dBm for 200G DP-4QAM and around 3 dBm for both 200G DP-8QAM and 200G DP-16QAM. The power > 5 dBm is quite high but it can be

explained by the large bandwidth of the DP-4QAM 57.24 GBaud signal (for more information about it: section "results" [5]). The system was simulated according to three different models: GN-Model Approximated Formula, GN-Model and GGN-Model. The final expression for the three models are summarized below:

$$G_{NLI} \approx \frac{4}{27} \frac{\gamma^2 P_{ch}}{\pi |\beta_2| \alpha R_s^3} \operatorname{asinh} \left(\frac{1}{4\alpha} \pi^2 |\beta_2| R_s^2 [N_{ch}^2]^{\frac{R_s}{\Delta f}} \right) \quad (6.1)$$

$$G_{NLI}(0) = \frac{64}{27} \gamma^2 N_s L_{s,eff}^2 \int_0^{B_{WDM}/2} \rho(\nu_1, 0) \chi(\nu_1, 0) |2\nu_1| \int_0^{\log_e(B_{WDM}/2\nu_1)} \left[G_{WDM}(\nu_1 e^{\nu_2}) G_{WDM}(\nu_1 e^{-\nu_2}) \right. \\ \left. \left[G_{WDM}(2\nu_1 \cosh(\nu_2)) + G_{WDM}(2\nu_1 \sinh(\nu_2)) \right] d\nu_2 d\nu_1 \right] \quad (6.2)$$

$$G_{NLI}(f) = \frac{16}{27} \gamma^2 \rho(z, f)^2 \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} G_{TX}(f_1) G_{TX}(f_2) G_{TX}(f_1 + f_2 - f) \\ \left| \int_0^z e^{j4\pi^2(f_1-f)(f_2-f)\beta_2\zeta} \frac{\rho(\zeta, f_1)\rho(\zeta, f_3)\rho(\zeta, f_2)}{\rho(\zeta, f)} d\zeta \right|^2 df_1 df_2 \quad (6.3)$$

The verification process was carried out as follows:

Figure 2 of [5] compares theoretical BER and measured BER for different OSNR. The measured BER were obtained on a back-to-back configuration, i.e. on the absence of optical link. Next table illustrates the minimum SNR in other to obtained a given BER.

Mod. Format	BER	SNR_{sens} [dB]	OSNR [dB]
DP-4QAM	1.0e-2	12.39	19
DP-8QAM	2.4e-2	12.13	17.4
DP-16QAM	3.4e-2	13.77	18

Table 6.3: Back-to-back values in terms of OSNR and SNR

Later, the number of span has been changed until the estimated SNR was as much as close to SNR_{sens} . The reason the curves have stepped-like shape is that on a real system the number of span should be an integer number. In other words, on theoretical analysis the maximum number of spans can be $N_s=13.5$, however, half span can be deployed on real systems. Figure 6.2 depicts the maximum length variation against launch power.

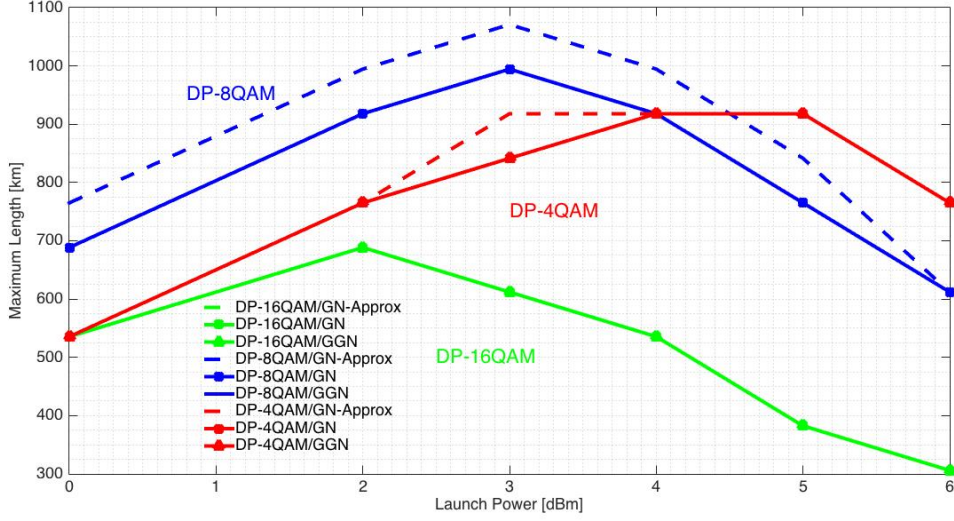


Figure 6.2: The optimal input power is 4 dBm, 3 dBm and 2 dBm for DP-4QAM, DP-8QAM and DP-16QAM respectively. Dashed line represent GN-Model Approximated Formula with incoherent accumulation. This model reports larger distances for both DP-4QAM and DP-8QAM, because for these mod. schemes the frequency spacing is higher, then NLI accumulation is more coherent. For DP-16QAM scheme the frequency spacing is lower.

The results depicted on Figure 6.2 shows that the estimations delivered by GN-Model and GGN-Model are equal. The main reason is that the "Channel Under Test" (CUT) was the central channel. In that channel the Raman Cross-Talk is negligible as the central channel delivers the same amount of power that others deliver to it. Finally, the estimation obtained through the approximated formula are higher than those obtained by numerical integration. The numerical integration approach has considered coherent accumulation, meanwhile the approximated formula 2.8 has considered incoherent one. For the incoherent accumulation case, a closed-formula can be derived to compute the maximum length for a required BER/OSNR (Equation 6.4). Therefore, the incoherent accumulation approach reports less NLI-like power than the coherent one. Consequently, the maximum length is higher.

$$M = \frac{1}{OSNR_{sens}} \frac{P_{ch}}{P_{ASE}^{(1)} + \eta^{(1)} P_{ch}^3} \quad (6.4)$$

6.2 Maximum Distance GN-Model

The scenario proposed is an uncompensated-like coherent optical transmission. It is described on [14], specifically on Figure 4.b. The transmitter are I-Q ones, the

6.3. EXPERIMENT AT ORANGE LABORATORY

span length is 100 km, the amplification is lumped-like and the fiber used was Pure-silica-core fiber (PSCF). As it has been said on section 4.2 maximum distance is defined as the distance such that satisfies a required SNR. The SNR is determined by the required BER. Finally, back-to-back penalty should be taken into account so that ISI and cross-talk noise are also considered. This parameter depends on modulation format and how far away are the channels placed one from another. The greater the frequency spacing, the smaller the penalty. PM-8QAM SMF scenario for different frequency spacing has been simulated, the simulated results provide the same optimal launched power and maximum lengths as it is depicted on Figure 4.b of [14]. Figure 6.3 depicts the results reported by my implementation.

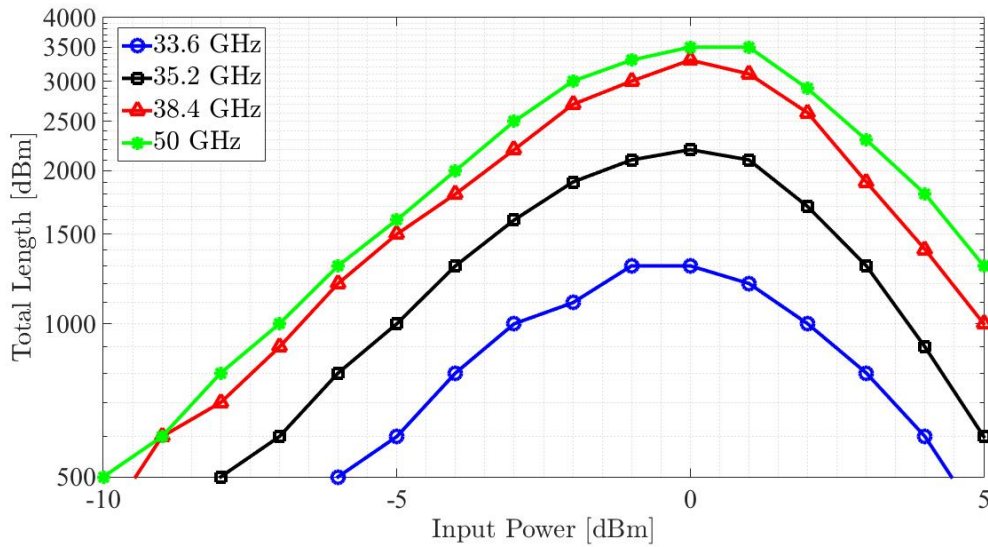


Figure 6.3: Maximum length for different input powers. SMF-QPSK case

The optimal launched power for all frequency spacing is around 0 dBm.

6.3 Experiment at Orange Laboratory

We simulate the experiment described in [15]. On this paper the GGN-Model was verified with experimental results and then enables us to use it.

6.3.1 Laboratory Setup

The optical link consists on 20 spans of 80 km each. The optical fiber is Corning SMF -28e+. 58 channels were configured, 7 of those were commercial 100 Gb/s transponders and 51 came from a comb of lasers modulated by two 28 GBaud/s 100 Gb/s laboratory transmitters. At the beginning, a WSS was used to combine the channels as well as after span number 6 and 13 in order to flatten the signal spectrum.

6.3.2 Results and Conclusions

Figure 6.4 shows the evolution of SNR_{NLI} along the link. Clearly, the reported performance for low and high channel is the same, as the GN-Model is symmetric with respect to the central frequency. Furthermore, the difference between both curves remains constant through the link.

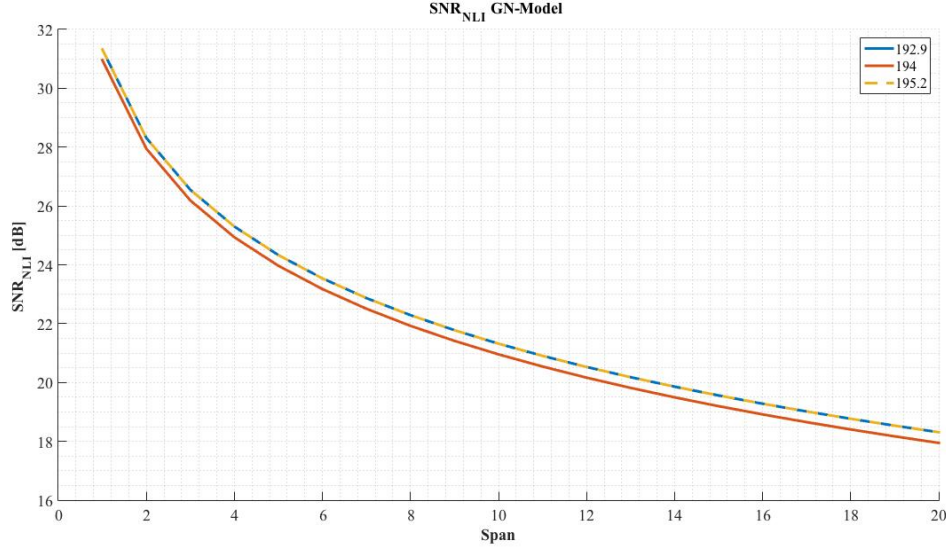


Figure 6.4: Red solid line represents the central channel, blue solid line left channel and yellow dashed line right one

Figure 6.5 depicts the evolution of SNR_{NLI} when tilt information was plugged into the model as a lumped effect. The left channel performance gets deteriorate along the link as its power increases, then the NLI-like does as well. On the other hand, the right channel performance increases along the link as its power decreases, then the NLI-like power does as well. Moreover, it is observed that after equalization stages (6th and 13th span) yellow (high channel) and blue (low channel) curves tend to get closer and if the WSS had been switched on from that span on, both curves will converge. However, they were switched off and both curves diverged.

6.3. EXPERIMENT AT ORANGE LABORATORY

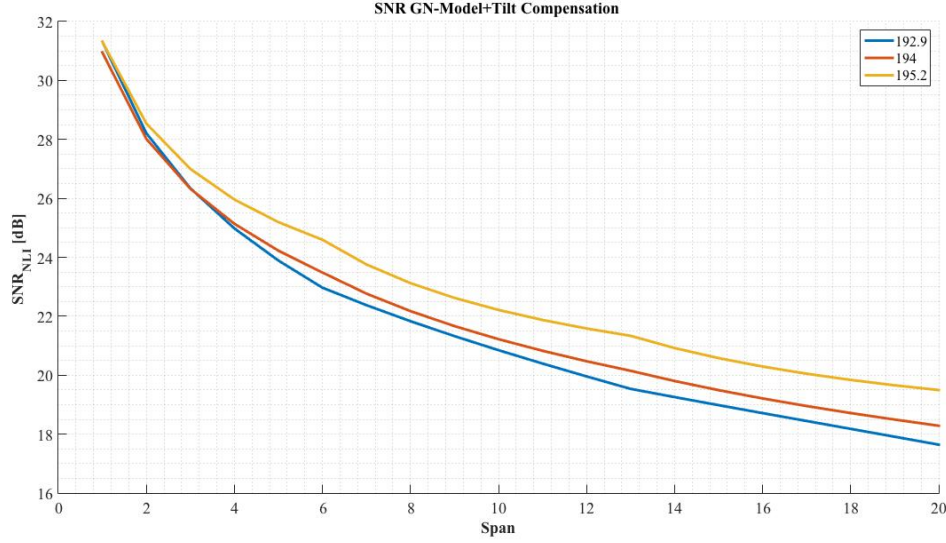


Figure 6.5: Red solid line represents the central channel, blue solid line left channel and yellow solid line right one. It is observed that equalization stages are placed on 6th and 13th spans.

