

TABLE DES MATIÈRES

	Page
INTRODUCTION	1
CHAPITRE 1 EFFETS DE POLARISATION DANS DES LIAISONS OPTIQUES	5
1.1 Polarisation d'une onde électromagnétique	5
1.2 États de polarisation	7
1.2.1 Le vecteur Stokes	7
1.2.2 Représentation du vecteur Stokes sur la sphère de Poincaré	9
1.2.3 Fluctuation des états de polarisation	10
1.3 Dispersion des modes de polarisation	12
1.3.1 Causes de la PMD	12
1.3.2 Paramètres de la PMD	14
1.4 Perte dépendante de polarisation	16
CHAPITRE 2 MONITORING DE DEUX LIAISONS OPTIQUES COHÉRENTES WDM	19
2.1 Introduction	19
2.2 Revue des expériences de <i>monitoring</i> et de caractérisation des effets de polarisation sur différentes installations optiques	19
2.2.1 <i>Monitoring</i> et caractérisation des effets de polarisation avec un transpondeur cohérent sur des liaisons optiques cohérentes enterrées	19
2.2.2 <i>Monitoring</i> et caractérisation des effets de polarisation avec un polarimètre sur différentes installations optiques	21
2.3 Description des expériences de <i>monitoring</i> réalisées sur deux liaisons optiques cohérentes	23
2.3.1 Expérience de <i>monitoring</i> d'une liaison optique cohérente du réseau Canarie	24
2.3.2 Expérience de <i>monitoring</i> d'une liaison optique cohérente du réseau Verizon	25
2.4 Transpondeur cohérent DP-QPSK à 40 Gbit/s	27
2.4.1 Modulation DP-QPSK à 40 Gbit/s	27
2.4.2 Transmetteur cohérent à 40 Gbit/s	29
2.4.3 Récepteur cohérent DP-QPSK à 40 Gbit/s	30
2.4.3.1 Schéma du récepteur cohérent DP-QPSK	30
2.4.3.2 Module de traitement du signal numérique	32
2.5 Fonction « polarimètre » du synthétiseur de polarisation Agilent N7786B	34
2.6 Méthode d'acquisition des données	37
2.6.1 Méthode d'acquisition des données sur la liaison du réseau CANARIE ..	37
2.6.2 Méthode d'acquisition des données sur la liaison du réseau Verizon	38
2.6.2.1 Programme d'acquisition des données avec le transpondeur cohérent	38
2.6.2.2 Programme d'acquisition des données avec le polarimètre	40

CHAPITRE 3	MONITORING ET CARACTERISATION DES VALEURS DE DGD ET DE PDL AVEC UN TRANSPONDEUR COHÉRENT DP-QPSK À 40 GBIT/S	41
3.1	Introduction.....	41
3.2	Traitement des données enregistrées par le transpondeur cohérent.....	42
3.3	Évolution temporelle des valeurs de DGD et des valeurs de PDL	45
3.3.1	Méthode de calcul et d'analyse de l'évolution temporelle des valeurs de DGD et de PDL	45
3.3.2	Analyse des courbes d'évolution temporelle de DGD et de PDL	47
3.4	Moyenne et écart type des valeurs de DGD et de PDL	50
3.4.1	Méthode de calcul des moyennes et des écarts types des valeurs de DGD et de PDL	51
3.4.2	Analyse des courbes d'évolution temporelle des moyennes et des écarts types des valeurs de DGD et de PDL pour chaque minute de mesure	52
3.5	Autocorrélation des valeurs de DGD et de PDL.....	57
3.5.1	Méthode de calcul de la fonction d'autocorrélation des valeurs de DGD et de PDL	58
3.5.1.1	Étape 1 : calcul de la moyenne des valeurs de DGD et de PDL pour chaque seconde de mesure	59
3.5.1.2	Étape 2 : filtrage super gaussien des valeurs de $\langle DGD \rangle$ et des valeurs de $\langle PDL \rangle$ pour chaque seconde de mesure.....	59
3.5.1.3	Étape 3 : calcul de la fonction d'autocorrélation des valeurs de $\langle DGD \rangle$ filtrées et des valeurs de $\langle PDL \rangle$ filtrées.....	64
3.5.1.4	Étape 4 : calcul du temps de corrélation des valeurs de $\langle DGD \rangle$ et de $\langle PDL \rangle$ filtrées.....	66
3.5.2	Analyse des temps de corrélation des valeurs de DGD et de PDL.....	67
3.6	Étude de la fonction de densité des valeurs de DGD mesurées sur le lien du réseau Verizon.....	70
3.6.1	Méthode de calcul et d'ajustement de la fonction de densité de probabilité des valeurs de DGD	70
3.6.1.1	Étape 1 : calcul de l'histogramme normalisé des valeurs de DGD	72
3.6.1.2	Étape 2 : ajustement de l'histogramme des valeurs de DGD.....	74
3.6.1.3	Étape 3 : évaluation de l'ajustement par le coefficient de corrélation de Pearson.....	76
3.6.2	Ajustement de l'histogramme des valeurs de DGD par la fonction de densité de Maxwell	77
3.7	Conclusion du chapitre	80

CHAPITRE 4	MONITORING ET CARACTÉRISATION DES VITESSES DE ROTATION ANGULAIRE DES ÉTATS DE POLARISATION AVEC UN POLARIMÈTRE.....	81
4.1	Introduction.....	81
4.2	Traitement des données enregistrées par le polarimètre.....	82
4.3	Étude de la fonction de densité des vitesses de rotation angulaire des états de polarisation.....	86
4.3.1	Méthode de calcul et d'ajustement de l'histogramme des vitesses de rotation angulaire des états de polarisation.....	87
4.3.1.1	Étape 1 : calcul de l'histogramme normalisé des vitesses de rotation angulaire.....	88
4.3.1.2	Étape 2 : ajustement de l'histogramme des vitesses de rotation angulaire.....	90
4.3.1.3	Étape 3 : évaluation de l'ajustement de l'histogramme des vitesses de rotation angulaire des états de polarisation.....	93
4.3.2	Ajustement de l'histogramme des vitesses de rotation angulaire des états de polarisation par la fonction de densité de Rayleigh.....	95
4.3.3	Ajustement de l'histogramme des vitesses de rotation angulaire supérieures à 1.1 rad/ms par la fonction de densité exponentielle.....	98
4.3.4	Ajustement de l'histogramme entier des vitesses de rotation angulaire par une fonction de densité double.....	100
4.3.4.1	Présentation de la fonction de densité double f_{GEI} sur l'histogramme entier des vitesses de rotation angulaire des états de polarisation.....	100
4.3.4.2	Méthode et analyse de l'ajustement de l'histogramme entier des vitesses de rotation angulaire par la fonction de densité double f_{GEI}	102
4.4	Caractérisation des paramètres des effets transitoires de polarisation.....	105
4.4.1	Méthode de calcul des paramètres des effets transitoires de polarisation.....	105
4.4.2	Méthode d'analyse statistique des paramètres caractéristiques des effets transitoires de polarisation.....	109
4.4.3	Analyse des valeurs maximales d'élasticité des effets transitoires de polarisation.....	110
4.4.4	Analyse des valeurs de durée des effets transitoire de polarisation.....	113
4.4.5	Analyse des valeurs d'intervalle de temps entre deux effets transitoires de polarisation adjacents.....	115
4.5	Conclusion du chapitre.....	117
	CONCLUSION.....	119
ANNEXE I	INTERFACE D'ACQUISITION TELNET DES PARAMÈTRES DE <i>MONITORING</i> ET DE DIAGNOSTIQUE AVEC UN TRANSPONDEUR COHÉRENT DP-QPSK.....	125

ANNEXE II	ANALYSEUR DE TEXTE PHP POUR L'EXTRACTION DES DONNÉES DANS LES FICHIERS TEXTES	127
ANNEXE III	STRUCTURE DES FICHIERS DE MESURE GÉNÉRÉS PAR LE PROGRAMME D'ACQUISITION AVEC LE TRANSPONDEUR COHÉRENT	129
ANNEXE IV	STRUCTURE DES FICHIERS DE MESURE GÉNÉRÉS PAR LE PROGRAMME D'ACQUISITION AVEC LE POLARIMÈTRE	131
ANNEXE V	OUTIL DE TRAITEMENT DES DONNÉES ENREGISTRÉES AVEC LE TRANSPONDEUR COHÉRENT DP-QPSK À 40 GBIT/S	133
ANNEXE VI	ÉTAPE 1 : CRÉATION D'UN RÉPERTOIRE DE FICHIERS CSV ENREGISTRÉS AVEC LE TRANSPONDEUR COHÉRENT	137
ANNEXE VII	ÉTAPE 2.1 : LECTURE BINAIRE DES FICHIERS CSV ET CONVERSION EN FORMAT TEXTE NUMÉRIQUE	139
ANNEXE VIII	ÉTAPE 2.2 : CONVERSION DES FICHIERS TEXTES NUMÉRIQUES EN MATRICES MATLAB	141
ANNEXE IX	ÉTAPE 3 : CALCUL ET INSERTION DES VALEURS D'HORODATAGE	143
ANNEXE X	ÉTAPE 4 : SUPPRESSION DES VALEURS TRANSITOIRES CORRESPONDANT À LA RÉINITIALISATION DE LA CARTE OCLD	145
ANNEXE XI	ÉTAPE 5 : CONCEPTION D'UNE STRUCTURE DE DONNÉES INDEXÉE PAR « CLÉ » DES MESURES DU TRANSPONDEUR COHÉRENT	147
ANNEXE XII	CONCATÉNATION DES MATRICES DES VALEURS DE DGD ET DE PDL À PARTIR DE LA STRUCUTURE DE DONNÉES INDEXÉE	149
ANNEXE XIII	CALCUL DE LA DÉRIVÉE TEMPORELLES DES VALEURS DE DGD ET DE PDL ENREGISTRÉES DANS LA STRUCTURE DE DONNÉES INDEXÉE	151
ANNEXE XIV	AFFICHAGE DES COURBES D'ÉVOLUTION TEMPORELLE DES VALEURS DE DGD ET DES VALEURS DE PDL	153

ANNEXE XV	CALCUL DE LA MOYENNE ET DE L'ÉCART TYPE DES VALEURS DE DGD ET DE PDL À CHAQUE MINUTE DE MESURE.....	155
ANNEXE XVI	CALCUL DE LA DÉRIVÉE TEMPORELLE DE LA MOYENNE ET DE L'ÉCART TYPE DES VALEURS DE DGD ET DE PDL À CHAQUE MINUTE DE MESURE.....	157
ANNEXE XVII	AFFICHAGE DE LA MOYENNE ET DE L'ÉCART TYPE DES VALEURS DE DGD ET DE PDL AINSI QUE LEURS VALEURS DE DÉRIVÉES TEMPORELLES RESPECTIVES POUR CHAQUE MINUTE DE MESURE	159
ANNEXE XVIII	ÉTAPE 1 : CALCUL DE LA MOYENNE DES VALEURS DE DGD ET DE PDL POUR CHAQUE SECONDE DE MESURE	165
ANNEXE XIX	ÉTAPE 2 : FILTRAGE SUPER GAUSSIEN DES VALEURS DE <DGD> ET DE <PDL> POUR CHAQUE SECONDE DE MESURE	167
ANNEXE XX	ÉTAPE 3 : CALCUL DE LA FONCTION D'AUTOCORRÉLATION DES VALEURS DE <DGD> FILTRÉES ET DES VALEURS DE <PDL> FILTRÉES.....	169
ANNEXE XXI	ÉTAPE 4 : CALCUL DU TEMPS DE CORRÉLATION DES VALEURS DE <DGD> ET DE <PDL> FILTRÉES	171
ANNEXE XXII	AFFICHAGE ET SUPERPOSITION DES FONCTIONS D'AUTOCORRÉLATIONS DE DGD ET DE PDL SUR LA LIAISON DU RÉSEAU CANARIE ET VERIZON.....	173
ANNEXE XXIII	RÉSULTAT DE LA FONCTION D'AUTOCORRÉLATION APPLIQUÉE SUR LES VALEURS DE <PDL> FILTRÉES.....	177
ANNEXE XXIV	CALCUL ET AFFICHAGE DE LA FONCTION DE DENSITÉ DE PROBABILITÉ DES VALEURS DE DGD MESURÉES AVEC LE TRANSPONDEUR COHÉRENT	179
ANNEXE XXV	AJUSTEMENT DE LA FONCTION DE DENSITÉ DE PROBABILITÉ DES VALEURS DE DGD PAR LA FONCTION DE DENSITÉ DE MAXWELL	183
ANNEXE XXVI	MÉTHODE DE TRAPÈZE POUR LE CALCUL DE L'AIRE D'UN HISTOGRAMME.....	187

ANNEXE XXVII	PRÉSENTATION DE L'ALGORITHME DE LEVENBERG-MARQUARDT	189
ANNEXE XXVIII	APPROXIMATION DE L'ERREUR STANDARD DU PARAMÈTRE D'UNE FONCTION DE DENSITÉ DE PROBABILITÉ	193
ANNEXE XXIX	TABLE DE LA LOI DE STUDENT	195
ANNEXE XXX	OUTIL DE TRAITEMENT DES DONNÉES ENREGISTRÉES AVEC LE POLARIMÈTRE	197
ANNEXE XXXI	ÉTAPE 1 : CRÉATION D'UNE STRUCTURE DE DONNÉES D'ADRESSE DE FICHIERS MATLAB	201
ANNEXE XXXII	ÉTAPE 2 : EXTRACTION ET NORMALISATION DES VECTEURS STOKES	203
ANNEXE XXXIII	ÉTAPE 3 : CALCUL DES VITESSES DE ROTATION ANGULAIRE DES ÉTATS DE POLARISATION	205
ANNEXE XXXIV	ÉTAPE 4: SUPPRESSION DES VALEURS TRANSITOIRES DES VITESSES DE ROTATION ANGULAIRE DES ÉTATS DE POLARISATION	207
ANNEXE XXXV	ÉTAPE 5 : INSERTION DES VALEURS D'HORODATAGE AUX VITESSES DE ROTATION ANGULAIRE DES ÉTATS DE POLARISATION	209
ANNEXE XXXVI	ÉTAPE 6 : CONCEPTION DE LA STRUCTURE DE DONNÉES INDEXÉE « PAR CLÉ » DES MESURES DU POLARIMÈTRE	211
ANNEXE XXXVII	CALCUL DE LA FONCTION DE DENSITÉ DES VITESSES DE ROTATION ANGULAIRE DES ÉTATS DE POLARISATION	213

ANNEXE XXXVIII	AJUSTEMENT DE LA FONCTION DE DENSITÉ DES VITESSES DE ROTATION ANGULAIRE DES ÉTATS DE POLARISATION	217
ANNEXE XXXIX	CALCUL DES BARRES D'ERREURS ET DES INCERTITUDES D'UN HISTOGRAMME AVEC UNE ÉCHELLE LOGARITHMIQUE	225
ANNEXE XL	PROPRIÉTÉS DU MODÈLE <i>NONLINEARMODEL</i> DU MODULE <i>STATISTICS TOOLBOX</i>	227
ANNEXE XLI	AFFICHAGE DE LA FONCTION DE DENSITÉ DE PROBABILITÉ DES VITESSES DE ROTATION ANGULAIRE ET DES AJUSTEMENTS	229
ANNEXE XLII	DÉTECTION ET ENREGISTREMENT DES EFFETS TRANSITOIRES DE POLARISATION.....	233
ANNEXE XLIII	CALCUL DES PARAMÈTRES CARACTÉRISTIQUES DES EFFETS TRANSITOIRES DE POLARISATION	235
ANNEXE XLIV	CALCUL DE LA FONCTION DE DENSITÉ DE PROBABILITÉ DES VALEURS MAXIMALES DE VITESSE DE ROTATION ANGULAIRE ET D'ÉLASTICITÉ DES EFFETS TRANSITOIRES DE POLARISATION.....	239
ANNEXE XLV	CALCUL DE LA FONCTION DE DENSITÉ DE PROBABILITÉ CONJOINTE DES VALEURS MAXIMALES DE VITESSE DE ROTATION ANGULAIRE ET DES VALEURS DE DURÉE DES EFFETS TRANSITOIRES DE POLARISATION.....	241
ANNEXE XLVI	AFFICHAGE DE LA FONCTION DE DENSITÉ DE PROBABILITÉ CONJOINTE DES PARAMÈTRES DES EFFETS TRANSITOIRES DE POLARISATION	243
ANNEXE XLVII	CALCUL, AJUSTEMENT ET AFFICHAGE DE LA FONCTION DE DENSITÉ DE PROBABILITÉ DES INTERVALLES DE TEMPS ENTRE DEUX EFFETS TRANSITOIRES DE POLARISATION ADJACENTS	249
BIBLIOGRAPHIE		253

LISTE DES TABLEAUX

		Page
Tableau 3.1	Caractérisation des mesures de DGD, de $ dDGD/dt $, de PDL et de $ dPDL/dt $ par le calcul de la moyenne, de l'écart type et de la valeur maximale	49
Tableau 3.2	Temps de corrélation τ_{HWHM} et $\tau_{1/e}$ et de leur incertitude Δ_{HWHM} et $\Delta_{1/e}$ des valeurs de $\langle DGD \rangle$ filtrées et des valeurs de $\langle PDL \rangle$ filtrées sur le lien du réseau CANARIE	68
Tableau 3.3	Temps de corrélation τ_{HWHM} et $\tau_{1/e}$ et de leur incertitude Δ_{HWHM} et $\Delta_{1/e}$ des valeurs de $\langle DGD \rangle$ filtrées et des valeurs de $\langle PDL \rangle$ filtrées sur le lien du réseau Verizon	69
Tableau 3.4	Coefficients de corrélation de Pearson sur plusieurs régions de l'histogramme normalisé $f_{DGD}(x)$ des valeurs de DGD par la fonction de densité de Maxwell	79
Tableau 4.1	Évaluation de l'ajustement de la fonction de densité de Rayleigh $f(x, \sigma_{opt})$ sur les valeurs de l'histogramme $f_{ARR}(x)$ des vitesses de rotation angulaire	97
Tableau 4.2	Évaluation de l'ajustement de la fonction $f_E(x, \lambda)$ sur l'histogramme des vitesses rapides de rotation angulaire comprises entre 1.1 rad/ms et 2.0 rad/ms	99

LISTE DES FIGURES

	Page
Figure 1.1	Vues en deux dimensions d'ondes lumineuses polarisées Tirée de Hui <i>et al.</i> (2006, p189).....6
Figure 1.2	États de polarisation représentés sur la sphère de Poincaré Adaptée de Cantrell (2004, p.33).....9
Figure 1.3	Représentation de la vitesse de rotation angulaire des états de polarisation sur la sphère de Poincaré11
Figure 1.4	Schéma des stress intrinsèques et des stress extrinsèques qui modifient la structure du cœur de la fibre Adaptée de Chan (2010, p.102)13
Figure 1.5	Schéma de la DGD après la propagation du signal optique dans une section de liaison optique Adaptée de Liétart (2005).....15
Figure 1.6	Schéma de la PDL lors de la transmission du signal optique dans un dispositif photonique Adaptée d'Agilent (2008, p.3)16
Figure 2.1	Schéma général de la liaison ÉTS-Ciena déployée sur le lien Montréal-Ottawa du réseau CANARIE Adaptée de Tremblay <i>et al.</i> (2012)24
Figure 2.2	Schéma de la liaison aérienne de la côte Ouest des États-Unis Adaptée de Crawford <i>et al.</i> (2013)26
Figure 2.3	Diagramme de constellation du format de modulation DP-QPSK Adaptée de Roberts <i>et al.</i> (2009, p.2)28
Figure 2.4	Schéma à haut niveau du transmetteur cohérent DP-QPSK Adaptée de Roberts <i>et al.</i> (2009, p.3)29
Figure 2.5	Schéma de l'architecture générale du récepteur cohérent Adaptée de Roberts <i>et al.</i> (2009, p.3)31
Figure 2.6	Schéma fonctionnel du module de traitement de signal électronique du transpondeur cohérent à 40 Gbit/s Adaptée de Roberts <i>et al.</i> (2009, p.5)32

Figure 2.7	Schéma à haut niveau de l'analyseur de polarisation intégré au synthétiseur de polarisation Adaptée de la fiche technique d'Agilent (2013).....	35
Figure 2.8	Diagramme de fonctions du programme d'acquisition des données avec le transpondeur cohérent du 25 au 30 octobre 2012.....	39
Figure 2.9	Diagramme de fonctions du programme d'acquisition des données avec le transpondeur cohérent du 5 au 11 novembre 2012.....	39
Figure 2.10	Diagramme de fonctions du programme d'acquisition des données enregistrées avec le polarimètre	40
Figure 3.1	Séquence de traitement des fichiers de mesure enregistrés par la carte OCLD.....	42
Figure 3.2	Structure de données indexée par « clé » des paramètres de <i>monitoring</i> lors d'une période d'acquisition réalisée avec le transpondeur cohérent.....	44
Figure 3.3	Méthode de calcul et d'analyse des valeurs de DGD et de PDL horodatées mesurées par le transpondeur cohérent.....	46
Figure 3.4	Courbes d'évolution temporelle des valeurs de DGD et des valeurs de PDL mesurées à chaque seconde par le transpondeur cohérent.....	48
Figure 3.5	Courbes d'évolution temporelle des valeurs de DGD et des valeurs de PDL mesurées à chaque milliseconde par le transpondeur cohérent	48
Figure 3.6	Méthode de calcul de la moyenne et de l'écart type des valeurs de DGD et de PDL ainsi que de leur dérivée temporelle respective pour chaque minute de mesure	51
Figure 3.7	Évolution temporelle des valeurs de $\langle DGD \rangle$, de $ d\langle DGD \rangle/dt $, de $\langle PDL \rangle$ et de $ d\langle PDL \rangle/dt $ à chaque minute sur le lien optique cohérent du réseau Canarie	53
Figure 3.8	Évolution temporelle des valeurs de $\langle DGD \rangle$, de $\langle PDL \rangle$, de $ d\langle DGD \rangle/dt $ et de $ d\langle PDL \rangle/dt $ pour chaque minute sur le lien optique cohérent du réseau Verizon.....	54
Figure 3.9	Évolution temporelle des valeurs de σ_{DGD} , de σ_{PDL} , de $ d\langle \sigma_{DGD} \rangle/dt $ et de $ d\langle \sigma_{PDL} \rangle/dt $ à chaque minute sur le lien optique cohérent du réseau CANARIE.....	55

Figure 3.10	Évolution temporelle des valeurs de σ_{DGD} , de σ_{PDL} , de $ d\langle\sigma_{DGD}\rangle/dt $ et de $ d\langle\sigma_{PDL}\rangle/dt $ à chaque minute sur le lien optique cohérent du réseau Verizon	56
Figure 3.11	Méthode de calcul de la fonction d'autocorrélation et des temps de corrélation τ_{HWHM} et $\tau_{1/e}$ sur les valeurs de DGD et sur les valeurs de PDL	58
Figure 3.12	Séquence de calcul du filtrage super gaussien de la transformée de Fourier des valeurs de $\langle DGD \rangle$ et de $\langle PDL \rangle$	60
Figure 3.13	Partie imaginaire des valeurs de $\langle DGD \rangle$ sur le lien optique cohérent du réseau Verizon	61
Figure 3.14	Réponse spectrale du filtre super gaussien d'ordre 4	63
Figure 3.15	Séquence de calcul de la fonction d'autocorrélation normalisée sur les valeurs de $\langle DGD \rangle$ filtrées et sur les valeurs de $\langle PDL \rangle$ filtrées	65
Figure 3.16	Fonctions d'autocorrélation ACF_{norm} des valeurs de $\langle DGD \rangle$ filtrées sur la liaison du réseau Canarie et sur la liaison du réseau Verizon durant 5 jours	67
Figure 3.17	Méthode de calcul et d'ajustement de l'histogramme normalisé des valeurs de DGD mesurées sur le lien du réseau Verizon	71
Figure 3.18	Ajustement de l'histogramme normalisé des valeurs de DGD par la fonction de densité de Maxwell	77
Figure 4.1	Séquence de traitement des fichiers de mesure enregistrés avec le polarimètre	82
Figure 4.2	Structure de données indexée obtenue après l'exécution du script Matlab dédié au traitement de données enregistrées au polarimètre	86
Figure 4.3	Méthode de calcul et d'ajustement de la fonction de densité des vitesses de rotation angulaire des états de polarisation	87
Figure 4.4	Ajustement de l'histogramme $f_{ARR}(x)$ des vitesses de rotation angulaire par la fonction de densité de Rayleigh	96
Figure 4.5	Ajustement de l'histogramme normalisé des vitesses de rotation angulaire par la fonction de densité de Rayleigh et par la fonction de densité exponentielle	99