Finally, the performance at the end of the link is depicted on Figure 6.6. Same results as in [15] has been obtained for Strategy 1 and Strategy 2. Strategy 1 refers to the broadly used GN-Model with incoherent accumulation, assuming flat launch power at the input of each span. Strategy 2 improves the previous strategy by plugging tilt information into the model. The tilt information was composed of both the fiber loss and the cross-talk due to SRS. The latter has been estimated for a common peak polarization-averaged Raman efficiency value for SMF of 0.39 1/W/km. Furthermore, after the 6th and 13th span the equalization stage is added. Strategy 1 matches correctly with the measured values at the central frequency, however, it fails on side channels. For Strategy 2, the performance becomes closer to the real one. However, the slope is not well estimated. As a result, the GGN-Model is highly needed to estimate the side channel for no tilt compensation system. The GGN-Model was not simulated for 58 channels, but 31 channels in order to make the simulation faster with the available resources. It is important to highlight that 50% of total NLI PSD is generated by around 4 channels for 20 spans (see Figure 2.5). Furthermore, the normalized, respect 101 channels, generated NLI PSD is 0.82 for 31 channels and 0.9 for 58 channels. Therefore, the analysis can be considered equivalent. Figure 6.7 depicts the values reported by GN and GGN-Model for 31 Channels.

6.3. EXPERIMENT AT ORANGE LABORATORY

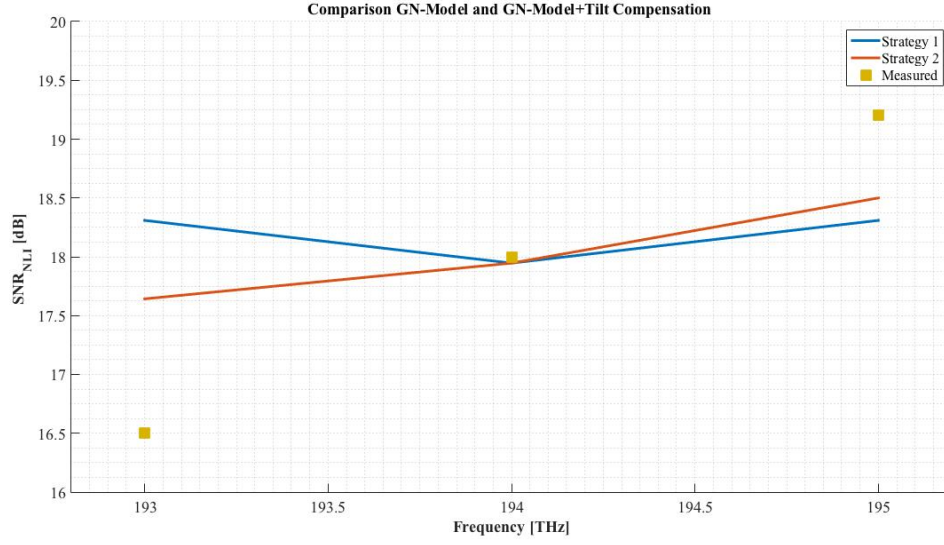


Figure 6.6: Red solid line represents the central channel, blue solid line left channel and yellow solid line right one

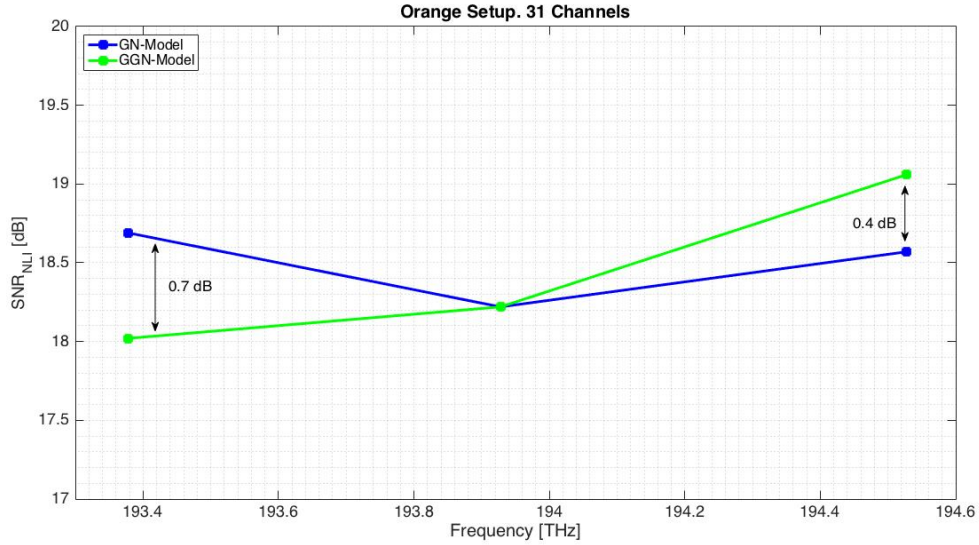


Figure 6.7: Blue solid line represents GN-Model and green solid line represents GGN-Model. GN-Model reports an over estimation at 193.5 THz with respect to GGN-Model and an under estimation at 194.5 THz with respect to GGN-Model.

Difference between GGN-Model and GN-Model reported by [15] at 193.5 THz and 194.5 THz is around 0.7 and 0.5 dB respectively. My own simulations reports 0.7 and 0.4 dB respectively.

6.4 Experiment at Coriant Laboratory. Full C-Band + 15 L-Band

We transmit 111 channels through five spans of 100.8 km (504 km). The experiment has been carried out at Coriant GmBh labs. The main goal of the experiment is to prove the inaccuracy of GN-Model estimation on side channels when SRS-induced crosstalk is not compensated. The SRS-induced crosstalk is a propagation effect that affects the four wave mixing process that occurs on the propagation along the optical fiber. The GN-Model do not consider the spatial-frequency variation of the modal field amplitude profile. Therefore, under/over estimations on side channels are reported.

6.4.1 Laboratory Setup

Full C-Band + 15 L-Band channels experiment was carried out at Coriant laboratory facility. The laboratory set up consists on 5 spans of Corning SMF-28 ULL optical fiber, where 111 32 GBaud/s optical channels are input. 96 channels come from a comb of lasers on C-Band and 15 channels (In Figure 6.8 20 channels on L-Band are shown, however, only 15 were switched on) come from another comb of lasers on L-Band (Figure 6.9 and 6.10 show input WDM signal for both C and L-Band). Both set of channels are combined and input together into the optical link. At the end of the link the WDM signal is divided again in C and L Band and each signal is sent to its EDFA amplifier, as no C+L Amplifier was available. After the amplification stage both signals are combined again and are input into next span. The setup scheme is depicted on Figure 6.8.

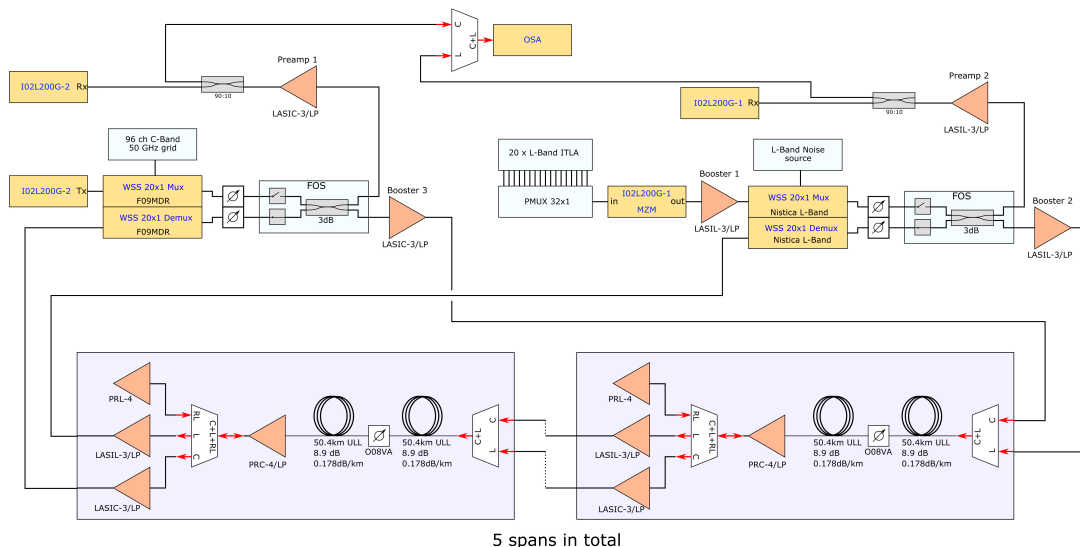


Figure 6.8: Experiment Architecture

6.4. EXPERIMENT AT CORIANT LABORATORY. FULL C-BAND + 15 L-BAND

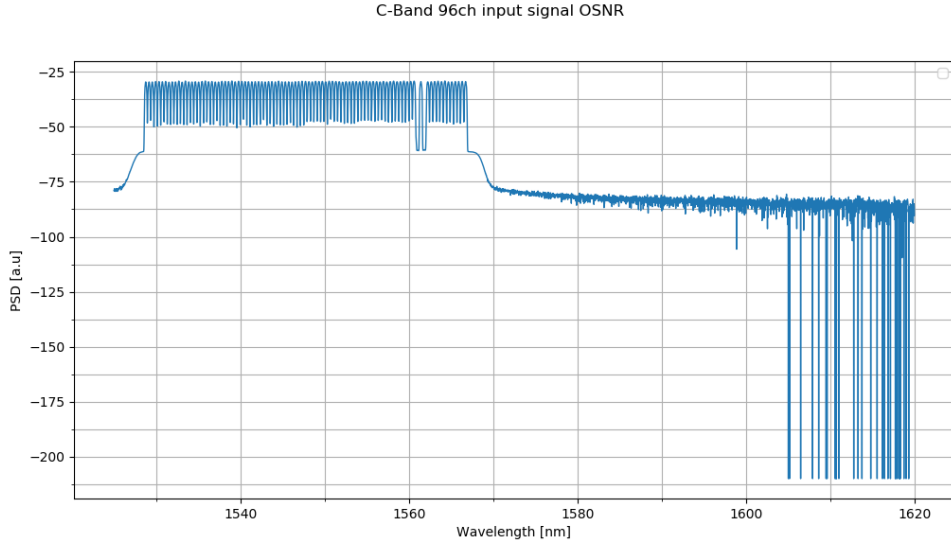


Figure 6.9: Input 96 Channels C-Band

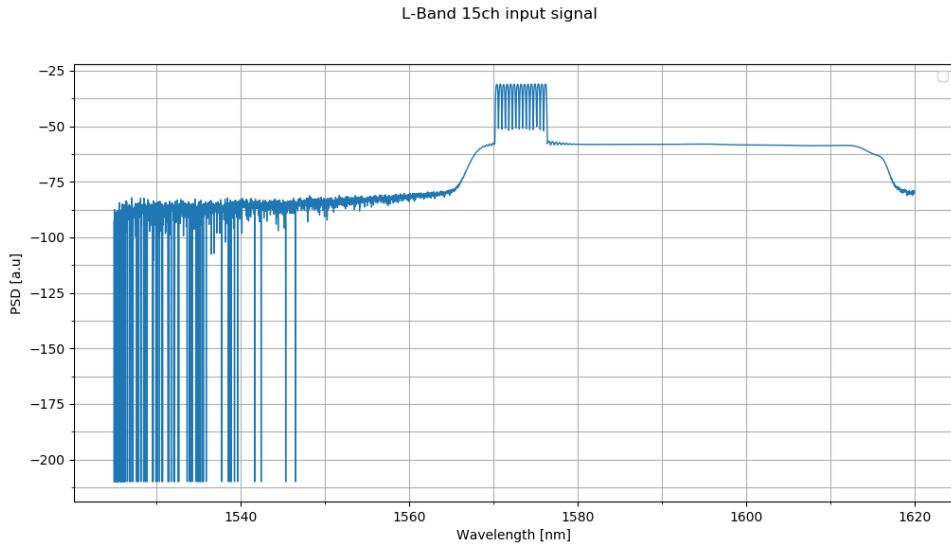


Figure 6.10: Input 15 Channels L-Band

Table 6.4 reports the fiber parameters provided by the manufacturer.

Attenuation Constant α	Dispersion Parameter D	Non-Linear Coefficient γ
0.16 (dB/km)	18.1 (ps/nm/km)	1.0088 (1/W/km)

Table 6.4: Main Fiber Parameters

Table 6.5 reports the system parameters.

6.4. EXPERIMENT AT CORIANT LABORATORY. FULL C-BAND + 15 L-BAND

Symbol Rate	Roll-off factor	Modulation Format	FEC Overhead
32 (<i>GBaud/s</i>)	0.15	QPSK	15 %

Table 6.5: System Parameters

The values depicted on Table 6.4 are averaged values reported by the manufacturer available on the data-sheet. However, the exact loss produced at each span has been measured (WWG OLP-18C Power Meter). It is critical to be accurate on the input power at each span, as a small variance on the input power value can affect substantially to the NLI-power estimate.