Figure 4.6	Méthode d'ajustement de la fonction de densité double f_{GEI} par l'application du module <i>NonlinearModel</i> de la <i>Statistics Toolbox</i> de Matlab	103
Figure 4.7	Ajustement de l'histogramme entier des vitesses de rotation angulaire des états de polarisation par la fonction f_{GEI}	104
Figure 4.8	Méthode de calcul des paramètres caractéristiques des effets transitoires de polarisation	106
Figure 4.9	Histogramme des valeurs d'élasticité des effets transitoires de polarisation	110
Figure 4.10	Histogramme $F(ARR_{max}, \phi_{max})$ des vitesses maximales de rotation angulaire ARR_{max} et des élasticités ϕ_{max} des effets transitoires de polarisation	112
Figure 4.11	Histogramme normalisé $f_{\Delta\tau}(x)$ des valeurs de durée des effets transitoires de polarisation	113
Figure 4.12	Histogramme $F(ARR_{max}, \Delta\tau)$ des vitesses maximales de rotation angulaire ARR_{max} et des durées $\Delta\tau$ des effets transitoires de polarisation	114
Figure 4.13	Ajustement de la fonction des intervalles de temps $f_{\Delta T}(x)$ par la fonction de densité exponentielle $f_E(x, \lambda)$ à l'aide d'une échelle logarithmique	116

LISTE DES ABRÉVIATIONS, SIGLES ET ACRONYMES

ACF	AutoCorrelation Function
BER	Bit Error Rate
CAN	Convertisseur Analogique-Numérique
CD	Chromatic Dispersion
CSV	Comma-Separated Values
DAC	Digital To Analog Converter
DGD	Differential Group Delay
DOP	Degree Of Polarization
DP-QPSK	Dual Polarization Quadrature Phase Shift Keying
DSP	Digital Signal Processing
EDFA	Erbium-doped Fiber Amplifier
FEC	Forward Error Correction
FIR	Finite Impulse Response
JPDF	Joint Probability Density Function
NDSF	Non Dispersion-Shifted Fiber
NZ-DSF	Non Zéro Dispersion-Shifted Fiber
OCLD	Optical Channel Laser Detector
OPGW	Optical Ground Wire
QPSK	Quadrature Phase Shift Keying
PBC	Polarization Beam Combiner
PBS	Polarization Beam Splitter

PDG	Polarization Dependent Gain
PDL	Polarization Dependent Loss
PDF	Probability Density Function
PHP	Hypertext PreProcessor
PMD	Polarization Mode Dispersion
PBS	Principal State Of Polarization
RZ	Return To Zero
ROADM	Reconfigurable Optical Add Drop Multiplexer
SNR	Signal To Noise Ratio
SOP	State Of Polarization
WDM	Wavelength-division multiplexing

Rapport-Gratuit.com

INTRODUCTION

Le transpondeur cohérent joue un rôle prédominant dans la transmission des signaux optiques à haut débit multiplexés en polarisation et encodés par le format de modulation DP-QPSK. Malgré la compensation des effets de dispersion chromatique (CD) et de dispersion des modes de polarisation (PMD) et malgré le *tracking* des états de polarisation par les filtres numériques implémentés dans le module du traitement de signal numérique (DSP) du transpondeur cohérent, les performances du système de transmission sont toujours confrontées aux dégradations du rapport signal optique sur bruit (OSNR) causées par la combinaison des effets de PMD, de la perte dépendante de polarisation (PDL), du gain dépendant de polarisation (PDG) et de fluctuations des états de polarisation qui se produisent sur les liaisons optiques cohérentes d'après Chan (2010, p.30). Peu de travaux de *monitoring* des effets de polarisation et des paramètres de performance ont été recensés jusqu'à présent en raison de la difficulté d'accès aux terminaux situés aux extrémités des liaisons optiques commerciales et aux systèmes de transmission cohérents à fibre optique.

Objectifs de la recherche

Ce projet de recherche effectué en collaboration avec Ciena et avec Verizon a pour objectif **le *monitoring* et la caractérisation des effets de polarisation et des paramètres de performance mesurés avec un transpondeur cohérent DP-QPSK à 40 Gbit/s durant plusieurs semaines sur deux liaisons optiques cohérentes**; l'une majoritairement enterrée du réseau de CANARIE, l'autre majoritairement aérienne du réseau de Verizon. Ce projet de recherche comporte ainsi deux expériences de *monitoring* qui apportent de nouvelles connaissances sur la nature et le comportement des effets de polarisation mesurés sur deux installations optiques différentes et s'inscrivent dans la lignée des travaux réalisés par Peterson (2004), Matsuda *et al.* (2009), Woodward *et al.* (2012), Nelson (2012) et Woodward *et al.* (2014). **Un polarimètre mesure parallèlement les états de polarisation d'un signal optique à 10 Gbit/s qui sert de source polarisée pour la caractérisation des fluctuations des états de polarisation sur la liaison aérienne longue distance du réseau Verizon.** Ce

type de configuration, jamais testée et mise en place au meilleur de nos connaissances, permet de valider les mesures réalisées avec le transpondeur cohérent par celles réalisées avec le polarimètre, ce qui démontre **l'originalité de cette seconde expérience de *monitoring***.

Contributions dans le cadre du projet de recherche

Ma contribution dans le cadre de ce projet de recherche a tout d'abord porté sur le traitement des données enregistrées par le transpondeur cohérent et par le polarimètre durant plusieurs semaines, ce qui a représenté un véritable défi. Un outil de traitement des données **a été développé pour chaque instrument et a pour objectif la création d'une structure de données indexée par « clé » des paramètres de *monitoring* mesurés avec le transpondeur cohérent et avec le polarimètre**. Ces structures de données indexées par « clé » ont facilité l'accès aux milliards de données acquises, la gestion de la mémoire des systèmes d'exploitation dédiée aux mesures et l'évolution d'outils d'analyse des effets de polarisation mesurés sur les deux liaisons optiques cohérentes.

Ma contribution a ensuite porté sur **le développement d'outils d'analyse qui se présentent sous la forme de scripts Matlab pour caractériser la nature et le comportement des effets de polarisation mesurés avec le transpondeur cohérent et avec le polarimètre**. Les outils d'analyse ayant pour objectif la caractérisation des valeurs de PMD et de PDL mesurées avec le transpondeur cohérent implémentent les calculs de la fonction de dérivée temporelle, de la moyenne et de l'écart type pour chaque minute de mesure et de la fonction d'autocorrélation normalisée en vue de l'obtention du temps de corrélation de ces effets de polarisation. Un outil d'analyse statistique a été développé particulièrement sur les valeurs de PMD mesurées avec le transpondeur cohérent et intègre le calcul et l'ajustement de la fonction de densité de probabilité des valeurs de PMD par la fonction de densité de Maxwell.

Deux outils d'analyse ont ensuite été développés pour la caractérisation des fluctuations des états de polarisation mesurés avec le polarimètre. Le premier outil d'analyse intègre le

calcul et l'ajustement de la fonction de densité de probabilité des vitesses de rotation angulaire des états de polarisation par plusieurs fonctions de densité théoriques (p. ex. fonction de densité de Rayleigh, etc.) pour approfondir le modèle statistique des fluctuations des états de polarisation développé par Peterson *et al.* (2004) et par Matsuda *et al.* (2009). Le second outil d'analyse a pour objectif **la caractérisation des effets transitoires de polarisation constatés parmi les fluctuations des états de polarisation mesurés avec le polarimètre**. Cet outil implémente la détection des effets transitoires de polarisation, le calcul des leurs paramètres caractéristiques et le calcul des fonctions de densité de probabilité conjointe à une ou deux variables pour caractériser la nature et le comportement des effets transitoires de polarisation qui se produisent sur la liaison aérienne longue distance du réseau Verizon.

Organisation du mémoire

Ce mémoire est organisé en quatre chapitres. Les effets de polarisation qui se produisent sur les liaisons optiques sont présentés dans le premier chapitre de ce mémoire. Les liaisons optiques cohérentes, les instruments de mesure et les méthodes d'acquisition des données réalisées dans le cadre des expériences de *monitoring* en collaboration avec Ciena et Verizon sont ensuite décrits dans le second chapitre de ce mémoire. Une revue de littérature traitant du *monitoring* des liaisons optiques cohérentes avec un transpondeur cohérent et avec un polarimètre est préalablement présentée en début du second chapitre. Les résultats des outils d'analyse appliqués sur les valeurs de PMD et de PDL mesurées avec le transpondeur cohérent sont décrits dans le troisième chapitre de ce mémoire. De la même manière, les résultats des outils d'analyse appliqués sur les états de polarisation mesurés avec le polarimètre sont discutés dans le quatrième chapitre. Les étapes de traitement des données permettant la création d'une structure de données indexée « par clé » sont préalablement décrites en début du troisième et du quatrième chapitre de ce mémoire.

CHAPITRE 1

EFFETS DE POLARISATION DANS DES LIAISONS OPTIQUES

1.1 Polarisation d'une onde électromagnétique

La polarisation d'une onde lumineuse qui se propage dans un guide d'onde (p. ex.. fibre optique) est définie par le vecteur de son champ électromagnétique \vec{E} . Le signal optique haut débit encodé par le format de modulation DP-QPSK utilisé dans les liaisons optiques cohérentes comporte deux modes de polarisation orthogonaux. Le mode de polarisation X désigne le champ électrique transversal et le mode de polarisation Y désigne le champ magnétique transversal du signal transmis. L'équation (1.1) décrit la formule générale d'une onde lumineuse polarisée qui se propage dans une fibre optique.

$$\vec{E} = \vec{a}_x E_{x0} \cos(\omega t - kz) + \vec{a}_y E_{y0} \cos(\omega t - kz - \varphi) \quad (1.1)$$

Tirée de Hui *et al.* (2006, p.189)

Les vecteurs \vec{a}_x et \vec{a}_y sont des vecteurs unitaires et définissent la direction et le sens de polarisation de l'onde lumineuse selon Hui *et al.* (2006). L'énergie de l'onde lumineuse est égale à l'addition du champ électrique transversal \vec{E}_x et du champ magnétique transversal \vec{E}_y . Les valeurs d'amplitude du mode de polarisation X et du mode de polarisation Y sont égales respectivement à E_{x0} et à E_{y0} . La polarisation du signal dépend ainsi du temps t , de la fréquence ω de l'onde lumineuse, de l'axe de propagation de la fibre optique z et du retard de la phase entre les deux polarisations orthogonales φ du champ électromagnétique selon l'équation (1.1). La figure (1.1) présente plusieurs vues en deux dimensions d'une onde lumineuse polarisée en fonction des amplitudes des composantes orthogonales des vecteurs \vec{E}_x et \vec{E}_y et du retard de la phase φ entre le mode de polarisation X et le mode de polarisation Y.

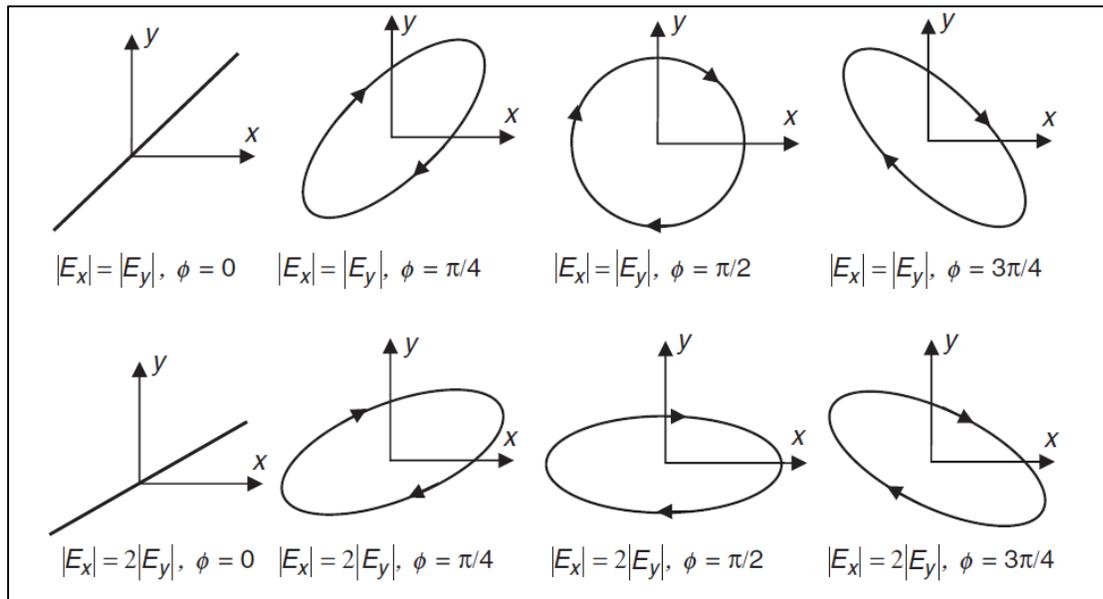


Figure 1.1 Vues en deux dimensions d'ondes lumineuses polarisées
Tirée de Hui *et al.* (2006, p189)

La figure 1.1 montre que la polarisation de la lumière peut être représentée dans un plan. Si la valeur de la phase ϕ est égale à 0, alors la polarisation est dite linéaire. Lorsque la phase est égale à $\pi/2$ et lorsque les valeurs des composantes orthogonales E_{x0} et E_{y0} sont égales, alors la polarisation de l'onde lumineuse est dite circulaire (Voir Figure 1.1). Le sens de polarisation de l'onde lumineuse est défini par le signe de ϕ . Lorsque le retard de phase ϕ est positif, alors la polarisation de l'onde lumineuse est dite elliptique avec un sens de rotation vers la droite, c'est-à-dire dans le sens des aiguilles d'une montre. Lorsque le retard de phase ϕ est négatif, alors l'ellipse qui caractérise la polarisation de l'onde lumineuse se propage dans le sens inverse des aiguilles d'une montre.

En raison des variations aléatoires de la phase ϕ durant la propagation du signal au long des sections de la liaison optique, l'intensité lumineuse d'un signal optique comprend toujours une partie polarisée et une partie non polarisée d'après Hui *et al.* (2006, p.189). De cette manière, l'addition de la puissance optique de la partie polarisée $P_{polarisé}$ et de la puissance optique de la partie non polarisée $P_{non\ polarisé}$ est égale à la puissance totale d'une onde lumineuse partiellement polarisée. L'équation (1.2) formule le calcul du degré de polarisation

(*degree of polarization*, DOP) qui désigne le rapport de la puissance de la partie polarisée sur la puissance total du signal optique émis par un laser.

$$DOP = \frac{P_{Polarisé}}{P_{polarisé} + P_{non\ polarisé}} \quad (1.2)$$

Tirée de Hui *et al.* (2006, p.190)

La valeur de DOP est égale à 0 lorsque le signal est complètement non polarisé et est égale à 1 lorsque le signal est complètement polarisé selon l'équation (1.2).

1.2 États de polarisation

Cette section de chapitre présente de façon détaillée la représentation mathématique et la représentation géométrique des états de polarisation (*State of polarization*, SOP).

1.2.1 Le vecteur Stokes

En raison des nombreux paramètres qui représentent mathématiquement les états de polarisation d'une onde lumineuse (p. ex. les amplitudes maximales E_{x0} et E_{y0} , etc.), l'interprétation et l'analyse de la nature et du comportement des états de polarisation d'un signal optique sont souvent sources de confusion d'après Hui *et al.* (2006, p.190). Le vecteur Stokes est l'outil mathématique le plus populaire qui permet de décrire efficacement la nature et le comportement des états de polarisation d'un signal optique et est déterminé par quatre paramètres indépendants désignés par les variables S_0 , S_1 , S_2 , S_3 . Chaque paramètre de Stokes est égal à la puissance optique d'un état de polarisation particulier, ce qui signifie qu'il dépend des valeurs maximales de puissance E_{x0} et E_{y0} des polarisations orthogonales et de la valeur du retard de phase φ entre les modes de polarisation orthogonaux d'après Hui *et al.* (2006). Les équations (1.3 – 1.6) formulent le calcul de chacun des paramètres de Stokes à partir de l'équation (1.1).

$$S_0 = P = |E_{x0}|^2 + |E_{y0}|^2 \quad (1.3)$$

$$S_1 = |E_{x0}|^2 - |E_{y0}|^2 \quad (1.4)$$

$$S_2 = 2|E_{x0}||E_{y0}| \cos \varphi \quad (1.5)$$

$$S_3 = 2|E_{x0}||E_{y0}| \sin \varphi \quad (1.6)$$

Tirées de Hui *et al.* (2006, p.190)

Le paramètre S_0 du vecteur Stokes est la puissance totale de l'onde lumineuse désignée par la variable P composée de la puissance de la partie polarisée $P_{polarisé}$ et de la partie non polarisée $P_{non\ polarisé}$ du signal optique. L'équation (1.7) décrit le calcul de la puissance optique de la partie polarisée d'un signal optique à partir des paramètres de Stokes d'après Hui *et al.* (2006, p.190).

$$P_{polarisé} = S_1^2 + S_2^2 + S_3^2 \quad (1.7)$$

Le vecteur Stokes est normalisé en divisant les paramètres S_1, S_2, S_3 par la puissance totale S_0 pour caractériser correctement la nature et le comportement des états de polarisation du signal optique. L'équation (1.8) formule le calcul de normalisation des vecteurs Stokes selon Hui *et al.* (2006).

$$s_1 = \frac{S_1}{S_0} \quad s_2 = \frac{S_2}{S_0} \quad s_3 = \frac{S_3}{S_0} \quad (1.8)$$

L'équation (1.9) formule le calcul du degré de polarisation à partir des valeurs des paramètres de Stokes S_1, S_2, S_3 d'après Hui *et al.* (2006).

$$DOP = \sqrt{s_1^2 + s_2^2 + s_3^2} \quad (1.9)$$

De cette manière, les états de polarisation de la partie polarisée d'un signal optique sont toujours représentés par le vecteurs Stokes normalisé (s_1, s_2, s_3) et par le degré de polarisation DOP d'après Hui *et al.* (2006, p.191).

1.2.2 Représentation du vecteur Stokes sur la sphère de Poincaré

Les valeurs des paramètres s_1, s_2 et s_3 du vecteur Stokes normalisé sont réelles et comprises entre -1 et +1. Le vecteur Stokes normalisé s'insère dans un espace à trois dimensions appelée la sphère de Poincaré dont la norme est égale au degré de polarisation du signal optique et les unités des axes (s_1, s_2, s_3) s'expriment en radian. L'extrémité du vecteur Stokes se situe toujours sur la surface de la sphère lorsque le signal est complètement polarisé et se situe toujours à l'intérieure de la sphère lorsque le signal optique est partiellement polarisé. La figure 1.2 montre la représentation géométrique d'états de polarisation particuliers sur la sphère de Poincaré et illustrés préalablement à la figure 1.1.

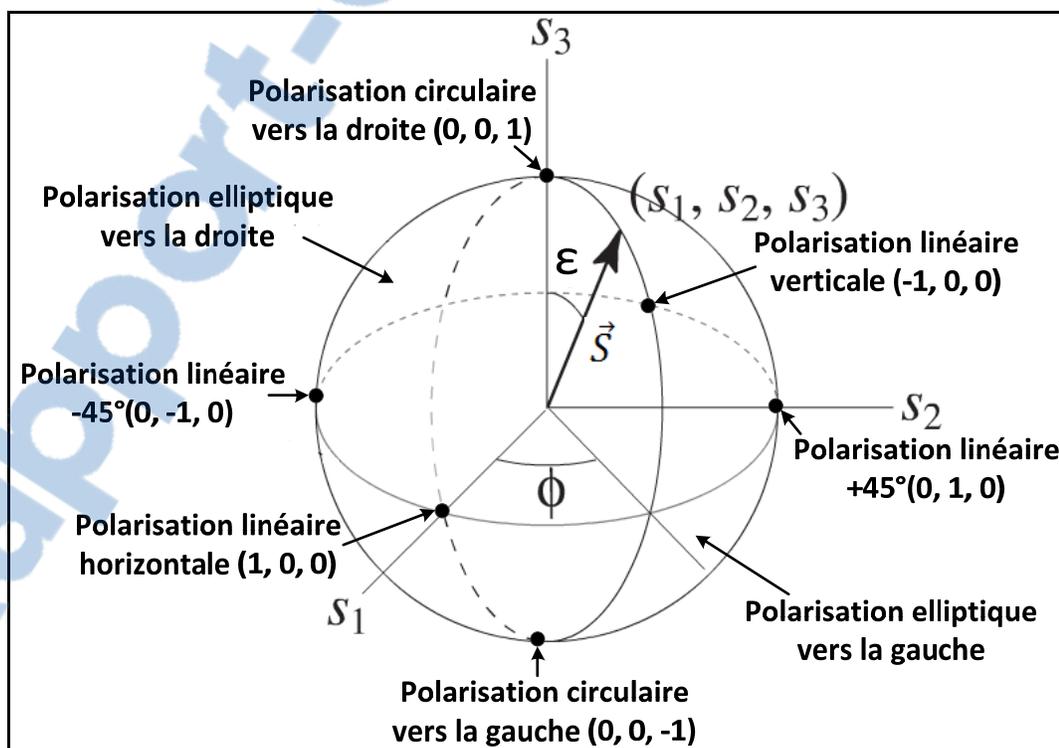


Figure 1.2 États de polarisation représentés sur la sphère de Poincaré
Adaptée de Cantrell (2004, p.33)

La polarisation est dite linéaire lorsque seul le paramètre S_1 du vecteur Stokes est différent de 0. La polarisation est orientée de 45° lorsque seule la composante S_2 n'est pas nulle. Enfin, la polarisation du signal est dite circulaire lorsque seule la composante S_3 est différente de 0. Les valeurs des paramètres Stokes informent sur le sens de la rotation (vers la gauche ou vers la droite) ou sur la nature de polarisation (horizontale ou verticale). Si les valeurs des paramètres S_1 , S_2 et S_3 sont différents de 0, alors la polarisation est de nature elliptique (*Voir* Figure 1.2). Les coordonnées polaires sphériques des états de polarisation sur la sphère de Poincaré sont constituées de la norme du vecteur Stokes, c'est-à-dire du degré de polarisation DOP, de l'angle azimutal ϕ et de l'angle polaire ϵ qui indiquent la direction de la polarisation du signal optique dans l'espace. Plus précisément, l'angle ϕ définit l'angle du vecteur Stokes dans le plan horizontal et l'angle polaire ϵ indique l'orientation elliptique de l'état de polarisation du signal optique (*Voir* Figure 1.2). Les équations (1.10 – 1.11) formulent le calcul de l'angle azimutal ϕ et le calcul de l'angle polaire ϵ . L'angle azimutal ϕ est compris entre 0 et 2π , alors que l'angle polaire ϵ s'étend entre $-\pi/2$ et $\pi/2$ selon Cantrell (2004).

$$2\Phi = \tan^{-1} \frac{S_2}{S_1} \quad (1.10)$$

$$2\epsilon = \tan^{-1} \frac{S_3}{\sqrt{S_1^2 + S_2^2}} \quad (1.11)$$

1.2.3 Fluctuation des états de polarisation

Les fluctuations des états de polarisation sont des effets de polarisation pouvant dégrader la qualité de transmission d'un signal optique multiplexé en polarisation. Les valeurs des paramètres de Stokes (S_1 , S_2 , S_3) correspondant aux états de polarisation peuvent fluctuer très rapidement lors de la propagation du signal dans les liaisons optiques, spécialement celles déployées sur des installations aériennes d'après Chan (2010, p. 36). Les limites des systèmes de transmission à fibre optique dans le *tracking* des fluctuations rapides des états de polarisation du signal peuvent générer d'importantes erreurs de transmission d'après Chan (2010, p.30).