Table 6.6 depicts the loss at each span and the loss source.

Span	Device	Loss [dB]
1	C+L Combiner	0.63
1	Fiber	16.6
1	PRC-4LP + Splitter	1.8
2	C+L Combiner	0.7
2	Fiber	16.9
2	PRC-4LP + Splitter	2.3
3	C+L Combiner	0.75
3	Fiber	16.65
3	PRC-4LP + Splitter	2.2
4	C+L Combiner	0.7
4	Fiber	16.9
4	PRC-4LP + Splitter	0.63
5	C+L Combiner	0.55
5	Fiber	16.45
5	PRC-4LP + Splitter	2.23

Table 6.6: Span Losses

The SRS cross talk causes a power difference of around 8 dB (C-Band) and 1 dB (L-Band) between low frequency channels and high frequency after 502 km. The ripple caused by the EDFA masked a bit the SRS-effect on L-Band. Figure 6.11 and 6.12 illustrate the output spectrum.

6.4. EXPERIMENT AT CORIANT LABORATORY. FULL C-BAND + 15 L-BAND

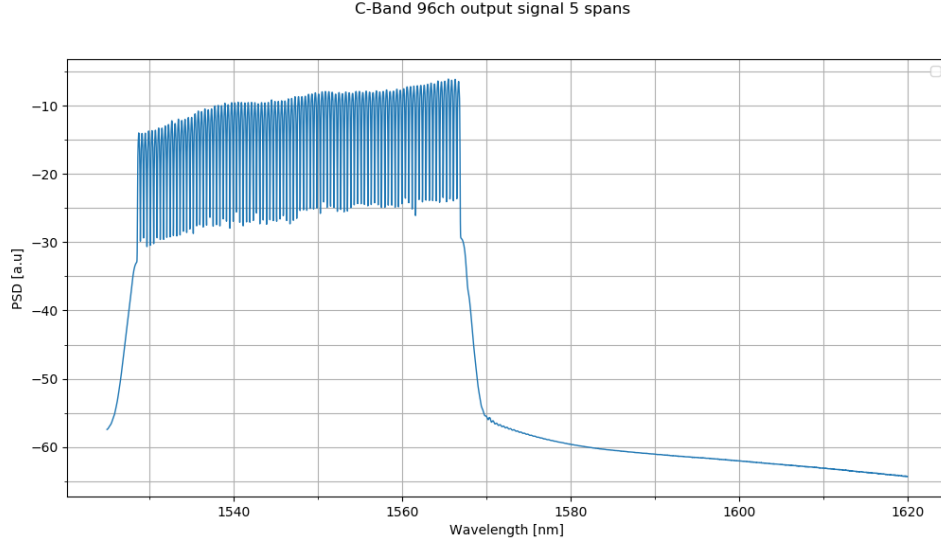


Figure 6.11: Output 96 Channels C-Band after 5 spans

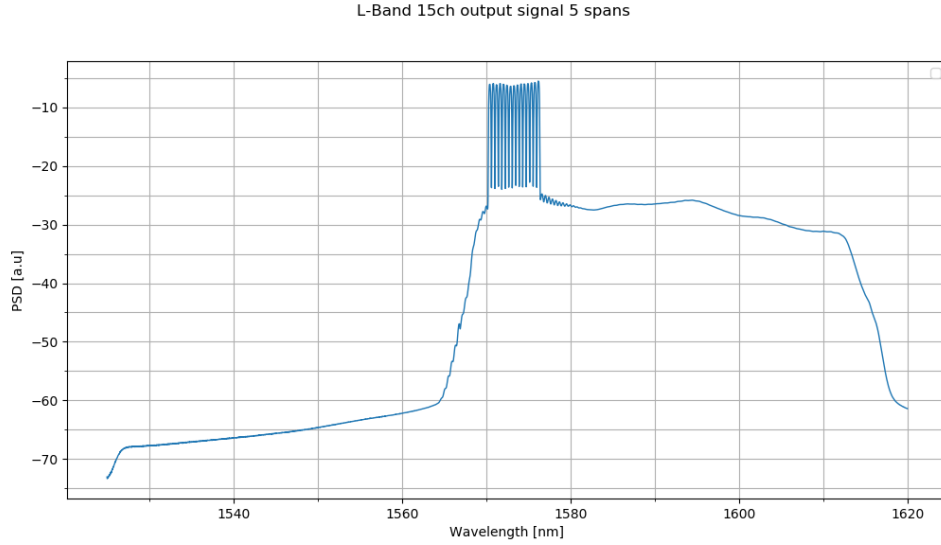


Figure 6.12: Output 15 Channels L-Band after 5 spans

6.4.2 Measurements and Conclusions

As it has been said above, the purpose is to show the inaccuracy of GN-Model on side channels when the tilt-effect is not compensated. To do so, SNR_{nli} is measured and calculated as defined on Chapter 3. First of all, SNR_{b2b} is measured in order to understand how much the ASE noise (measured with OSA YOKOGAWA AQ6370D) and other effects impair the system performance. Figure 6.13 depicts back-to-back measurements.

6.4. EXPERIMENT AT CORIANT LABORATORY. FULL C-BAND + 15 L-BAND

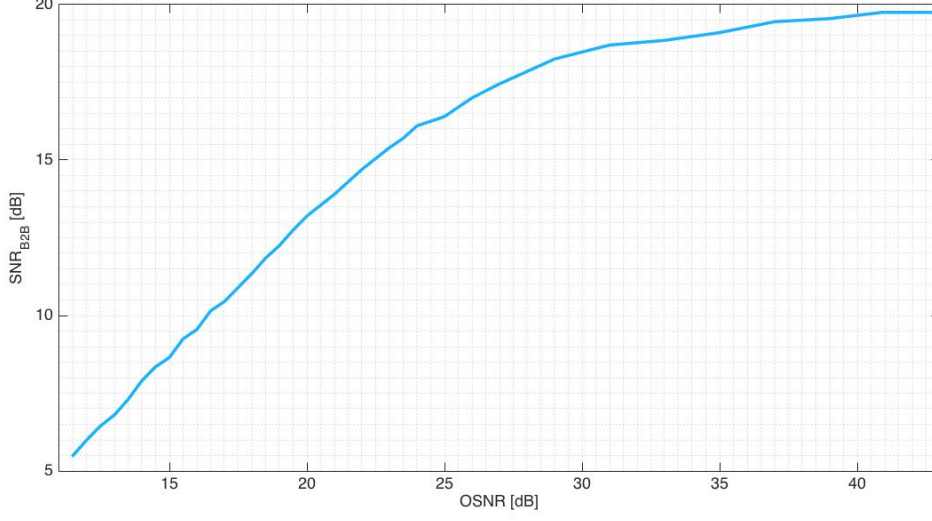


Figure 6.13: Back-to-back measurements

According to the back-to-back measurements, it is clearly observed that the receiver saturates at maximum SNR equal to around 20 dB. Then, when non-linear effects does not impair the performance (back-to-back), there are other effects beyond the ASE noise. Therefore, the back-to-back definition explained on Chapter 4 must be recast. Instead of $SNR_{b2b} = P_{ch}/P_{ase}$, SNR_{b2b} is redefined taking into account receiver impairments as $SNR_{b2b} = P_{ch}/(P_{ase} + P_{others})$. These impairments are named P_{others} since their sources are out of the scope of this thesis. The sources can be quantification noise or electrical noise. The method of measurement is defined as follows: The output OSNR is measured after five spans. Later, the receiver measures the so-called EVM defined in 4.1, which relates the signal and the error power. This magnitude considers both ASE noise, receiver and non-linear effects. After some mathematical computations non-linear impairments can be analyzed.

$$\begin{aligned}
 SNR_{rx} &= \frac{P_{ch}}{P_{ase} + P_{others} + P_{nli}} \\
 \frac{1}{SNR_{rx}} &= \frac{P_{ase} + P_{others} + P_{nli}}{P_{ch}} \\
 \frac{1}{SNR_{rx}} &= \frac{1}{SNR_{nli}} + \frac{1}{SNR_{b2b}} \\
 \frac{1}{SNR_{nli}} &= \frac{1}{SNR_{rx}} - \frac{1}{SNR_{b2b}}
 \end{aligned} \tag{6.5}$$

The strategies used to compute NLI-power are Strategy 1 and 2 defined on previous section. Strategy 1 compute only compute GN-Model at the beginning of first span and assumes that the accumulation is incoherent. Furthermore, Strategy 1 assumes that the spectrum remains flat through the whole link. In other words, Strategy

6.4. EXPERIMENT AT CORIANT LABORATORY. FULL C-BAND + 15 L-BAND

1 does not evaluate the SRS-induced cross-talk. On the other hand, Strategy 2 considers the SRS-induced cross-talk as a lumped effect. It means that the fiber propagation is modeled as a filter and the WDM signal passes through its transfer function. The block diagram is shown on Figure 6.14. $\rho(L_s, f)^2$ represents both the fiber loss and cross-talk due to SRS. It is considered as a lumped-effect instead of a distributed effect. Figure 6.15 depicts the measured values and the estimated ones. The maximum channel power was limited by the maximum total output power provided by the EDFA amplifier. It would have been more suitable to equalize every 5 or 6 span and then the total length had been larger. This option was not available as the link was loop-based and the equalizer was at the end of each span. Either the equalizer was switched on for every span or it was switched off for every span. The latter would not be useful as we were interested on tilted spectrum. Furthermore, the EDFA amplifier was considered flat. However, it is not true at all, as it is observed on Figure 6.11. Moreover, the measured values for the channel placed at high frequencies are not reliable as its values are quite low and the receiver might have not measured it properly. Finally, the GGN-Model was not able to be used because of lack of hardware resources. All over, the results show clearly that the GN-Model can estimate the central channel, but it fails to estimate side channels. Strategy 2 also fails because it considers the propagation process, that also affects the NLI generation, as a lumped-effect instead of a distributed one. GGN-Model analyzes it as a distributed effect and that is the reason of its accuracy.

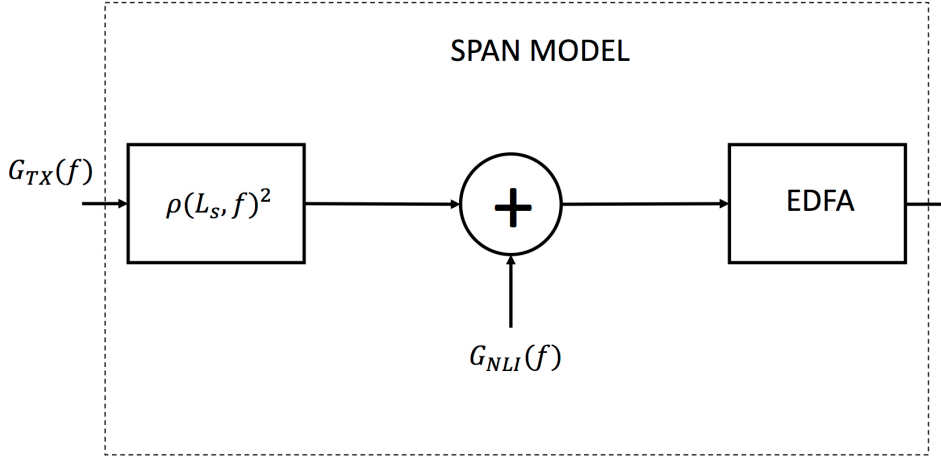


Figure 6.14: Span Block Diagram

6.4. EXPERIMENT AT CORIANT LABORATORY. FULL C-BAND + 15 L-BAND

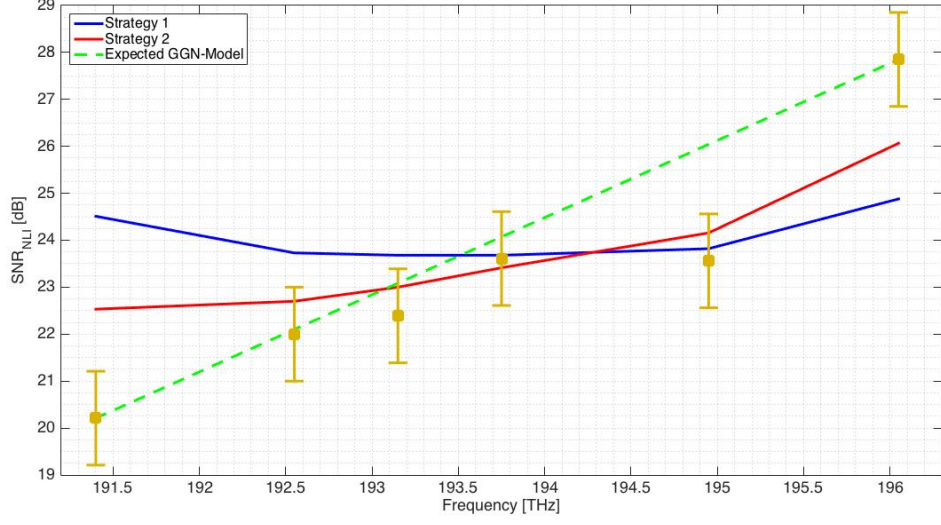


Figure 6.15: Blue solid line represents Strategy 1 and red solid line represents Strategy 2. Yellow bars represent ± 1 dB around the measured value. In other words, an estimation inside the interval $[x_{measured} - 1, x_{measured} + 1]$ is accepted.