Bao *et al.* (2004) mentionne que les fluctuations des états de polarisation se manifestent par le déplacement angulaire entre deux vecteurs Stokes normalisés $\vec{s}(t_1)$ et $\vec{s}(t_2)$ enregistrés aux temps t_1 et t_2 . Le déplacement par unité de temps définit la vitesse de rotation angulaire de deux vecteurs Stokes exprimés en rad/ms et représenté par un arc sous-tendu sur la sphère de Poincaré. La figure 1.3 décrit l'arc sous-tendu entre deux vecteurs Stokes enregistrés aux temps t_1 et t_2 désignant la vitesse de rotation angulaire $\frac{d\vec{s}}{dt}$ des états de polarisation du signal optique transmis.

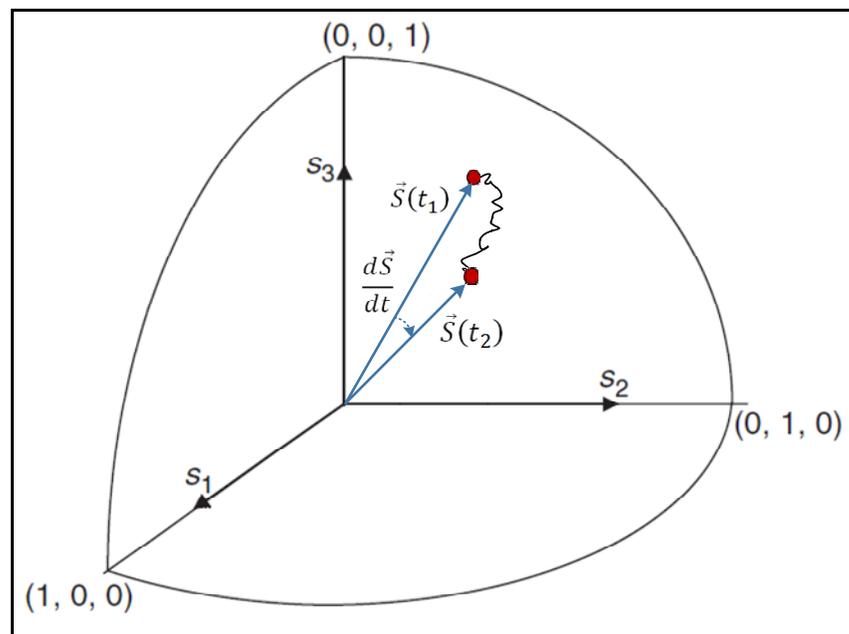


Figure 1.3 Représentation de la vitesse de rotation angulaire des états de polarisation sur la sphère de Poincaré

La vitesse de rotation angulaire $\frac{d\vec{s}}{dt}$ des états de polarisation est représentée par un arc qui relie deux points sur la trajectoire des vecteurs Stokes sur la sphère de Poincaré (*Voir* Figure 1.3). L'équation (1.12) formalise le calcul de la vitesse de rotation angulaire des états de polarisation désignée par la variable $\frac{d\vec{s}}{dt}$ adaptée de Bao *et al.* (2004).

$$\frac{d\vec{S}}{dt} = \frac{\text{acos}(\overrightarrow{s(t_1)} \cdot \overrightarrow{s(t_2)})}{t_2 - t_1} \quad (1.12)$$

L'équation (1.12) est un outil mathématique très efficace pour la caractérisation des fluctuations des états de polarisation selon Bao *et al.* (2004) car celle-ci calcule précisément à la fois les fluctuations de l'angle azimut φ et de l'angle polaire ε sur la sphère de Poincaré.

1.3 Dispersion des modes de polarisation

Le premier effet de polarisation qui dégrade la transmission du signal optique dans les liaisons optiques à haut débit est la dispersion des modes de polarisation (*Polarization Mode dispersion*, PMD) d'après Peterson *et al.* (2004). La PMD est un effet qui élargit temporellement l'impulsion du signal et est à l'origine d'interférences inter-symbole (ISI) ce qui génère des erreurs de détection et l'augmentation du taux d'erreur binaire (*Bit Error Ratio*, BER) si elle n'est pas compensée de manière optique ou électronique.

1.3.1 Causes de la PMD

La dispersion des modes de polarisation est causée par la biréfringence du cœur de la fibre, terminologie caractérisant l'asymétrie elliptique du cœur de la fibre optique et des déformations de la gaine causées par le stress au long du câble (Peterson, 2004, Chapitre 2).

La fibre qui a un cœur parfaitement circulaire détient un indice de réfraction identique sur toute la structure du cœur selon Peterson *et al.* (2004). Les modes de polarisation orthogonaux d'un signal optique se propagent à la même vitesse dans une fibre avec un cœur parfaitement circulaire alors qu'il se produit un retard de propagation entre les modes de polarisation orthogonaux du signal dans une fibre optique caractérisée par un cœur elliptique, élargissant ainsi l'impulsion du signal optique tout au long des sections d'une liaison optique.

La « non circularité » du cœur de la fibre optique, c'est-à-dire la biréfringence géométrique est causée par des erreurs lors du processus de fabrication de la fibre optique d'après Hui *et*

al. (2006, p.409). De nombreuses perturbations peuvent se produire lors de la transmission du signal optique et générer des effets de stress non symétriques sur toute la structure du cœur de la fibre optique. La figure 1.4 décrit les différents stress mécaniques qui modifient la structure et la biréfringence du cœur de la fibre optique.

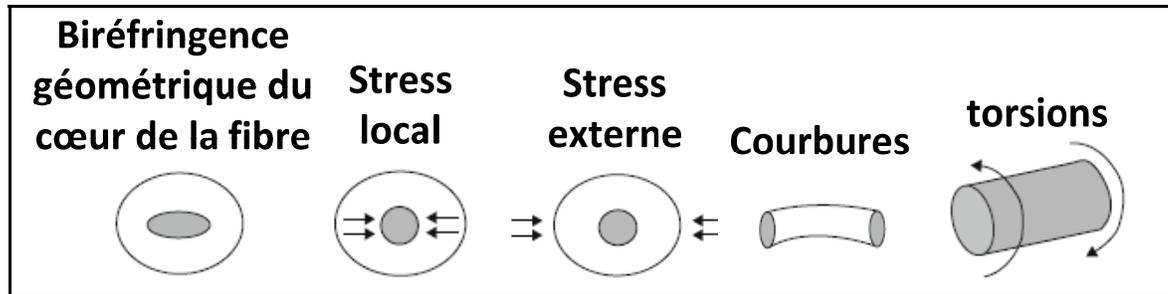


Figure 1.4 Schéma des stress intrinsèques et des stress extrinsèques qui modifient la structure du cœur de la fibre
Adaptée de Chan (2010, p.102)

Deux types de stress mécanique se distinguent selon Hui *et al.* (2006, p.409): les perturbations intrinsèques et les perturbations extrinsèques.

Les perturbations intrinsèques indiquent les manipulations réalisées sur les fibres optiques et sur le câble lors du déploiement d'une liaison optique. Les forces et les pressions exercées sur la fibre optique peuvent être à l'origine de courbures et de torsions qui désignent une source de stress local (*Voir* Figure 1.4). Les contraintes de déploiement de la fibre optique et l'inhomogénéité de sa structure causent de la biréfringence localisée tout au long des sections de la liaison optique quelle que soit son type d'installation.

Les perturbations extrinsèques désignent des sources de stress mécaniques externes qui se produisent aléatoirement au long des sections d'une liaison optique. Les variations de températures durant le jour et la nuit et à travers les saisons (p. ex. le réchauffement des températures atmosphériques en été, etc.) sont une source de stress mécanique externe qui peut modifier la structure du cœur de la fibre optique. Bien que beaucoup de liaisons optiques soient enfouies, elles sont toujours exposées aux variations de température et au stress mécanique externe correspondant d'après Chan (2010, p.101). Les liaisons optiques

aériennes sont particulièrement exposées aux vitesses de vents qui provoquent des effets de balancements sur le câble de fibres optiques. Enfin, les vibrations des passages de train sont une source supplémentaire de stress externe sur les liaisons déployées à proximité des chemins de fer en raison de la facilité de l'installation et du déploiement du câble d'après Chan (2010, p.102).

1.3.2 Paramètres de la PMD

Les modes de polarisation orthogonaux voyagent selon deux constantes de propagation distinctes, elles-mêmes dépendantes de l'écart d'indice effectif de réfraction désigné par la variable Δn_{eff} entre les deux axes de polarisation orthogonaux du signal optique selon Hui *et al.* (2006). De cette manière, l'écart d'indice effectif de réfraction Δn_{eff} est la mesure de la biréfringence du cœur de la fibre optique et est comprise entre 10^{-7} et 10^{-5} pour les fibres monomodes actuelles selon Hui *et al.* (2006). La dispersion des modes de polarisation est définie par la mesure du retard entre le mode de polarisation le plus rapide et le mode de polarisation le plus lent du signal optique. Ces modes de polarisation sont appelés les principaux états de polarisation (*Principal state of polarization*, PSP) d'un signal optique. L'équation (1.13) formule le calcul du retard de propagation désigné par la variable $\Delta\tau$ en fonction de l'écart d'indice effectif de réfraction Δn_{eff} , de la longueur de la liaison L et de la vitesse de la lumière c selon Hui *et al.* (2006, p.410).

$$\Delta\tau = \frac{L\Delta n_{eff}}{c} \quad (1.13)$$

Le retard de propagation des modes de polarisation orthogonaux X et Y (*Differential Group Delay*, DGD) est exprimée par la variable $\Delta\tau$ en ps et fluctuent dans le temps et tout au long de la liaison à cause des modifications aléatoires de la biréfringence du cœur de la fibre (*Voir* section 1.3.1). La figure 1.5 schématise la DGD après la propagation du signal dans une liaison optique.

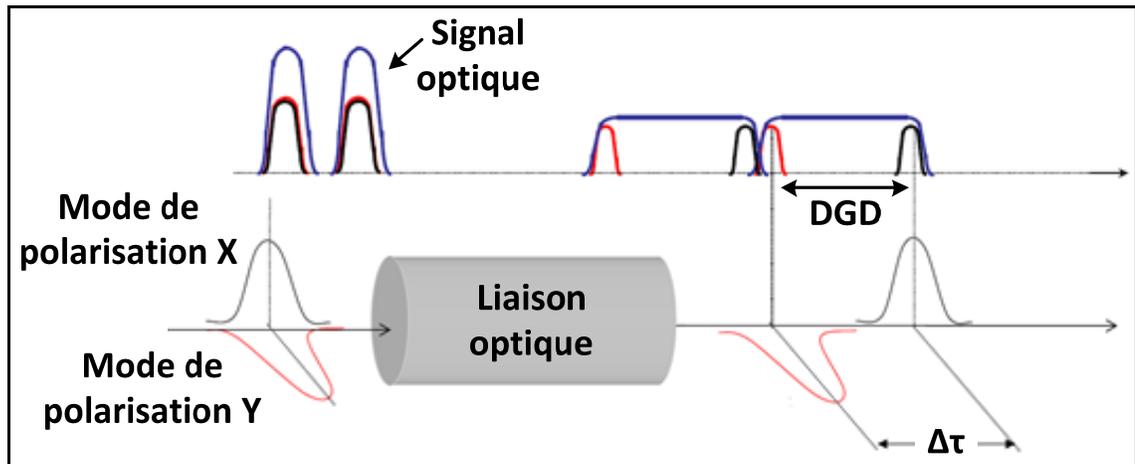


Figure 1.5 Schéma de la DGD après la propagation du signal optique dans une section de liaison optique
Adaptée de Liétart (2005)

La DGD est un effet de polarisation qui varie en fonction du temps et en fonction de la fréquence du signal optique d'après Peterson (2004, chapitre 2). La PMD est égale à la valeur moyenne des valeurs de DGD instantanée dans le temps et sur une plage de longueurs d'onde particulière d'après Hui *et al.* (2006) et s'exprime aussi en ps par la variable $\langle \Delta\tau \rangle$. Cette variable définit une valeur caractéristique de la dispersion des modes polarisation d'une liaison optique. La valeur de la PMD est proportionnelle à la racine carrée de la longueur de la liaison. L'équation (1.14) formule le calcul de la valeur de la PMD $\langle \Delta\tau \rangle$ caractéristique d'une liaison optique tirée de Hui *et al.* (2006).

$$\langle \Delta\tau \rangle = \langle \Delta\tau_c \rangle \times \sqrt{L} \quad (1.14)$$

Le coefficient PMD de la fibre ou de la liaison $\langle \Delta\tau_c \rangle$ est exprimé généralement en ps/ $\sqrt{\text{km}}$. Celle-ci permet d'estimer avec précision la valeur de la PMD $\langle \Delta\tau \rangle$ pour les liaisons optiques métropolitaines et pour les liaisons optiques de longue portée.

1.4 Perte dépendante de polarisation

La perte dépendante de polarisation (*Polarization Dependant Loss*, PDL) définit le rapport d'extinction de polarisation du signal c'est-à-dire la variation maximale de la perte d'insertion en fonction des états de polarisation du signal optique à l'entrée du dispositif photonique d'après Hui et al. (2006, p.438). De cette manière, les valeurs de PDL sont calculées par le rapport de la valeur maximale sur la valeur minimale de la puissance du signal optique transmise à la sortie du dispositif photonique, lui-même désignant les composants optiques passifs disposés sur la liaison optique (p. ex. les commutateurs, isolateurs, coupleurs, filtres, circulateurs, etc.). La PDL est un effet de polarisation majeur qui dégrade la qualité des signaux optiques, particulièrement ceux multiplexés en polarisation, en causant des fluctuations de la puissance optique en fonction de la polarisation du signal, ce qui entraîne des variations du rapport du signal optique sur bruit (OSNR) au site de réception comme le mentionne Chan (2010, p.105). Même si la PDL de chaque composant optique peut être définie avec une valeur déterministe, la valeur globale de la PDL du lien varie de façon aléatoire au cours du temps et en fonction de la longueur d'onde en raison des fluctuations des états de polarisation du signal optique à l'entrée de chaque composant optique disposé tout au long des sections de la liaison optique. La figure 1.6 schématise la perte dépendante de polarisation qui se produit lors de la transmission d'un signal optique polarisé dans un dispositif photonique.

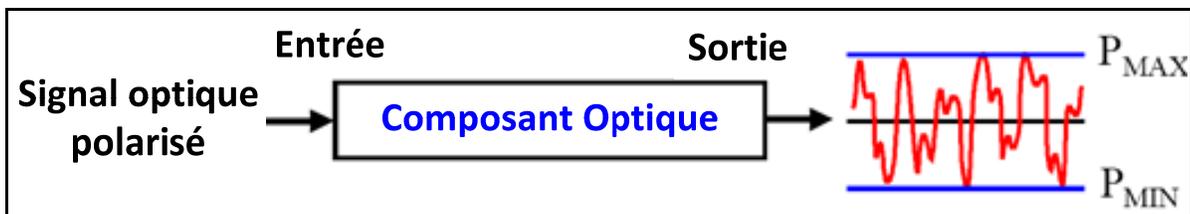


Figure 1.6 Schéma de la PDL lors de la transmission du signal optique dans un dispositif photonique
Adaptée d'Agilent (2008, p.3)

L'équation (1.15) formule le calcul de la définition générale de la PDL désignée par la variable Γ d'après Hui *et al.* (2006, p.439).

$$PDL_{dB} = \Gamma = 10 \log \left(\frac{P_{max}/P_{in} - P_{min}/P_{in}}{P_{max}/P_{in} + P_{min}/P_{in}} \right) = 10 \log \left(\frac{P_{max} - P_{min}}{P_{max} + P_{min}} \right) \quad (1.15)$$

P_{max} et P_{min} définissent la valeur minimale et la valeur maximale de puissance du signal optique transmise par le dispositif photonique respectivement. P_{in} est la valeur de puissance du signal optique à l'entrée du dispositif photonique. La valeur de la PDL Γ est exprimée en dB.

La valeur de la perte dépendante de polarisation (PDL) qui se produit dans une fibre optique est habituellement très faible, alors que la valeur de la PDL caractéristique de la liaison est affectée par tous les effets de PDL qui se produisent à la sortie de chaque composant optique disposé sur la liaison optique d'après Hui *et al.* (2010, p.445). La PDL caractéristique de la liaison ne peut pas être obtenue par une simple addition de la PDL de chaque composant optique en raison des fluctuations aléatoires des états de polarisation qui se produisent durant la transmission du signal optique.

Un effet de polarisation supplémentaire qui dégrade la qualité de transmission du signal optique est le gain dépendant de polarisation (*Polarisation Dépendant Gain*, PDG). Celui-ci se produit dans les amplificateurs optiques en raison de la variation du gain en fonction du mode de polarisation du signal optique. Bien que la valeur de PDG d'un seul amplificateur est plutôt faible et négligeable, les effets de PDG induits par une cascade d'amplificateurs optiques disposés sur une liaison optique cohérente peuvent causer des fluctuations de l'ordre de plusieurs dB sur les valeurs d'OSNR d'après Chan (2010, p.106). Enfin, l'interaction des effets de polarisation de PDL, de PDG et de PMD peuvent être à l'origine de fortes dégradations de la transmission du signal supérieures à celles induites en additionnant indépendamment chacun de ces effets de polarisation au site de réception selon Chan (2010, p.106).

CHAPITRE 2

MONITORING DE DEUX LIAISONS OPTIQUES COHÉRENTES WDM

2.1 Introduction

Ce chapitre décrit les expériences de *monitoring* réalisées sur deux liaisons optiques cohérentes longue distance avec pour objectif d'apporter une meilleure compréhension de la nature et du comportement des effets de polarisation mesurés parallèlement avec un transpondeur cohérent et avec un polarimètre. Les schémas des liaisons optiques cohérentes étudiées, les instruments de mesure déployés aux extrémités des liaisons et les méthodes d'acquisition des données développées lors des expériences de *monitoring* sont décrits tout au long de ce chapitre. Une revue de littérature traitant du *monitoring* des effets de polarisation avec un transpondeur cohérent puis avec un polarimètre sur différentes installations optiques est préalablement introduite en début de ce chapitre.

2.2 Revue des expériences de *monitoring* et de caractérisation des effets de polarisation sur différentes installations optiques

2.2.1 *Monitoring* et caractérisation des effets de polarisation avec un transpondeur cohérent sur des liaisons optiques cohérentes enterrées

Un transpondeur cohérent peut être déployé en tant qu'instrument de mesure de dispersion chromatique (CD), de dispersion des modes de polarisation (PMD), de pertes dépendantes en polarisation (PDL), des états de polarisation (SOP), du taux d'erreur binaire avant l'application du code de correction d'erreur (pre-FEC BER), du rapport signal sur bruit électrique (ESNR) et d'en déduire la valeur du facteur de qualité Q du signal transmis selon Chan (2010, p.366). Ces paramètres de *monitoring* sont calculés après analyse des coefficients des filtres numériques à réponse impulsionnelle finie (FIR) implémentés dans le circuit de traitement du signal numérique (DSP) d'après Chan (2010, p.366).

Woodward *et al.* (2012) ont caractérisé les fluctuations des valeurs de DGD, de PDL, du paramètre S_1 du vecteur Stokes (S_1, S_2, S_3) et du taux d'erreur binaire avant le code de correction d'erreur (pre-FEC BER) mesurées avec un transpondeur cohérent sur trois liaisons optiques enterrées déterminées par des longueurs égales à 30 km, à 75 km et à 273 km respectivement. Deux méthodes d'acquisition des données sont développées par Woodward *et al.* (2012). Les valeurs de DGD, de PDL, de vecteurs Stokes et de pre-FEC BER ont été mesurées toutes les 10 ms durant 93 s et toutes les secondes durant 2.6 h. Les valeurs moyennes de chaque paramètre ont été calculées à chaque seconde et à chaque minute de mesures pour la réduction du bruit généré par le système de transmission optique. Woodward *et al.* (2012) ont par la suite calculé les valeurs de dérivée temporelle de DGD, de PDL et de φ qui définit l'angle relatif au paramètre S_1 sur la sphère de Poincaré. Ces travaux ont permis de constater différentes activités en polarisation en fonction du lien optique cohérent et de l'emplacement du terminal (côté Est ou côté Ouest). Par la suite, Woodward *et al.* (2014) ont approfondi la caractérisation des valeurs de DGD et de S_1 mesurées durant chaque seconde de mesure avec un transpondeur cohérent sur les mêmes installations optiques que celles décrites précédemment. Woodward *et al.* (2014) ont mesuré et ont calculé la moyenne des valeurs de DGD et de S_1 sur une minute durant 18 mois totalisant plus de 10 000 heures de mesures ce qui définit, au meilleur de nos connaissances, l'étude des effets de polarisation la plus longue et la plus complète réalisée à ce jour avec un transpondeur cohérent. Woodward *et al.* (2014) ont ensuite calculé la fonction d'autocorrélation et la fonction de densité de probabilité sur les valeurs de DGD et de S_1 en vue d'étudier avec approfondissement les fluctuations de ces différents paramètres, ce qui constitue des méthodes d'analyse avancées pour la caractérisation des effets de polarisations sur des liaisons optiques enterrées durant de longues périodes d'acquisition. Les expériences de Woodward *et al.* (2012) et de Woodward *et al.* (2014) ont permis de valider que la fonction de densité de Maxwell ajuste avec précision l'histogramme des valeurs de $\langle DGD \rangle$, désignant la moyenne des valeurs de DGD mesurées sur une minute avec un transpondeur cohérent. De plus, les résultats du calcul de la fonction d'autocorrélation ACF ont permis de constater que le temps de corrélation τ_{HWHM} spécifique aux valeurs de DGD et de S_1 varie de 1.3 à 4.1 jours et de 0.18 à 1.7 jours respectivement, ce qui signifie que les valeurs de DGD et de S_1 sont caractérisées par une

grande stabilité en raison des faibles variations de température et de l'absence de module de compensation de dispersion sur ces installations enterrées d'après Woodward *et al.* (2014).

2.2.2 **Monitoring et caractérisation des effets de polarisation avec un polarimètre sur différentes installations optiques**

Un polarimètre, appelé aussi analyseur de polarisation (*Polarization Analyzer*), est un instrument qui permet de mesurer avec précision les états de polarisation du signal optique à la réception d'après Hui *et al.* (2006, p.193). Cet instrument a été utilisé par Peterson *et al.* (2004) et par Matsuda *et al.* (2009) pour caractériser avec précision la fonction de densité de probabilité des fluctuations des états de polarisation mesurées sur une liaison optique enterrée ou aérienne. Peterson *et al.* (2004) ont mesuré les vecteurs Stokes à une fréquence d'échantillonnage égale à 4 kHz pour caractériser les fluctuations des états de polarisation d'une source lumineuse polarisée sur deux liaisons optiques; l'une composée de 110 km de fibres enterrées, l'autre constituée de 220 km de fibres aériennes. De plus, Peterson *et al.* (2004) ont déployé un brouilleur des modes de polarisation, appelé plus communément *Polarization Scrambler*, sur un banc de test optique pour reproduire au laboratoire la fonction de densité de probabilité des fluctuations des états de polarisation observées sur les liaisons optiques enterrées ou aériennes. Des tests effectués à différentes vitesses de brouillage des modes de polarisation au laboratoire ont permis d'approcher la fonction de densité de probabilité des fluctuations des états de polarisation observées sur la liaison optique enterrée et sur la liaison optique aérienne. Matsuda *et al.* (2009) ont mesuré les vecteurs Stokes d'un signal optique RZ-DQPSK à 43 Gbit/s avec une fréquence d'échantillonnage égale à 0.2 kHz sur une liaison aérienne de 133.6 km comportant 4 sections de fibres de 33.4 km. Peterson *et al.* (2004) et Matsuda *et al.* (2009) ont démontré que l'histogramme des fluctuations des états de polarisation mesurés avec un polarimètre peut être ajusté par la fonction de densité de Rayleigh que ce soit sur une liaison optique enterrée ou sur une liaison optique aérienne.