Finally, η_{nl} parameter allow us to verify if SNR_{nli} was properly measured. This parameter is defined as [6]:

$$\eta_{nl} = (1/SNR_{nli}) / (P_{ch} - \Delta P)^2 \quad (6.6)$$

$\Delta P = 0.75$ dB and is equivalent to connector loss. Connector losses leads that the input power into the fiber is no exactly the expected one. Furthermore, NLI power varies with the cube of the input power. Therefore, a small difference on the input power leads to a high difference on the NLI power. η_{nl} should be constant. Figure 6.16 shows both $\eta_{measured}$ and $\eta_{simulated}$.

6.5. 15 CHANNELS ON C-BAND AND 35 CHANNELS ON L-BAND

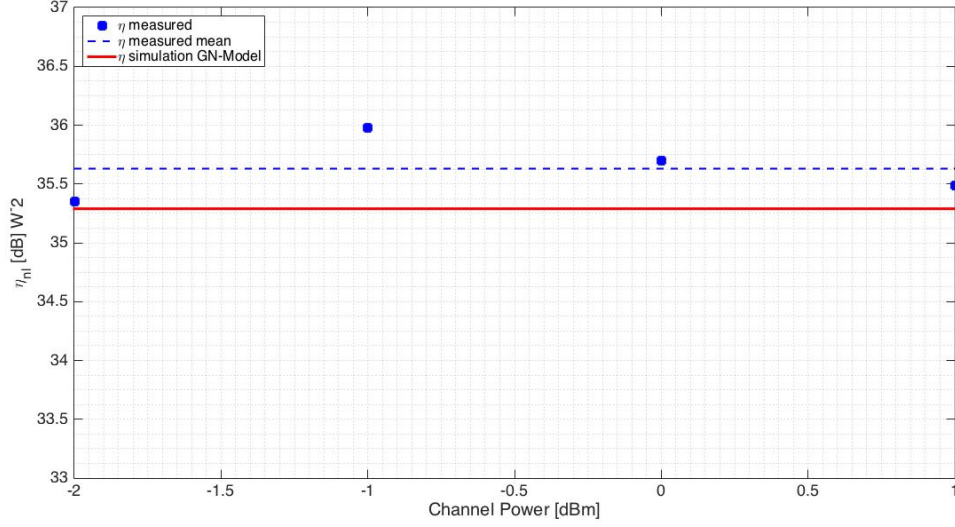


Figure 6.16: Measure for channel at 193.75 THz. Square-like dots represents the measurements of η_{nl} and solid red line the simulated one. 0.5 dB maximum difference among measurements is reported.

The difference between the averaged measured value and the simulated one is around 0.3 dB, which is an acceptable difference.

6.5 15 Channels on C-Band and 35 Channels on L-Band

Coriant lab scenario was simulated in order to obtain the maximum length reachable when 50 channels are input into the system. 15 of theses are placed on L-Band (from 190.2 THz to 190.9 THz) and 35 are placed on C-Band (191.35 THz to 193.05 THz). The laboratory setup is almost like the one used on section 6.4. However, the signal power is equalized every 5 spans. Last five spans were not equalized in order to see the SRS cross-talk effect. Then, PM-QPSK signals at 32 GBaud/s are transmitted, the channels are separated 50 GHz and the fiber is Corning SMF-28 ULL. The purpose of this simulation is to evaluate the maximum distance achievable by the system under the conditions summarized above. Table summarized the relation between three different BER values and their minimum required SNR.

Bit Error Rate (BER)	Signal-to-Noise Ratio (SNR)
1.19E-02	7.3 dB
5.50E-03	8.35 dB
8.55E-03	10.15 dB

Table 6.7: BER-SNR

6.5. 15 CHANNELS ON C-BAND AND 35 CHANNELS ON L-BAND

Channel number 3 (190.3 THz) and channel number 33 (191.8 THz) were analyzed. The metric used to study the system performance is the Signal-to-Noise Ratio as described in Chapter 4. The NLI noise-like power was computed through both GN-Model and GGN-Model. As expected, for the channel placed on L-Band the GN-Model failed. For instance, the maximum distance reported by GN-Model, in order to obtain a BER equal to $5.50\text{E-}03$ at the receiver side, is 2000 km greater than that reported by GGN-Model. Figure 6.17

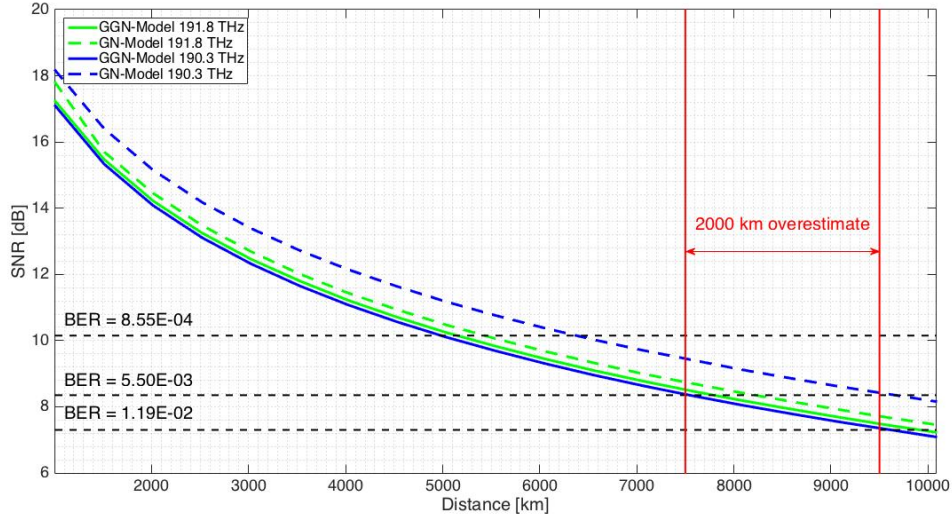


Figure 6.17: Blue solid line GGN-Model at 190.3 THz, blue dashed line GN-Model at 190.3, green solid line GGN-Model at 191.8 THz and green dashed line GN-Model at 191.8 THz. Both models report similar values at 191.8 THz. On the other hand, GN-Model fails at 190.3 THz.

Chapter 7

Conclusions

The final chapter entitled "*Conclusions*" is devoted to summarized main results obtained and to highlight futures developments to be carried out. Through this Master Thesis Project, GN-Model and GGN-Model have been analyzed and compared. Both models were implemented in order to carry out simulations to estimate the performance of an optical system through signal-to-noise ratio. Thus, the performance is measured in terms of Quality of Transmission of the physical layer.

Regarding the implementation and coding efficiency, one of the key point is the number of samples definition to solve the spatial integral for GGN-Model. It has been proved that incorrect estimates are reported if this value is not properly chosen. One possibility for futures implementations is to set the number of samples dynamically as it has been proved in this thesis that the optimal one changes a lot when the spectral components involved in Four-Wave-Mixing process are close to the frequency under analysis and when they are further in terms of frequency. The goal should be to improve the solution of the numerical integral.

Regarding the system performance estimation, it has been demonstrated that GN-Model is still accurate for the central channel of C-Band even if few channels were added on L-Band (15 Channels). However, it fails on side channels of C-Band due to cross-talk produced by Stimulated Raman Scattering. Measurements on L-Band would also be interesting in order to verify GN-Model reliability on this L-Band, whose commercial interest is growing every year.

Finally, GGN-Model estimates with high accuracy channels affected by cross-talk due to Stimulated Raman Scattering. In other words, GGN-Model considers this effect as distributed and its results are much better than the GN-Model, which considers it as a lumped effect. In terms of maximum distance estimation, GN-Model over-estimated in 2000 km the result provided by GGN-Model for a C+L system. In my opinion, future developments should be focus on testing GGN-Model on L-Band.

Appendices

Appendix A

Matlab Files

GN-Model. Hyperbolic Coordinates

Here it is shown the GN-Model in Hyperbolic Coordinates

```
1 % @Author: Angel Alonso
2 % Company: Coriant GmbH Muenchen
3 % Univesity: Politecnico di Torino & Universidad Politecnica
   de Madrid
4
5 % This returns NLI PSD at f.
6 % It uses Hyperbolic Coordinates
7 % Bibliography: Pierluigi Poggiolini : "Th GN Model of Non-
   Linear Propagation
8 %                                     in Uncompensated
   Coherent Optical Systems"
9
10 %
   =====

11 %                                     INPUT
12 %
   =====

13 %
14 % fiberParams: Type: Struct. It contains fiber parameters (
   alpha,beta2 and gamma)
15 % signal: Type: Struct. It contains signal parameters (power
   ,roll-off,Bw and Rs)
16 % sys: Type: Struct. It contains system parameters (number
   of channels ,
17 % span length , number of spans , bw, EDFA Gain, Figure Noise)
```

```

18 % f: Type: Double. Frequency under analysis. Baseband. f=0
    central channel [THz]
19 % gtx: Type: Double Array. WDM Signal Spectrum [W/Hz]
20 %
21 %

```

```

22 %                                OUTPUT
23 %

```

```

24 %
25 % GNLI0: Type: Double. NLI PSD at f [W/Hz]
26
27
28 function GNLI0=GNLI_f_hyp(fiberParams,signal,sys,f,gtx)
29     nSamples = floor(sys.Bw*23000/2);
30     freq = linspace(-sys.Bw/2,sys.Bw/2,nSamples);
31     % gtx = GTX(signal,sys,freq);
32     % Parameters
33     Leff = (1-exp(-2*fiberParams.alpha*sys.Ls))/(2*
        fiberParams.alpha); % Effective span length
34     resolv1=16; % After evaluating the integral
        convergence. Our resolution along v1 is 16
        samples/GHz
35     nSamplesv1=floor(resolv1*sys.Bw*1000);
36     % Let's compute GNRF at f=0
37     v1=linspace(0,sys.Bw,2*nSamplesv1);
38     %pace=v1(2);
39     p0num=1-exp(-2*fiberParams.alpha*sys.Ls)*exp(1i*4*pi
        ^2*fiberParams.GVD*sys.Ls*v1.^2);
40     p0dem=2*fiberParams.alpha-1i*4*pi^2*fiberParams.GVD*
        v1.^2;
41     p0=abs(p0num./p0dem).^2; %FWM efficiency for f=0
42     p0=p0/(Leff^2);
43     p0=10*log10(p0);
44     th=-30; % Threshold to speed up the numerical
        integration. The bigger, the faster.
45     ind=find(p0>=th);
46     p0=p0(ind);
47     p0=10.^(p0/10);
48     v1=v1(ind);
49     x0num=sin(2*sys.Ns*pi^2*v1.^2*fiberParams.GVD*sys.Ls
        ).^2;

```

```

50     x0dem=sin(2*pi^2*v1.^2*fiberParams.GVD*sys.Ls).^2;
51     x0=x0num./x0dem;
52     x0(1)=sys.Ns^2;
53
54     A=p0.*x0.*abs(2*v1);
55     A=A/max(A);
56     A=10*log10(A);
57     ind2 = find(A<-20);
58     Gnli0=zeros(1,length(v1));
59     fprintf('Processed 0%%')
60     for jj =1:length(v1)
61         if(jj==1)
62             v2upLim=log(sys.Bw/(0.0001));
63         else
64             v2upLim = log(sys.Bw/(v1(jj)));
65         end
66         if ismember(jj,ind2)
67
68         else
69             v2 = linspace(0,v2upLim,nSamplesv1);
70             freq01 = v1(jj).*exp(v2)+f;
71             freq02 = v1(jj).*exp(-v2)+f;
72             freq03 = 2*v1(jj).*cosh(v2)+f;
73             freq04 = -v1(jj).*exp(-v2)+f;
74             freq05 = 2*v1(jj).*sinh(v2)+f;
75             freq06 = -v1(jj).*exp(v2)+f;
76             freq07 = -2*v1(jj).*sinh(v2)+f;
77             freq08 = -2*v1(jj).*cosh(v2)+f;
78             g01 = interp1(freq,gtx,freq01,'cubic');
79             g02 = interp1(freq,gtx,freq02,'cubic');
80             g03 = interp1(freq,gtx,freq03,'cubic');
81             g04 = interp1(freq,gtx,freq04,'cubic');
82             g05 = interp1(freq,gtx,freq05,'cubic');
83             g06 = interp1(freq,gtx,freq06,'cubic');
84             g07 = interp1(freq,gtx,freq07,'cubic');
85             g08 = interp1(freq,gtx,freq08,'cubic');
86             I0=g01.*g02.*g03+g01.*g04.*g05+g06.*g02.*g07
              +g06.*g04.*g08;
87             Gnli0(jj)=trapz(v2,I0); %Inner integral.
              Integration domain over v2
88         end
89         if mod(jj, 100) == 0
90             fprintf('\b\b\b%g%%', round(jj*100/length(v1)
              )))

```



```

91     end
92     end
93     II=p0.*x0.*Gnli0.*abs(2*v1);
94     GNLI0=trapz(v1,II); %Outer integral. Integration
        domain over v1
95     GNLI0=(32/27)*(fiberParams.gamma^2)*(Leff^2)*GNLI0*1
        e-12; % PSD in W/Hz
96
97     end

```

GN-Model. Hyperbolic Coordinates at f=0

Here it is shown the GN-Model in Hyperbolic Coordinates at f=0

```

1 % @Author: Angel Alonso
2 % Company: Coriant GmbH Muenchen
3 % Univesity: Politecnico di Torino & Universidad Politecnica
    de Madrid
4
5 % This returns NLI PSD at f.
6 % It uses Cartesian Coordinates
7 % Bibliography: Pierluigi Poggiolini : "Th GN Model of Non-
    Linear Propagation
8 %                                     in Uncompensated
    Coherent Optical Systems"
9
10 %
    =====
11 %                                     INPUT
12 %
    =====
13 %
14 % fiberParams: Type: Struct. It contains fiber parameters (
    alpha,beta2 and gamma)
15 % signal: Type: Struct. It contains signal parameters (power
    ,roll-off,Bw and Rs)
16 % sys: Type: Struct. It contains system parameters (number
    of channels ,
17 % span length , number of spans , bw, EDFA Gain, Figure Noise)
18 %
19 %
    =====

```

```

20 %                                     OUTPUT
21 %

22 %
23 % GNLI0: Type: Double. NLI PSD at f [W/Hz]
24
25 function GNLI0=GNLI0_hyp(fiberParams, signal, sys)
26 nSamples = floor(sys.Bw*23000/2);
27 f = linspace(-sys.Bw/2, sys.Bw/2, nSamples);
28 gtx = GTX(signal, sys, f);
29 % Parameters
30 Leff = (1-exp(-2*fiberParams.alpha*sys.Ls))/(2*fiberParams.alpha); % Effective span length
31 resolv1=16; % After evaluating the integral convergence. Our
    resolution along v1 is 16 samples/GHz
32 nSamplesv1=floor(resolv1*sys.Bw*1000);
33 % Let's compute GNRF at f=0
34 v1=linspace(0, sys.Bw/2, nSamplesv1);
35 %pace=v1(2);
36 p0num=1-exp(-2*fiberParams.alpha*sys.Ls)*exp(1i*4*pi^2*
    fiberParams.GVD*sys.Ls*v1.^2);
37 p0dem=2*fiberParams.alpha-1i*4*pi^2*fiberParams.GVD*v1.^2;
38 p0=abs(p0num./p0dem).^2; %FWM efficiency for f=0
39 p0=p0/(Leff^2);
40 p0=10*log10(p0);
41 th=-30; % Threshold to speed up the numerical integration.
    The bigger, the faster.
42 ind=find(p0>=th);
43 p0=p0(ind);
44 p0=10.^(p0/10);
45 v1=v1(ind);
46 x0num=sin(2*sys.Ns*pi^2*v1.^2*fiberParams.GVD*sys.Ls).^2;
47 x0dem=sin(2*pi^2*v1.^2*fiberParams.GVD*sys.Ls).^2;
48 x0=x0num./x0dem;
49 x0(1)=sys.Ns^2;
50
51 A=p0.*x0.*abs(2*v1);
52 A=A/max(A);
53 A=10*log10(A);
54 ind2 = find(A<-20);
55 Gnli0=zeros(1, length(v1));
56 for jj =1:length(v1)

```

```

57     if (jj==1)
58         v2upLim=log(sys.Bw/(2*0.0001));
59     else
60         v2upLim = log(sys.Bw/(2*v1(jj)));
61     end
62     if ismember(jj, ind2)
63
64     else
65         v2 = linspace(0,v2upLim,nSamplesv1);
66         freq01 = v1(jj)*exp(v2);
67         freq02 = v1(jj)*exp(-v2);
68         freq03 = 2*v1(jj)*cosh(v2);
69         freq05 = 2*v1(jj)*sinh(v2);
70         max_f01 = max(freq01);
71         max_f02 = max(freq02);
72         max_f03 = max(freq03);
73         max_f05 = max(freq05);
74         min_f01 = min(freq01);
75         min_f02 = min(freq02);
76         min_f03 = min(freq03);
77         min_f05 = min(freq05);
78         if or(max_f01<-sys.Bw/2,min_f01>sys.Bw/2)
79             Gnli0(jj)=0;
80         elseif or(max_f02<-sys.Bw/2,min_f02>sys.Bw/2)
81             Gnli0(jj)=0;
82         elseif and(or(max_f03<-sys.Bw/2,min_f03>sys.Bw/2),or
83             (max_f05<-sys.Bw/2,min_f05>sys.Bw/2))
84             Gnli0(jj)=0;
85         else
86             g01 = interp1(f,gtx,freq01,'cubic');
87             g02 = interp1(f,gtx,freq02,'cubic');
88             g03 = interp1(f,gtx,freq03,'cubic');
89             g05 = interp1(f,gtx,freq05,'cubic');
90         end
91         I0=g01.*g02.*(g03+g05);
92         Gnli0(jj)=trapz(v2,I0); %Inner integral. Integration
93             domain over v2
94     end
95     if mod(jj, 100) == 0
96         fprintf('%g%% \n', round(jj*100/length(v1)))
97     end
98 end
99 II=p0.*x0.*Gnli0.*abs(2*v1);

```

```

98 GNLI0=trapz(v1,II); %Outer intergral. Integration domain
    over v1
99 GNLI0=(64/27)*(fiberParams.gamma^2)*(Leff^2)*GNLI0*1e-12; %
    PSD in W/Hz
100
101 end

```

GN-Model. Cartesian Coordinates

Here it is shown the GN-Model in Cartesian Coordinates

```

1 % @Author: Angel Alonso
2 % Company: Coriant GmbH Muenchen
3 % Univesity: Politecnico di Torino & Universidad Politecnica
    de Madrid
4
5 % This returns NLI PSD at f.
6 % It uses Cartesian Coordinates
7 % Bibliography: Pierluigi Poggiolini : "Th GN Model of Non-
    Linear Propagation
8 %                                     in Uncompensated
    Coherent Optical Systems"
9
10 %

```

```

11 %                                     INPUT
12 %

```

```

13 %
14 % fiberParams: Type: Struct. It contains fiber parameters (
    alpha,beta2 and gamma)
15 % signal: Type: Struct. It contains signal parameters (power
    ,roll-off,Bw and Rs)
16 % sys: Type: Struct. It contains system parameters (number
    of channels ,
17 % span length, number of spans, bw, EDFA Gain, Figure Noise)
18 % f: Type: Double. Frequency under analysis. Baseband. f=0
    central channel [THz]
19 % gtx: Type: Double Array. WDM Signal Spectrum [W/Hz]
20 %
21 %

```

```

22 %
23 %
24 %
25 % GNLI: Type: Double. NLI PSD at f [W/Hz]
26
27
28 function GNLI = GNLI_f(fiberParams , signal , sys , f , gtx)
29     nSamples = floor(sys.Bw*23000/8);
30     f2 = linspace(-sys.Bw/2,sys.Bw/2,nSamples);
31     f1 = linspace(-sys.Bw/2,sys.Bw/2,nSamples);
32
33     da = -2*fiberParams.alpha;
34     gtx_f2 = gtx;
35     gtx_f1 = gtx_f2;
36     I2 = zeros(1,nSamples);
37     n=nSamples;
38     [c ,index] = min(abs(f-f2));
39     for jj = 1:n
40         db = 1i*4*pi^2*(f1(jj)-f)*(f2-f)*fiberParams.GVD;
41         p1=abs((1-exp(da*sys.Ls).*exp(db*sys.Ls))./(-da-db))
            .^2;
42         if sys.Ns == 1
43             chi = ones(1,nSamples);
44         else
45             chi = (sin(db./2*sys.Ns*sys.Ls)./sin(db./2*sys.Ls))
                .^2;
46         end
47         vec2 = find(db==0);
48         chi(vec2) = sys.Ns^2;
49         if index-jj>=0
50             ze = zeros(1,index-jj);
51             gtx_f3 = horzcat(ze,gtx_f2(1:(n-length(ze))));
52             I2(jj)=trapz(f2,p1.*chi.*gtx_f1(jj).*gtx_f2.*
                gtx_f3);
53         else
54             ze = zeros(1,abs(index-jj));
55             gtx_f3 = horzcat(gtx_f2(1+length(ze):end),ze);
56             I2(jj)=trapz(f2,p1.*chi.*gtx_f1(jj).*gtx_f2.*
                gtx_f3);
57     end
58

```

```

59         if mod(jj , 100) == 0
60             fprintf( '%g%% \n' , round( jj *100/nSamples))
61         end
62
63     end
64     save( 'FUNCTION.mat' , 'I2 ');
65     I1 = trapz( f1 , I2 );
66     GNLI=16/27*fiberParams .gamma^2*I1*1e-12;
67
68 end

```

GGN-Model. Matrix Approach

Here it is shown the GGN-Model

```

1 % @Author: Angel Alonso
2 % Company: Coriant GmbH Muenchen
3 % Univesity: Politecnico di Torino & Universidad Politecnica
  de Madrid
4
5 % This returns NLI PSD at f.
6 %
  =====
7 %                               INPUT
8 %
  =====
9 %
10 % fiberParams: Type: Struct. It contains fiber parameters (
    alpha , beta2 and gamma)
11 % signal: Type: Struct. It contains signal parameters (power
    , roll-off , Bw and Rs)
12 % sys: Type: Struct. It contains system parameters (number
    of channels ,
13 % span length , number of spans , bw, EDFA Gain, Figure Noise)
14 % f: Type: Double. Frequency under analysis. Baseband. f=0
    central channel [THz]
15 % fibProf: Double Matrix Type. Power evolution profile
16 % sigPSD: Double Array. WDM Signal Spectrum [W/Hz]
17 % zeta_samples: Double Type. It set the number of samples of
    z
18 %

```

```

19 %

```

```

20 %                                     OUTPUT
21 %

```

```

22 %
23 % GGf: Type: Double. NLI PSD at f [W/Hz]
24
25 function GGf = GGNLI(fiberParams, signal, sys, f, fibProf,
    sigPSD, zeta_samples)
26 nSamples=floor(sys.Bw*23000/8);
27 f2=linspace(-sys.Bw/2,sys.Bw/2,nSamples);
28 f1=linspace(-sys.Bw/2,sys.Bw/2,nSamples);
29 gtx_f2 = sigPSD;
30 gtx_f1 = gtx_f2; % PSD Signal f1
31 I2 = zeros(1,nSamples); % Inner Integral
32 II2 = zeros(1,nSamples); % Inner Integral
33 n=nSamples;
34 [c,index] = min(abs(f-f2));
35 z = linspace(0,sys.Ls,zeta_samples);
36 fib = zeros(n,length(z));
37 for i=1:length(z)
38     fib(:,i)=interp1(sys.fCh,fibProf(:,i),f2,'cubic');
39 end
40 fib_f = fib(index,:);
41 fib_f = sqrt(fib_f);
42 fib_f = repmat(fib_f,n,1);
43
44 for jj=1:n
45     if mod(jj, 100) == 0
46         fprintf('%g%% \n', round(jj*100/nSamples))
47     end
48     % fprintf('%g \n',jj)
49     if gtx_f1(jj) == 0
50         I2(jj) = 0;
51     else
52         f3 = f1(jj)+f2-f;
53         if index-jj>=0
54             ze = zeros(1,index-jj);
55             db = 1i*4*pi^2*(f1(jj)-f)*(f2((index-jj+1):end)-
                f)*fiberParams.GVD;
56             db = reshape(db,length(db),1);

```

```

57     db = repmat(db,1,length(z));
58     Z = repmat(z,size(db,1),1);
59     F = db.*Z;
60     F = exp(F);
61     fib_f1 = fib(jj,:);
62     fib_f1 = sqrt(fib_f1);
63     fib_f1 = repmat(fib_f1,size(db,1),1);
64     fib_f2 = sqrt(fib);
65     fib_f3 = fib_f2(1:(n-length(ze)),:);
66     test = fib_f1.*fib_f((index-jj+1):end,:).*fib_f2
        ((index-jj+1):end,:)./fib_f3;
67     F = F.*test;
68     F = trapz(z,F,2);
69     F = abs(F).^2;
70     F = horzcat(ze,F');
71     gtx_f3 = horzcat(ze,gtx_f2(1:(n-length(ze))));
72     I2(jj)=trapz(f2,F.*gtx_f1(jj).*gtx_f2.*gtx_f3);
73
74     else
75         ze = zeros(1,abs(index-jj));
76         db = 1i*4*pi^2*(f1(jj)-f)*(f2(1:(end-abs(index-
            jj)))-f)*fiberParams.GVD;
77         db = reshape(db,length(db),1);
78         db = repmat(db,1,length(z));
79         Z = repmat(z,size(db,1),1);
80         F = db.*Z;
81         F = exp(F);
82         fib_f1 = fib(jj,:);
83         fib_f1 = sqrt(fib_f1);
84         fib_f1 = repmat(fib_f1,size(db,1),1);
85         fib_f3 = fib_f2((1+length(ze)):end,:);
86         test = fib_f1.*fib_f(1:(end-abs(index-jj)),:).*
            fib_f2(1:(end-abs(index-jj)),:)./fib_f3;
87         F = F.*test;
88         F = trapz(z,F,2);
89         F = abs(F).^2;
90         F = horzcat(F',ze);
91         gtx_f3 = horzcat(gtx_f2(1+length(ze):end),ze);
92         I2(jj)=trapz(f2,F.*gtx_f1(jj).*gtx_f2.*gtx_f3);
93     end
94 end
95 end
96 save('I2_I2_june_2.mat','I2');
97 I1=trapz(f1,I2);