Waddy *et al.* (2001) et Boa *et al.* (2004) ont mesuré les vecteurs Stokes d'une source polarisée avec un polarimètre sur une liaison aérienne de 34 km. Ces travaux se démarquent des précédents par le calcul de la fonction d'autocorrélation normalisée appliquée sur les

vecteurs Stokes (S_1 , S_2 , S_3) pour chaque période d'acquisition en vue d'étudier le temps de corrélation, c'est-à-dire la durée de la relation temporelle qui existe entre une mesure à l'instant t et une mesure à l'instant $t+\tau$. La fréquence d'échantillonnage des mesures de vecteurs Stokes est égale à 100 Hz dans les travaux de Waddy *et al.* (2001) et de Bao *et al.* (2004). Les résultats de ces travaux ont permis de montrer différentes activités en polarisation corrélées aux variations des températures atmosphériques et aux vitesses des vents mesurées par une station météorologique proche du site d'installation.

Waddy *et al.* (2005) ont mesuré les valeurs de DGD et les valeurs de PDL en fonction du temps à une fréquence d'échantillonnage égale à 5 kHz et en fonction de la longueur d'onde du signal transmis avec une résolution de 12.5 GHz (soit 0.1 nm) à l'aide d'un analyseur de PMD/PDL sur une liaison aérienne de 80 km. Waddy *et al.* (2005) ont calculé la fonction d'autocorrélation normalisée sur les valeurs de DGD et sur les valeurs de PDL en fonction du temps, ce qui a permis de constater que les temps de corrélation sur une période totalisant 50 min de mesures des valeurs de DGD et des valeurs de PDL sont égales à 78.75 s et à 26.25 s respectivement. Les résultats de ces travaux confirment qu'il se produit des fluctuations rapides et importantes de DGD et de PDL sur des liaisons optiques aériennes. De surcroît, Kawasaki *et al.* (2007) ont aussi appliqué la fonction d'autocorrélation normalisée ainsi que la fonction de densité de probabilité sur les fluctuations des moyennes des valeurs de DGD mesurées toutes les 15 minutes durant 5 mois avec un analyseur de PMD sur deux liaisons optiques enterrées composées de 24.5 km et de 79.7 km de fibres G.653 respectivement. Kawasaki *et al.* (2007) ont confirmé que le temps de corrélation des valeurs de DGD s'étendait sur plusieurs heures, ce qui signifie que les fluctuations des valeurs de DGD qui se produisent sur une liaison optique enterrée sont bien moins rapides et moins élevées que celles mesurées sur une liaison optique aérienne selon Waddy *et al.* (2005).

Dernièrement, Nelson (2012) a mesuré avec un polarimètre les effets transitoires de polarisation caractérisés par un seuil de fluctuation égale à $10^\circ/100$ ms (soit 0.175 rad/100ms) sur une liaison optique WDM enterrée de 1800 km. Le polarimètre a été utilisé en mode rafale à une fréquence d'échantillonnage égale à 5 kHz puis à 50 kHz durant une

période de 100 ms pour la capture et l'enregistrement des vecteurs Stokes correspondant aux effets transitoires de polarisation dépassant le seuil précédemment défini. 515 effets transitoires de polarisation ont été capturés durant 79 jours d'observation. Nelson (2012) a ensuite calculé pour chaque effet transitoire de polarisation l'élasticité et la valeur maximale des vitesses de rotation angulaire mesurées par le polarimètre durant chaque effet transitoire de polarisation détecté. Cette toute nouvelle méthode de caractérisation a permis de constater que la majorité des effets transitoires de polarisation étaient plutôt élastiques ($< 11^\circ$) mais qu'il pouvait subsister des effets transitoires de polarisation dont l'élasticité, la vitesse maximale et la durée sont très supérieures aux valeurs normales. Quatre effets transitoires de polarisation très élastiques n'ont d'ailleurs pu être clairement caractérisés en raison d'une vitesse atteignant 90° en moins de $20 \mu\text{s}$ et d'une durée d'acquisition insuffisante au polarimètre d'après Nelson (2012).

2.3 Description des expériences de *monitoring* réalisées sur deux liaisons optiques cohérentes

Cette section décrit deux expériences de *monitoring* réalisées dans le cadre de ce projet de maîtrise et dans le cadre de la collaboration entre le laboratoire de recherche de Ciena à Ottawa et le *Laboratoire de technologies de réseaux*. Un transpondeur cohérent DP-QPSK à 40 Gbit/s du système de transmission optique Ciena OM6500™ a été utilisé en tant qu'instrument de mesure des paramètres de *monitoring* et des performances pour la caractérisation du signal transmis sur deux liaisons optiques cohérentes. La première expérience de *monitoring*, en collaboration avec Ciena et CANARIE, constitue une première méthode d'analyse et de caractérisation des effets de polarisation mesurés avec le transpondeur cohérent durant quatre semaines sur une liaison optique cohérente majoritairement enterrée. La seconde expérience de *monitoring* a été réalisée en collaboration avec Ciena et Verizon sur une liaison optique commerciale aérienne longue distance. Des mesures en parallèle sont réalisées sur cette liaison avec un polarimètre toutes les $50 \mu\text{s}$ (soit à une fréquence d'échantillonnage égale à 20 kHz) et avec le même transpondeur cohérent que lors de la première expérience de *monitoring* à ceci près que sa fréquence d'échantillonnage est à présent égale à 1 kHz. Ce type de configuration n'a jamais été testé et

mise en place dans les travaux précédents de *monitoring* démontrant ainsi l'originalité de cette seconde expérience de *monitoring* (Voir section 2.2). Les méthodes d'analyse et de caractérisation précédemment décrites dans la section 2.2 sont appliquées lors de ces expériences de *monitoring* pour tester, valider ou approfondir les résultats des travaux précédents de *monitoring* réalisés avec un transpondeur cohérent ou avec un polarimètre.

2.3.1 Expérience de *monitoring* d'une liaison optique cohérente du réseau Canarie

La première liaison optique cohérente étudiée dans le cadre du projet de *monitoring* est une liaison optique de 230 km majoritairement enterrée qui relie le *Laboratoire de technologies de réseaux* et le laboratoire de recherche de Ciena à Ottawa par le biais du réseau WDM de CANARIE (« *Canada's advanced research and innovation network* ») à 10 Gbit/s. La figure 2.1 présente le schéma général de la liaison ÉTS-Ciena du réseau CANARIE.

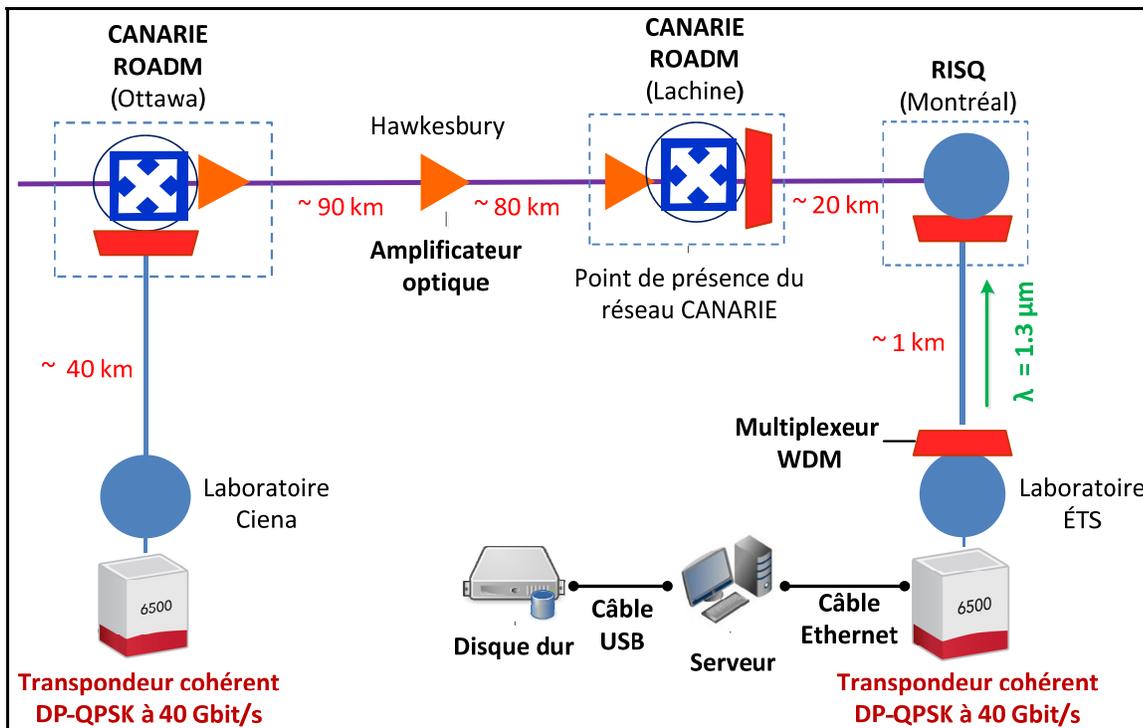


Figure 2.1 Schéma général de la liaison ÉTS-Ciena déployée sur le lien Montréal-Ottawa du réseau CANARIE
Adaptée de Tremblay *et al.* (2012)

Cette liaison est composée au total de quatre principaux segments (spans) d'environ 40, 90, 80 et de 20 km respectivement. Celle-ci inclut deux sites ROADM (*Reconfigurable Add Drop Multiplexer*) et des amplificateurs optique EDFA (*Erbium Doped Fiber Amplifiers*) (Voir Figure 2.1). Deux types de fibre caractérisent le lien ÉTS-Ciena. La fibre déployée sur la liaison présentée par la figure 2.1 est principalement de type NZ-DSF (*Non Zero – Dispersion-Shifted Fiber*) sous l'appellation commerciale LEAF[®] du manufacturier Corning. La seconde fibre déployée est de type NDSF (*Non Dispersion-Shifted Fiber*). Le lien ÉTS-Ciena est aussi constitué de deux types d'installation. Environ 80 % des câbles optiques du lien ÉTS-Ciena sont enfouis. Le segment reliant le laboratoire de recherche de Ciena et le site ROADM à Ottawa ainsi que le lien Montréal-Lachine sont caractérisés par une installation aérienne (Voir Figure 2.1). Des multiplexeurs WDM 1310/1550 sont intégrés au site de l'ÉTS et du RISQ en vue de la transmission en parallèle d'un signal optique à 1310 nm utilisé dans une autre expérience indépendante. Les cartes OCLD (*Optical Channel Laser Detector*), c'est-à-dire les transpondeurs cohérents DP-QPSK à 40 Gbit/s, déployées sur le site de Ciena et sur le site de l'ÉTS font parties du système de transmission à fibre optique OM6500[™] de Ciena (Voir Figure 2.1). Les transpondeurs cohérents ont été déployés et activés en collaboration avec les experts de CANARIE et de Ciena. La période de mesure a débuté le 14 octobre 2011 et a pris fin le 16 novembre 2011.

2.3.2 Expérience de *monitoring* d'une liaison optique cohérente du réseau Verizon

Cette seconde expérience de *monitoring* est réalisée sur une liaison optique cohérente du réseau Verizon aux États-Unis de près de 900 km. Cette liaison a été déployée sur des câbles de garde à fibres optiques plus connu sous la standardisation OPGW (« *Optical ground wire* »). Ce type d'installation est plus sensible aux fluctuations des effets de polarisation que celles subsistant sur d'autres types d'installations (p. ex. liaison optique enterrée) tel que démontré dans les travaux de *monitoring* effectués par Bao *et al.* (2004), Waddy *et al.* (2005) et Matsuda *et al.* (2009). En effet, ce type de liaison est soumis à de nombreuses perturbations issues de l'environnement externe telles que les variations de la température, de la pression atmosphérique et des vitesses de vent (effet de balancement des câbles). La figure

2.2 décrit le schéma de la liaison majoritairement aérienne déployée sur la côte Ouest des États-Unis du réseau Verizon adapté de Crawford *et al.* (2013).