```



```

98 GGf=16/27*fiberParams.gamma^2*I1*I1*1e-12;
99 end

```

GGN-Model. Loop/Matrix Approach

Here it is shown the GGN-Model

```

1 % @Author: Angel Alonso
2 % Company: Coriant GmbH Muenchen
3 % Univesity: Politecnico di Torino & Universidad Politecnica
  de Madrid
4
5 % This returns NLI PSD at f.
6 %
  =====
7 %                               INPUT
8 %
  =====
9 %
10 % fiberParams: Type: Struct. It contains fiber parameters (
    alpha,beta2 and gamma)
11 % signal: Type: Struct. It contains signal parameters (power
    ,roll-off,Bw and Rs)
12 % sys: Type: Struct. It contains system parameters (number
    of channels ,
13 % span length , number of spans , bw, EDFA Gain, Figure Noise)
14 % f: Type: Double. Frequency under analysis. Baseband. f=0
    central channel [THz]
15 % fibProf: Double Matrix Type. Power evolution profile
16 % sigPSD: Double Array. WDM Signal Spectrum [W/Hz]
17 % zeta_samples: Double Type. It set the number of samples of
    z
18 %
19 %
  =====
20 %                               OUTPUT
21 %
  =====
22 %
23 % GGf: Type: Double. NLI PSD at f [W/Hz]

```

```

24
25 function GGf = GGNLI_f2(fiberParams, signal, sys, f, fibProf,
    sigPSD, zeta_samples)
26 nSamples=floor(sys.Bw*23000/8);
27 f2=linspace(-sys.Bw/2,sys.Bw/2,nSamples);
28 f1=linspace(-sys.Bw/2,sys.Bw/2,nSamples);
29 gtx_f2 = sigPSD;
30 gtx_f1 = gtx_f2; % PSD Signal f1
31 nonull = find(not(gtx_f2==0));
32 first_nonull = nonull(1);
33 last_nonull = nonull(end);
34 I2 = zeros(1,nSamples); % Inner Integral
35
36 n=nSamples;
37 ns_thz = nSamples/sys.Bw;
38 ns_ch = floor(signal.BwCh*ns_thz);
39 ns_gap = floor(ns_thz*(sys.delta_f-signal.BwCh));
40 [c,index] = min(abs(f-f2));
41 z = linspace(0,sys.Ls,zeta_samples);
42 fib = zeros(n,length(z));
43 for i=1:length(z)
44     fib(:,i)=interp1(sys.fCh,fibProf(:,i),f2,'cubic');
45 end
46 fib_f = fib(index,:);
47 fib_f = sqrt(fib_f);
48
49
50 for jj=1:n
51
52     if mod(jj, 100) == 0
53         fprintf('%g%% \n', round(jj*100/nSamples))
54     end
55     % fprintf('%g \n',jj)
56     if gtx_f1(jj) == 0
57         I2(jj) = 0;
58     else
59         if index-jj>=0
60             if abs(index-jj)<50
61
62                 end
63                 ze = zeros(1,index-jj);
64                 id_f3 = horzcat(ze,(1:n-length(ze)));
65                 for kk = 1:sys.nCh %loop for every channel.

```

```

66         low = (ns_ch+ns_gap+1)*(kk-1) + first_nonnull
67         ;
68         high = (ns_ch+ns_gap+1)*(kk-1) + ns_ch+
69             first_nonnull;
70         id_f3_k = id_f3(low:high);
71         null_f3 = find(id_f3_k == 0);
72         if isempty(null_f3)
73             in=0;
74         else
75             in = null_f3(end);
76         end
77         id_f3_k_real = id_f3_k(in+1:end);
78         id_f2_k_real = (low+in):high;
79         db = 1i*4*pi^2*(f1(jj)-f)*(f2(id_f2_k_real)-
80             f)*fiberParams.GVD;
81         db = reshape(db,length(db),1);
82         db = repmat(db,1,length(z));
83         Z = repmat(z,size(db,1),1);
84         F = db.*Z;
85         F = exp(F);
86         fibb_f = repmat(fib_f,size(db,1),1);
87         fib_f1 = fib(jj,:);
88         fib_f1 = sqrt(fib_f1);
89         fib_f1 = repmat(fib_f1,size(db,1),1);
90         fib_f2 = fib(id_f2_k_real,:);
91         fib_f2 = sqrt(fib_f2);
92         fib_f3 = fib(id_f3_k_real,:);
93         fib_f3 = sqrt(fib_f3);
94         test = fib_f1.*fibb_f.*fib_f2./fib_f3;
95         F = F.*test;
96         F = trapz(z,F,2);
97         F = abs(F).^2;
98         F = F.';
99         A = gtx_f2(id_f2_k_real);
100         B = gtx_f2(id_f3_k_real);
101         freq = f2(id_f2_k_real);
102         I2(jj)=I2(jj)+trapz(freq,F.*gtx_f1(jj).*A.*B
103             ,2);
104     end
105 else
106     ze = zeros(1,abs(index-jj));
107     id_f3 =horzcat(1+length(ze):n,ze);
108     for kk = 1:sys.nCh

```

```

106         low = (ns_ch+ns_gap+1)*(kk-1) + first_nonnull
107             ;
108         high = (ns_ch+ns_gap+1)*(kk-1) + ns_ch+
109             first_nonnull;
110         id_f3_k = id_f3(low:high);
111         null_f3 = find(id_f3_k == 0);
112         if isempty(null_f3)
113             fin=0;
114         else
115             fin = length(null_f3);
116         end
117         id_f3_k_real = id_f3_k(1:end-fin);
118         id_f2_k_real = (low:high-fin);
119         db = 1i*4*pi^2*(f1(jj)-f)*(f2(id_f2_k_real)-
120             f)*fiberParams.GVD;
121         db = reshape(db,length(db),1);
122         db = repmat(db,1,length(z));
123         Z = repmat(z,size(db,1),1);
124         F = db.*Z;
125         F = exp(F);
126         fibb_f = repmat(fib_f,size(db,1),1);
127         fib_f1 = fib(jj,:);
128         fib_f1 = sqrt(fib_f1);
129         fib_f1 = repmat(fib_f1,size(db,1),1);
130         fib_f2 = fib(id_f2_k_real,:);
131         fib_f2 = sqrt(fib_f2);
132         fib_f3 = fib(id_f3_k_real,:);
133         fib_f3 = sqrt(fib_f3);
134         test = fib_f1.*fibb_f.*fib_f2./fib_f3;
135         F = F.*test;
136         F = trapz(z,F,2);
137         F = abs(F).^2;
138         F = F.';
139         A = gtx_f2(id_f2_k_real);
140         B = gtx_f2(id_f3_k_real);
141         freq = f2(id_f2_k_real);
142         I2(jj)=I2(jj)+trapz(freq,F.*gtx_f1(jj).*A.*B
143             ,2);
144     end
145 end
end

```

```

146
147 end
148 I1=trapz(f1,I2);
149 save('i2_20180617.mat','I2');
150 GGf=16/27*fiberParams.gamma^2*I1*1e-12;
151 end

```

GGN-Model. Hybrid Approach

Here it is shown the GGN-Model

```

1 % @Author: Angel Alonso
2 % Company: Coriant GmbH Muenchen
3 % Univesity: Politecnico di Torino & Universidad Politecnica
  de Madrid
4
5 % This returns NLI PSD at f.
6 %

```

```

7 %                               INPUT
8 %

```

```

9 %
10 % fiberParams: Type: Struct. It contains fiber parameters (
    alpha,beta2 and gamma)
11 % signal: Type: Struct. It contains signal parameters (power
    ,roll-off,Bw and Rs)
12 % sys: Type: Struct. It contains system parameters (number
    of channels ,
13 % span length , number of spans , bw, EDFA Gain, Figure Noise)
14 % f: Type: Double. Frequency under analysis. Baseband. f=0
    central channel [THz]
15 % fibProf: Double Matrix Type. Power evolution profile
16 % sigPSD: Double Array. WDM Signal Spectrum [W/Hz]
17 % zeta_samples: Double Type. It set the number of samples of
    z
18 % I2_gn: Double array type. I2 function to fill tails of I2
    computed
19 % through GGN-Model
20 % channel: Double type. Channel under test
21 %

```

```

22 %                                     OUTPUT
23 %

```

```

24 %
25 % GGf: Type: Double. NLI PSD at f [W/Hz]
26
27 function GGf = GGNLI_f3(fiberParams, signal, sys, f, fibProf,
    sigPSD, zeta_samples, I2_gn, channel)
28 nSamples=floor(sys.Bw*23000/2);
29 f2=linspace(-sys.Bw/2,sys.Bw/2,nSamples);
30 f1=linspace(-sys.Bw/2,sys.Bw/2,nSamples);
31 gtx_f2 = sigPSD;
32 gtx_f1 = gtx_f2; % PSD Signal f1
33 nonull = find(not(gtx_f2==0));
34 first_nonull = nonull(1);
35 last_nonull = nonull(end);
36 I2 = zeros(1,nSamples); % Inner Integral
37
38 n=nSamples;
39 ns_thz = nSamples/sys.Bw;
40 ns_ch = floor(signal.BwCh*ns_thz);
41 ns_gap = floor(ns_thz*(sys.delta_f-signal.BwCh));
42 [c,index] = min(abs(f-f2));
43 z = linspace(0,sys.Ls,zeta_samples);
44 fib = zeros(n,length(z));
45
46
47 if or(channel>1,channel<sys.nCh)
48     l = first_nonull + (ns_ch+ns_gap)*(channel-2);
49     h = first_nonull + (ns_ch+ns_gap)*(channel+1);
50 elseif channel == 1
51     l = 1;
52     h = no_null + (ns_ch+ns_gap)*(channel+1);
53 else
54     l = no_null + (ns_ch+ns_gap)*(channel-1);
55     h = length(f2);
56 end
57 for i=1:length(z)
58     fib(:,i)=interp1(sys.fCh,fibProf(:,i),f2,'cubic');
59 end
60 fib_f = fib(index,:);
61 fib_f = sqrt(fib_f);

```

```

62 for jj=1:h
63
64     if mod(jj , 100) == 0
65         fprintf( '%g%% \n', round((jj-1+1)*100/(h-1)))
66     end
67     %         fprintf('%g \n',jj)
68     if gtx_f1(jj) == 0
69         I2(jj) = 0;
70     else
71         if index-jj>=0
72             if abs(index-jj)<50
73
74                 end
75                 ze = zeros(1,index-jj);
76                 id_f3 = horzcat(ze,(1:n-length(ze)));
77                 for kk = 1:sys.nCh
78                     low = (ns_ch+ns_gap+1)*(kk-1) + first_nonnull
79                         ;
80                     high = (ns_ch+ns_gap+1)*(kk-1) + ns_ch+
81                         first_nonnull;
82                     id_f3_k = id_f3(low:high);
83                     null_f3 = find(id_f3_k == 0);
84                     if isempty(null_f3)
85                         in=0;
86                     else
87                         in = null_f3(end);
88                     end
89                     id_f3_k_real = id_f3_k(in+1:end);
90                     id_f2_k_real = (low+in):high;
91                     db = 1i*4*pi^2*(f1(jj)-f)*(f2(id_f2_k_real)-
92                         f)*fiberParams.GVD;
93                     db = reshape(db,length(db),1);
94                     db = repmat(db,1,length(z));
95                     Z = repmat(z,size(db,1),1);
96                     F = db.*Z;
97                     F = exp(F);
98                     fibb_f = repmat(fib_f , size(db,1),1);
99                     fib_f1 = fib(jj,:);
100                    fib_f1 = sqrt(fib_f1);
101                    fib_f1 = repmat(fib_f1 , size(db,1),1);
102                    fib_f2 = fib(id_f2_k_real,:);
103                    fib_f2 = sqrt(fib_f2);
104                    fib_f3 = fib(id_f3_k_real,:);
105                    fib_f3 = sqrt(fib_f3);

```

```

103         test = fib_f1.*fibb_f.*fib_f2./fib_f3;
104         F = F.*test;
105         F = trapz(z,F,2);
106         F = abs(F).^2;
107         F = F.';
108         A = gtx_f2(id_f2_k_real);
109         B = gtx_f2(id_f3_k_real);
110         freq = f2(id_f2_k_real);
111         I2(jj)=I2(jj)+trapz(freq,F.*gtx_f1(jj).*A.*B
112             ,2);
113
114     else
115         ze = zeros(1,abs(index-jj));
116         id_f3 =horzcat(1+length(ze):n,ze);
117         for kk = 1:sys.nCh
118             low = (ns_ch+ns_gap+1)*(kk-1) + first_nonnull
119                 ;
120             high = (ns_ch+ns_gap+1)*(kk-1) + ns_ch+
121                 first_nonnull;
122             id_f3_k = id_f3(low:high);
123             null_f3 = find(id_f3_k == 0);
124             if isempty(null_f3)
125                 fin=0;
126             else
127                 fin = length(null_f3);
128             end
129             id_f3_k_real = id_f3_k(1:end-fin);
130             id_f2_k_real = (low:high-fin);
131             db = 1i*4*pi^2*(f1(jj)-f)*(f2(id_f2_k_real)-
132                 f)*fiberParams.GVD;
133             db = reshape(db,length(db),1);
134             db = repmat(db,1,length(z));
135             Z = repmat(z,size(db,1),1);
136             F = db.*Z;
137             F = exp(F);
138             fibb_f = repmat(fib_f,size(db,1),1);
139             fib_f1 = fib(jj,:);
140             fib_f1 = sqrt(fib_f1);
141             fib_f1 = repmat(fib_f1,size(db,1),1);
142             fib_f2 = fib(id_f2_k_real,:);
143             fib_f2 = sqrt(fib_f2);
144             fib_f3 = fib(id_f3_k_real,:);
145             fib_f3 = sqrt(fib_f3);

```



```

143         test = fib_f1.*fibb_f.*fib_f2./fib_f3;
144         F = F.*test;
145         F = trapz(z,F,2);
146         F = abs(F).^2;
147         F = F.';
148         A = gtx_f2(id_f2_k_real);
149         B = gtx_f2(id_f3_k_real);
150         freq = f2(id_f2_k_real);
151         I2(jj)=I2(jj)+trapz(freq,F.*gtx_f1(jj).*A.*B
            ,2);
152     end
153 end
154
155
156
157 end
158
159 end
160 I2_gn(1:h) = 0;
161 I2 = I2 +I2_gn;
162 I1=trapz(f1,I2);
163 save('i2_20180617.mat','I2');
164 GGf=16/27*fiberParams.gamma^2*I1*1e-12;
165 end

```

System Parameters

Here it is shown the system parameters file

```

1 % @Author: Angel Alonso
2 % Company: Coriant GmbH Muenchen
3 % Univesity: Politecnico di Torino & Universidad Politecnica
   de Madrid
4
5 % This returns signal and sys structs
6
7
8 %
   =====
9 %                               INPUT
10 %
   =====

```

```

11 %
12 % delta_f: Type: Double. Spacing frequency
13 % nCh: Type: Double. Number of samples
14 % power: Type: Double. Input Channel power
15 % article: Type: String. switch flag
16 %
17 %

```

```

18 %                                OUTPUT
19 %

```

```

20 %
21 % signal: Type: Struct. It contains signal parameters (power
    , roll-off, Bw and Rs)
22 % sys: Type: Struct. It contains system parameters (number
    of channels,
23 % span length, number of spans, bw, EDFA Gain, Figure Noise)
24 function [signal, sys]=systemParams(delta_f, nCh, power, article
    )
25
26 switch article
27     case 'CORIANT3'
28         c = 299792458; % Speed Light in vacuum
29
30         % WDM Signal parameters
31         roll_off = 0.15; % Roll-off factor
32         Rs = 32e-3; % Channel Rate [TBaud/s]
33         power=10.^(power/10);
34         BwCh = (1+roll_off)*Rs; % Channel Bandwidth
35         signal=struct('roll_off', roll_off, 'Rs', Rs, 'power',
            power, 'BwCh', BwCh);
36
37         % System parameters
38         fo = 191.1250; % Central Wavelegnth [nm]
39         lo = c/(fo*10^3); % Central Frquency [THz]
40         F = 5; % Noise Figure [dB]
41         Flin = 10^(F/10); % Noise Figure in Linear Scale
42         G = 20; % EDFA Amplifier Gain [dB]
43         Glin = 10^(G/10); % EDFA Amplifier Gain in Linear
            Scale
44         Ls = 100.8; % span length [km]

```

```

45      %delta_f = 32e-3; % Frequency spacing between
        channels Nyquist delta_f=BwCh. Usually delta_f
        =3/2 BwCh
46      Bw = nCh*delta_f;
47      Ns = 1; % Spans number
48      fCh=(-Bw/2-delta_f/2)*ones(1,nCh)+delta_f*(1:nCh);
49      sys=struct('Flin',Flin,'GLin',Glin,'Ls',Ls,'nCh',nCh
        , 'delta_f',delta_f,'Bw',Bw,'Ns',Ns,'fCh',fCh,'fo'
        ,fo);
50  case 'CORIANT2'
51      c = 299792458; % Speed Light in vacuum
52
53      % WDM Signal parameters
54      roll_off = 0.15; % Roll-off factor
55      Rs = 32e-3; % Channel Rate [TBaud/s]
56      power=10.^(power/10);
57      BwCh = (1+roll_off)*Rs; % Channel Bandwidth
58      signal=struct('roll_off',roll_off,'Rs',Rs,'power',
        power,'BwCh',BwCh);
59
60      % System parameters
61      fo = 191.6250; % Central Wavelegnth [nm]
62      lo = c/(fo*10^3); % Central Frquency [THz]
63      F = 5; % Noise Figure [dB]
64      Flin = 10^(F/10); % Noise Figure in Linear Scale
65      G = 20; % EDFA Amplifier Gain [dB]
66      Glin = 10^(G/10); % EDFA Amplifier Gain in Linear
        Scale
67      Ls = 100.8; % span length [km]
68      %delta_f = 32e-3; % Frequency spacing between
        channels Nyquist delta_f=BwCh. Usually delta_f
        =3/2 BwCh
69      Bw = nCh*delta_f;
70      Ns = 1; % Spans number
71      fCh=(-Bw/2-delta_f/2)*ones(1,nCh)+delta_f*(1:nCh);
72      sys=struct('Flin',Flin,'GLin',Glin,'Ls',Ls,'nCh',nCh
        , 'delta_f',delta_f,'Bw',Bw,'Ns',Ns,'fCh',fCh,'fo'
        ,fo);
73  case 'CORIANT'
74      c = 299792458; % Speed Light in vacuum
75
76      % WDM Signal parameters
77      roll_off = 0.15; % Roll-off factor
78      Rs = 32e-3; % Channel Rate [TBaud/s]

```

```

79     power=10.^(power/10);
80     BwCh = (1+roll_off)*Rs; % Channel Bandwidth
81     signal=struct('roll_off',roll_off,'Rs',Rs,'power',
        power,'BwCh',BwCh);
82
83     % System parameters
84     fo = 193.15; % Central Wavelegnth [nm]
85     lo = c/(fo*10^3); % Central Frquency [THz]
86     F = 5; % Noise Figure [dB]
87     Flin = 10^(F/10); % Noise Figure in Linear Scale
88     G = 16.2893; % EDFA Amplifier Gain [dB]
89     Glin = 10^(G/10); % EDFA Amplifier Gain in Linear
        Scale
90     Ls = 100.8; % span length [km]
91     %delta_f = 32e-3; % Frequency spacing between
        channels Nyquist delta_f=BwCh. Usually delta_f
        =3/2 BwCh
92     Bw = nCh*delta_f;
93     Ns = 1; % Spans number
94     fCh=(-Bw/2-delta_f/2)*ones(1,nCh)+delta_f*(1:nCh);
95     sys=struct('Flin',Flin,'Glin',Glin,'Ls',Ls,'nCh',nCh
        ,'delta_f',delta_f,'Bw',Bw,'Ns',Ns,'fCh',fCh,'fo',
        ,fo);
96 case 'ORANGE'
97     % Modelling the Impact of SRS on NLI Generation in
        Commercial
98     % Equipment: an Experimental investigation
99     % Mattia Cantono
100    % Physical parameters/constants
101    c = 299792458; % Speed Light in vacuum
102
103    % WDM Signal parameters
104    roll_off = 0.15; % Roll-off factor
105    Rs = 32e-3; % Channel Rate [TBaud/s]
106    power=10.^(power/10);
107    BwCh = (1+roll_off)*Rs; % Channel Bandwidth
108    signal=struct('roll_off',roll_off,'Rs',Rs,'power',
        power,'BwCh',BwCh);
109
110    % System parameters
111    fo = 193.9778; % Central Wavelegnth [nm]
112    lo = c/(fo*10^3); % Central Frquency [THz]
113    F = 5; % Noise Figure [dB]
114    Flin = 10^(F/10); % Noise Figure in Linear Scale

```

```

115     G = 16; % EDFA Amplifier Gain [dB]
116     Glin = 10^(G/10); % EDFA Amplifier Gain in Linear
        Scale
117     Ls = 80; % span length [km]
118     %delta_f = 32e-3; % Frequency spacing between
        channels Nyquist delta_f=BwCh. Usually delta_f
        =3/2 BwCh
119     Bw = nCh*delta_f;
120     Ns = 1; % Spans number
121     fCh=(-Bw/2-delta_f/2)*ones(1,nCh)+delta_f*(1:nCh);
122     sys=struct('Flin',Flin,'Glin',Glin,'Ls',Ls,'nCh',nCh
        , 'delta_f',delta_f,'Bw',Bw,'Ns',Ns,'fCh',fCh,'fo'
        ,fo);
123 case 'TIM'
124
125     c = 299792458; % Speed Light in vacuum
126
127     % WDM Signal parameters
128     roll_off = 0.2; % Roll-off factor
129     Rs = 57.24e-3; % Channel Rate [TBaud/s]
130     power=10.^(power/10);
131     BwCh = (1+roll_off)*Rs; % Channel Bandwidth
132     signal=struct('roll_off',roll_off,'Rs',Rs,'power',
        power,'BwCh',BwCh);
133
134     % System parameters
135     fo = 193.4;
136     F = 6.5; % Noise Figure [dB]
137     Flin = 10^(F/10); % Noise Figure in Linear Scale
138     G = 24.1740; % EDFA Amplifier Gain [dB]
139     Glin = 10^(G/10); % EDFA Amplifier Gain in Linear
        Scale
140     Ls = 76.5; % span length [km]
141     %delta_f = 32e-3; % Frequency spacing between
        channels Nyquist delta_f=BwCh. Usually delta_f
        =3/2 BwCh
142     Bw = nCh*delta_f;
143     Ns = 10; % Spans number
144     fCh=(-Bw/2-delta_f/2)*ones(1,nCh)+delta_f*(1:nCh);
145     sys=struct('Flin',Flin,'Glin',Glin,'Ls',Ls,'nCh',nCh
        , 'delta_f',delta_f,'Bw',Bw,'Ns',Ns,'fCh',fCh,'fo'
        ,fo);
146
147 case 'MID.2'

```

```

148      % Modelling the Impact of SRS on NLI Generation in
          Commercial
149      % Equipment: an Experimental investigation
150      % Mattia Cantono
151      % Physical parameters/constants
152      c = 299792458; % Speed Light in vacuum
153
154      % WDM Signal parameters
155      roll_off = 0.2; % Roll-off factor
156      Rs = 32e-3; % Channel Rate [TBaud/s]
157      power=10.^(power/10);
158      BwCh = (1+roll_off)*Rs; % Channel Bandwidth
159      signal=struct('roll_off',roll_off,'Rs',Rs,'power',
          power,'BwCh',BwCh);
160
161      % System parameters
162      lo = 1545.3; % Central Wavelegnth [nm]
163      fo = c/(lo*10^3); % Central Frquency [THz]
164      F = 5; % Noise Figure [dB]
165      Flin = 10^(F/10); % Noise Figure in Linear Scale
166      G = 16; % EDFA Amplifier Gain [dB]
167      Glin = 10^(G/10); % EDFA Amplifier Gain in Linear
          Scale
168      Ls = 80; % span length [km]
169      %delta_f = 32e-3; % Frequency spacing between
          channels Nyquist delta_f=BwCh. Usually delta_f
          =3/2 BwCh
170      Bw = nCh*delta_f;
171      Ns = 1; % Spans number
172      fCh=(-Bw/2-delta_f/2)*ones(1,nCh)+delta_f*(1:nCh);
173      sys=struct('Flin',Flin,'Glin',Glin,'Ls',Ls,'nCh',nCh
          ,'delta_f',delta_f,'Bw',Bw,'Ns',Ns,'fCh',fCh,'fo'
          ,fo);
174
175      case '111Fig4a'
176          % 111 stands for the article called:
177          % Modeling of the Impact of Nonlinear Propagation
          Effects
178          % in Uncompensated Optical Coherent Transmission
          Links
179          % Figure 4.a
180          % First Author: Andrea Carena
181
182          % Physical parameters/constants

```

```

183     c = 299792458; % Speed Light in vacuum
184
185     % WDM Signal parameters
186     roll_off = 0.3; % Roll-off factor
187     Rs = 32e-3; % Channel Rate [TBaud/s]
188     power=10.^(power/10);
189     BwCh = (1+roll_off)*Rs; % Channel Bandwidth
190     signal=struct('roll_off',roll_off,'Rs',Rs,'power',
        power,'BwCh',BwCh);
191
192     % System parameters
193     lo = 1550; % Central Wavelegnth [nm]
194     fo = c/(lo*10^3); % Central Frquency [THz]
195     F = 5; % Noise Figure [dB]
196     Flin = 10^(F/10); % Noise Figure in Linear Scale
197     G = 22; % EDFA Amplifier Gain [dB]
198     Glin = 10^(G/10); % EDFA Amplifier Gain in Linear
        Scale
199     Ls = 100; % span length [km]
200     %delta_f = 32e-3; % Frequency spacing between
        channels Nyquist delta_f=BwCh. Usually delta_f
        =3/2 BwCh
201     Bw = nCh*delta_f;
202     Ns = 1; % Spans number
203     fCh=(-Bw/2-delta_f/2)*ones(1,nCh)+delta_f*(1:nCh);
204     sys=struct('Flin',Flin,'Glin',Glin,'Ls',Ls,'nCh',nCh
        , 'delta_f',delta_f,'Bw',Bw,'Ns',Ns,'fCh',fCh,'fo'
        ,fo);
205
206     case '111Fig4b'
207         % 111 stands for the article called:
208         % Modeling of the Impact of Nonlinear Propagation
        Effects
209         % in Uncompensated Optical Coherent Transmission
        Links
210         % Figure 4.b
211         % First Author: Andrea Carena
212
213         % Physical parameters/constants
214         c = 299792458; % Speed Light in vacuum
215
216         % WDM Signal parameters
217         roll_off = 0.3; % Roll-off factor
218         Rs = 32e-3; % Channel Rate [TBaud/s]

```

```

219     power=10.^(power/10);
220     BwCh = (1+roll_off)*Rs; % Channel Bandwidth
221     signal=struct('roll_off',roll_off,'Rs',Rs,'power',
        power,'BwCh',BwCh);
222
223     % System parameters
224     lo = 1550; % Central Wavelegnth [nm]
225     fo = c/(lo*10^3); % Central Frquency [THz]
226     F = 5; % Noise Figure [dB]
227     Flin = 10^(F/10); % Noise Figure in Linear Scale
228     G = 18; % EDFA Amplifier Gain [dB]
229     Glin = 10^(G/10); % EDFA Amplifier Gain in Linear
        Scale
230     Ls = 100; % span length [km]
231     %delta_f = 32e-3; % Frequency spacing between
        channels Nyquist delta_f=BwCh. Usually delta_f
        =3/2 BwCh
232     Bw = nCh*delta_f;
233     Ns = 1; % Spans number
234     fCh=(-Bw/2-delta_f/2)*ones(1,nCh)+delta_f*(1:nCh);
235     sys=struct('Flin',Flin,'Glin',Glin,'Ls',Ls,'nCh',nCh
        ,'delta_f',delta_f,'Bw',Bw,'Ns',Ns,'fCh',fCh,'fo',
        ,fo);
236 case 'C+LW11.2'
237     % c+L W11.2 stands for the article called:
238     % "DP-16QAM Based Coded Modulation Transmission
239     % in C+L Band System at Transoceanic Distance"
240     % First Author: H.Zhang
241
242     % Physical parameters/constants
243     c = 299792458; % Speed Light in vacuum
244
245     % WDM Signal parameters
246     roll_off = 0.01; % Roll-off factor
247     Rs = 34.18e-3; % Channel Rate [TBaud/s]
248     power=10.^(power/10);
249     BwCh = (1+roll_off)*Rs; % Channel Bandwidth
250     signal=struct('roll_off',roll_off,'Rs',Rs,'power',
        power,'BwCh',BwCh);
251
252     % System parameters
253     lo = 1546; % Central Wavelegnth [nm]
254     fo = c/(lo*10^3); % Central Frquency [THz]
255     F = 5.2; % Noise Figure [dB]

```



```

256     Flin = 10^(F/10); % Noise Figure in Linear Scale
257     G = 9.9; % EDFA Amplifier Gain [dB]
258     Glin = 10^(G/10); % EDFA Amplifier Gain in Linear
        Scale
259     Ls = 55; % span length [km]
260     %delta_f = 32e-3; % Frequency spacing between
        channels Nyquist delta_f=BwCh. Usually delta_f
        =3/2 BwCh
261     Bw = nCh*delta_f;
262     Ns = 121; % Spans number
263     fCh=(-Bw/2-delta_f/2)*ones(1,nCh)+delta_f*(1:nCh);
264     sys=struct('Flin',Flin,'Glin',Glin,'Ls',Ls,'nCh',nCh
        , 'delta_f',delta_f,'Bw',Bw,'Ns',Ns,'fCh',fCh,'fo'
        ,fo);
265 case '7fiber_SSMF'
266     % 111 stands for the article called:
267     % Modeling of the Impact of Nonlinear Propagation
        Effects
268     % in Uncompensated Optical Coherent Transmission
        Links
269     % Figure 4.b
270     % First Author: Andrea Carena
271
272     % Physical parameters/constants
273     c = 299792458; % Speed Light in vacuum
274
275     % WDM Signal parameters
276     roll_off = 0.01; % Roll-off factor
277     Rs = 15.625e-3; % Channel Rate [TBaud/s]
278     power=10.^(power/10);
279     BwCh = (1+roll_off)*Rs; % Channel Bandwidth
280     signal=struct('roll_off',roll_off,'Rs',Rs,'power',
        power,'BwCh',BwCh);
281
282     % System parameters
283     lo = 1550; % Central Wavelegnth [nm]
284     fo = c/(lo*10^3); % Central Frquency [THz]
285     F = 5.5; % Noise Figure [dB]
286     Flin = 10^(F/10); % Noise Figure in Linear Scale
287     G = 9.7014; % EDFA Amplifier Gain [dB]
288     Glin = 10^(G/10); % EDFA Amplifier Gain in Linear
        Scale
289     Ls = 51.06; % span length [km]

```

```

290         %delta_f = 32e-3; % Frequency spacing between
           channels Nyquist delta_f=BwCh. Usually delta_f
           =3/2 BwCh
291     Bw = nCh*delta_f;
292     Ns = 1; % Spans number
293     fCh=(-Bw/2-delta_f/2)*ones(1,nCh)+delta_f*(1:nCh);
294     sys=struct('Flin',Flin,'GLin',Glin,'Ls',Ls,'nCh',nCh
           , 'delta_f',delta_f,'Bw',Bw,'Ns',Ns,'fCh',fCh,'fo'
           ,fo);
295
296 end

```

Power Profile Evolution

Here it is shown power profile evolution computation.

```

1
2 % @Author: Angel Alonso
3 % Company: Coriant GmbH Muenchen
4 % Univesity: Politecnico di Torino & Universidad Politecnica
   de Madrid
5
6 % This returns power evolution profile rho(z,f)^2.
7 %
   =====
8 %                                     INPUT
9 %
   =====
10 %
11 % power: Type: Double array. channel power
12 % Cr: Type: Double. Raman Gain [1/W/km]
13 % fMax: Type: Double. Frequency shift , so Raman Gain is
   equal to Cr [THz]
14 % alpha: Type: Double array. field attenuation factor [1/km
   ]
15 % delta_f: Type: Double. Frequency spacing [THz]
16 % Ls: Type: Double. Span length [km]
17 % n: Type: Double. Channel index
18 % nCh: Type: Integer. Number of Channels
19 % J: Type: Double. Total Input Power [W]
20 % zeta_samples: Double Type. It set the number of samples of
   z

```

```

21 %
22 %

```

```

23 %                                OUTPUT
24 %

```

```

25 %
26 % rho: Type: Double Array. Power evolution profile.
27
28 function rho = fiber_var(power,Cr,fMax,alpha,delta_f,Ls,n,
    nCh,J, zeta_samples)
29
30
31 % First of all we compute the slope of the Raman Gain
    Profile (This slope
32 % includes 1/2 Aeff
33 % m = g'/2A
34 % slope = 4.9e-18;
35 % Aeff = 50e-12;
36 % G = slope/2/Aeff*delta_f*1000;
37 m = Cr/fMax;
38 G = m*delta_f;
39 z = linspace(0,Ls, zeta_samples);
40 Leff = (1-exp(-2*alpha(n).*z))./(2*alpha(n)); %
    Effective Length
41 A = zeros(nCh,length(z));
42 for ii = 1:nCh
43     A(ii,:) = A(ii,:) + (power(ii).*exp(G*J*(ii-1).*
        Leff));
44 end
45 A = sum(A,1);
46 rho = J*exp(-2*alpha(n).*z).*exp(G*J*(n-1).* Leff)./A;
47 end

```

Power Profile Evolution at the end of the link

Power evolution at the end of the link file.

```

1
2 % @Author: Angel Alonso
3 % Company: Coriant GmbH Muenchen

```

```

4 % Univesity: Politecnico di Torino & Universidad Politecnica
   de Madrid
5
6 % This returns power profile at the end of the span rh(Ls,f)
   ^2
7 %

```

```

8 %                                     INPUT
9 %

```

```

10 %
11 % power: Type: Double array. channel power
12 % Cr: Type: Double. Raman Gain [1/W/km]
13 % fMax: Type: Double. Frequency shift , so Raman Gain is
   equal to Cr [THz]
14 % alpha: Type: Double array. field attenuation factor [1/km
   ]
15 % delta_f: Type: Double. Frequency spacing [THz]
16 % Ls: Type: Double. Span length [km]
17 % n: Type: Double. Channel index
18 % nCh: Type: Integer. Number of Channels
19 % J: Type: Double. Total Input Power [W]
20 %

```

```

21 %                                     OUTPUT
22 %

```

```

23 %
24 % rho: Type: Double. Power profile at the of the span
25
26
27 function rho = fiber(power,Cr,fMax,alpha,delta_f,Ls,n,nCh,J)
28
29
30     % First of all we compute the slope of the Raman Gain
       Profile (This slope
31     % includes 1/2Aeff
32     % m = g'/2A
33     % slope = 4.9e-18;
34     % Aeff = 50e-12;

```

```

35 %      G = slope/2/Aeff*delta_f*1000;
36 m = Cr/fMax;
37 G = m*delta_f;
38 Leff = (1-exp(-2*alpha*Ls))/(2*alpha); % Effective
      Length
39 A = 0;
40 for ii = 1:nCh
41     A = A + (power(ii)*exp(G*J*(ii-1)*Leff));
42 end
43 rho = J*exp(-2*alpha*Ls)*exp(G*J*(n-1)*Leff)/A;
44
45
46 end

```

Bibliography

- [1] M. Cantono, D. Piori, A. Ferrari, C. Catanese, J. Thouras, J.-L. Augé, and V. Curri, “On the interplay of nonlinear interference generation with stimulated raman scattering for qot estimation,” *Journal of Lightwave Technology*, 2018.
- [2] P. Poggiolini and R. Gaudino, “Chapter 5: Systems using in-line optical amplifiers. course: Optical and wireless communication,” *Optical Communications Group of Politecnico di Torino*.
- [3] P. Poggiolini and R. Gaudino, “Chapter 7: Coherent systems. course: Optical and wireless communication,” *Optical Communications Group of Politecnico di Torino*.
- [4] J. K.Fischer, M. Cantono, V. Curri, R.-P. Braun, N. Costa, J. Pedro, E. Pincemin, P. Doare, C. Le Bouette, and A. Napoli, “Maximizing the capacity of installed optical fiber infrastructure via wideband transmission,” *ICTON*, 2018.
- [5] G. Khanna, T. Rahman, E. De Man, E. Riccardi, A. Pagano, A. C. Piat, B. Spinnler, S. Calabro, D. Rafique, U. Feiste, *et al.*, “Comparison of single carrier 200g 4qam, 8qam and 16qam in a wdm field trial demonstration over 612 km ssmf,” in *ECOC 2016; 42nd European Conference on Optical Communication; Proceedings of*, pp. 1–3, VDE, 2016.
- [6] G. Grammel, V. Curri, and J.-L. Auge, “Physical simulation environment of the telecommunications infrastructure project (tip),” in *Optical Fiber Communication Conference*, pp. M1D–3, Optical Society of America, 2018.
- [7] P. Poggiolini and R. Gaudino, “Chapter 6: Fiber chromatic dispersion. course: Optical and wireless communication,” *Optical Communications Group of Politecnico di Torino*.
- [8] P. Poggiolini and R. Gaudino, “Chapter 8: Impact of non-linear propagation effects on uncompensated coherent systems. course: Optical and wireless communication,” *Optical Communications Group of Politecnico di Torino*.
- [9] V. Curri, “Raman scattering. course: Optical transmission ii,” *Optical Communications Group of Politecnico di Torino*.

BIBLIOGRAPHY

- [10] D. Christodoulides and R. Jander, “Evolution of stimulated raman crosstalk in wavelength division multiplexed systems,” *IEEE Photonics Technology Letters*, vol. 8, no. 12, pp. 1722–1724, 1996.
- [11] P. Poggiolini, G. Bosco, A. Carena, V. Curri, Y. Jiang, and F. Forghieri, “The gn-model of fiber non-linear propagation and its applications,” *Journal of light-wave technology*, vol. 32, no. 4, pp. 694–721, 2014.
- [12] P. Poggiolini, “The gn model of non-linear propagation in uncompensated coherent optical systems,” *Journal of Lightwave Technology*, vol. 30, no. 24, pp. 3857–3879, 2012.
- [13] P. Poggiolini, G. Bosco, A. Carena, V. Curri, Y. Jiang, and F. Forghieri, “A detailed analytical derivation of the gn model of non-linear interference in coherent optical transmission systems,” *arXiv preprint arXiv:1209.0394*, 2012.
- [14] A. Carena, V. Curri, G. Bosco, P. Poggiolini, and F. Forghieri, “Modeling of the impact of nonlinear propagation effects in uncompensated optical coherent transmission links,” *Journal of Lightwave Technology*, vol. 30, no. 10, pp. 1524–1539, 2012.
- [15] M. Cantono, J.-L. Auge, and V. Curri, “Modelling the impact of srs on nli generation in commercial equipment: an experimental investigation,” in *Optical Fiber Communication Conference*, pp. M1D–2, Optical Society of America, 2018.