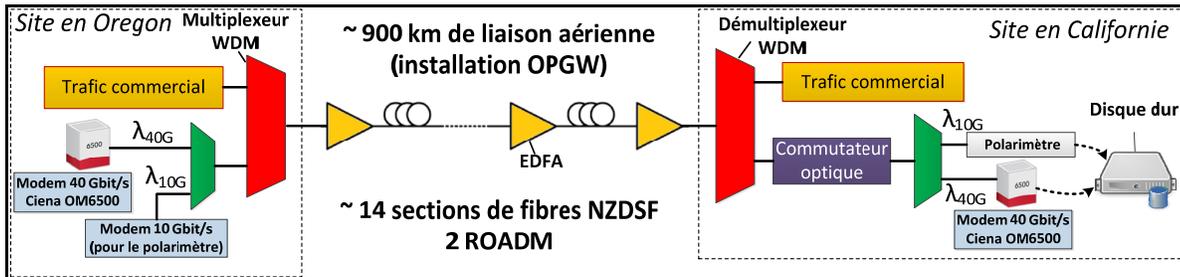


Figure 2.2 Schéma de la liaison aérienne de la côte Ouest des États-Unis
Adaptée de Crawford *et al.* (2013)

La liaison aérienne du réseau Verizon est constituée de deux sites ROADM et de 14 sections de fibres NZDSF selon Crawford *et al.* (2013). Cette liaison dispose de sites d'amplifications optiques EDFA pour la gestion de la puissance des signaux optiques WDM. Deux systèmes de transmission sont déployés sur les terminaux aux extrémités de la liaison localisées en Oregon et en Californie. Le transpondeur cohérent OM6500™ de Ciena à 40 Gbit/s ainsi qu'un modem à 10 Gbit/s sont les systèmes de transmission à fibre optique déployés au terminal en Oregon. Crawford *et al.* (2013) énoncent que le débit du signal émis par le transpondeur cohérent, appelé λ_{40G} , est égale à 40 Gbit/s alors que le débit du signal émis par le modem cohérent, désigné par λ_{10G} , est égale à 10 Gbit/s (*Voir* Figure 2.2). L'écart spectral entre ces deux longueurs d'onde λ_{40G} et λ_{10G} est égale à 100 GHz (soit 0.8 nm) d'après Crawford *et al.* (2013). Les longueurs d'onde λ_{40G} et λ_{10G} sont multiplexées une première fois dans une bande optique WDM avant d'être multiplexées une seconde fois pour les insérer aux longueurs d'onde dédiées au trafic commercial de données. Celles-ci sont espacées spectralement de plus de 500 GHz avec les longueurs d'onde dédiées au trafic commercial de données à très haut débit transmis sur cette liaison (*Voir* Figure 2.2). Les longueurs d'onde λ_{40G} et λ_{10G} sont transmises jusqu'au terminal localisé en Californie. Celles-ci sont ensuite extraites par l'application de deux démultiplexeurs WDM identiques à ceux déployés au terminal en Oregon. Un commutateur optique, appelé plus communément *Optical Switch*, est inséré entre le premier et le second démultiplexeur WDM pour la commande des instruments

de mesure. Le commutateur optique exécute un programme d'acquisition conçu par Crawford *et al.* (2013). Celui-ci a pour objectif, dans un premier temps, l'activation et l'acquisition des mesures des paramètres de *monitoring* de chaque instrument et, dans un second temps, le transfert des données acquis vers le disque dur (*Voir* Figure 2.2). Un second transpondeur cohérent DP-QPSK à 40 Gbit/s a été déployé au terminal en Californie en tant qu'instrument de mesures des paramètres de *monitoring* et de performances du signal caractérisé par la longueur d'onde λ_{40G} . La fonction « polarimètre » du synthétiseur de polarisation Agilent N7786B, déployé en parallèle au terminal en Californie, est activé pour la mesure avec précision des vecteurs Stokes (S_1 , S_2 , S_3) du signal caractérisée par la longueur d'onde λ_{10G} . De cette manière, le signal émis par le modem à 10 Gbit/s sert de source polarisée pour la mesure des paramètres de Stokes par le polarimètre lors de cette expérience de *monitoring* (*Voir* Figure 2.2). La première et la seconde série de mesure ont été générées du 25 octobre au 30 octobre 2012 et du 5 novembre au 11 novembre 2012 respectivement par le polarimètre et par le transpondeur cohérent. Une troisième série de mesure a été générée du 4 au 19 septembre 2013 par le polarimètre uniquement.

2.4 Transpondeur cohérent DP-QPSK à 40 Gbit/s

Le transmetteur et le récepteur du transpondeur cohérent DP-QPSK à 40 Gbit/s sont décrits dans cette section de chapitre avec pour objectif la description de la modulation et des caractéristiques du signal transmis, de la description synthétique des mesures de paramètres de *monitoring* par le module de traitement de signal numérique (module DSP) sur la liaison du réseau CANARIE et sur la liaison du réseau Verizon. Les informations techniques décrivant le système de transmission Ciena DP-QPSK à 40 Gbit/s tout au long de cette section sont tirées de Roberts *et al.* (2009).

2.4.1 Modulation DP-QPSK à 40 Gbit/s

La modulation des systèmes de transmission cohérents déployés aux extrémités des liaisons est sous la forme DP-QPSK (*Dual Polarization Quadrature Phase Shift Keying*). La figure

2.3 décrit le diagramme de constellation DP-QPSK du signal émis par le transpondeur cohérent à 40 Gbit/s.

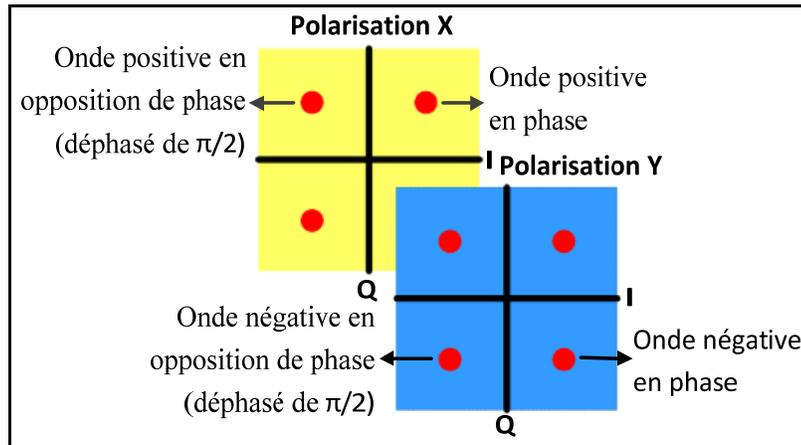


Figure 2.3 Diagramme de constellation du format de modulation DP-QPSK
Adaptée de Roberts *et al.* (2009, p.2)

Cette technique de modulation implémente un codage binaire sur quatre valeurs de phase pour chaque mode de polarisation X et Y du signal optique. Chaque valeur de phase correspond à un état. Un état désigne un symbole codé sous deux bits. Ces symboles peuvent être représentés dans un plan « IQ ». L'axe I du plan « IQ » définit la partie de l'onde en phase ou la partie de l'onde en opposition de phase. L'axe Q définit l'onde en quadrature, c'est-à-dire déphasée de 90° (Voir Figure 2.3). La vitesse de transmission des données de chaque symbole est égale à 11.5 Gbaud. Par conséquent, le débit binaire pour chaque mode de polarisation s'élève à 23 Gbit/s. Avec 2 modes de polarisation, le débit du signal optique DP-QPSK double et s'élève à 46 Gbit/s. La largeur spectrale du laser accordable est égale 0.1 nm, ce qui signifie que la bande passante optique du signal transmis s'élève à 12.5 GHz selon Roberts *et al.* (2009). Chaque signal transmis est centré dans un canal de 50 GHz situé dans la bande C de la grille de l'*International Telecommunication Union* (ITU).

2.4.2 Transmetteur cohérent à 40 Gbit/s

La figure 2.4 présente le schéma à haut niveau du transmetteur cohérent DP-QPSK qui fait partie du système de transmission OM6500™ de Ciena déployé dans les terminaux aux extrémités des liaisons du réseau CANARIE et du réseau Verizon.

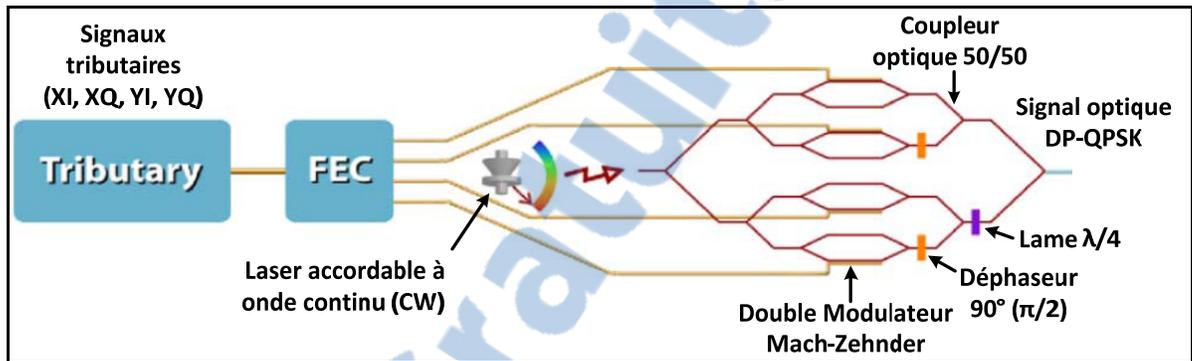


Figure 2.4 Schéma à haut niveau du transmetteur cohérent DP-QPSK
Adaptée de Roberts *et al.* (2009, p.3)

Quatre signaux tributaires sont générés par le processeur émetteur pour la transmission du signal à 40 Gbit/s. Ces signaux correspondent à chacune des composantes I et Q de la modulation QSPK des deux modes de polarisation orthogonaux du signal optique DP-QPSK. De cette manière, les signaux générés par le processeur émetteur sont désignés par XI, XQ, YI, YQ respectivement (Voir Figure 2.4). Ces signaux sont ensuite pré-codés par le module de code de correction d'erreur (« *Forward error correction* ») désigné par l'acronyme FEC (Voir Figure 2.4). Roberts *et al.* (2009) énonce que chaque flux de données est converti en signal analogique dans le domaine électrique par l'application d'un convertisseur numérique analogique (DAC). Un contrôleur de modulation (*modulator driver*) pilote et met en forme chacun des signaux analogiques (p. ex. contrôle de la tension, température, dissipation de la puissance, *etc.*) selon Roberts *et al.* (2009). Ces quatre signaux analogiques XI, XQ, YI, YQ modulent la source polarisée du laser accordable. Le codage des signaux en phase et des signaux en quadrature pour chaque polarisation orthogonale est effectué par chaque paire de modulateurs Mach-Zehnder (Voir Figure 2.4). Ensuite, les quatre signaux XI, XQ, YI, YQ convertis dans le domaine optique sont combinés à l'aide de coupleurs optiques, de

déphaseurs 90° et d'une lame quart d'onde $\lambda/4$ qui applique une rotation de 90° de la source polarisée pour l'obtention du mode de polarisation Y indiquant le champ magnétique transversale (*Voir* Figure 2.4). Les modes de polarisation X et Y de la source polarisée encodés par la modulation QPSK sont enfin multiplexés par un combineur des modes de polarisation (PBC), ce qui permet la transmission d'un signal optique unique caractérisé par la modulation DP-QPSK.

2.4.3 Récepteur cohérent DP-QPSK à 40 Gbit/s

Cette section a pour objectif la description de l'architecture photonique du récepteur cohérent DP-QPSK, la présentation des tolérances, des spécifications et des principales fonctions du module de traitement de signal numérique (DSP) et, enfin, l'énumération des paramètres de *monitoring* calculés par le module DSP.

2.4.3.1 Schéma du récepteur cohérent DP-QPSK

Le récepteur DP-QPSK intègre la technologie de détection cohérente qui repose sur la combinaison du signal optique DP-QPSK de la liaison par un signal optique émis par un laser accordable en tant qu'oscillateur local selon Roberts *et al.* (2009). La figure 2.5 présente le schéma du récepteur cohérent DP-QPSK du transpondeur cohérent déployé sur la liaison du réseau CANARIE et sur la liaison du réseau Verizon.

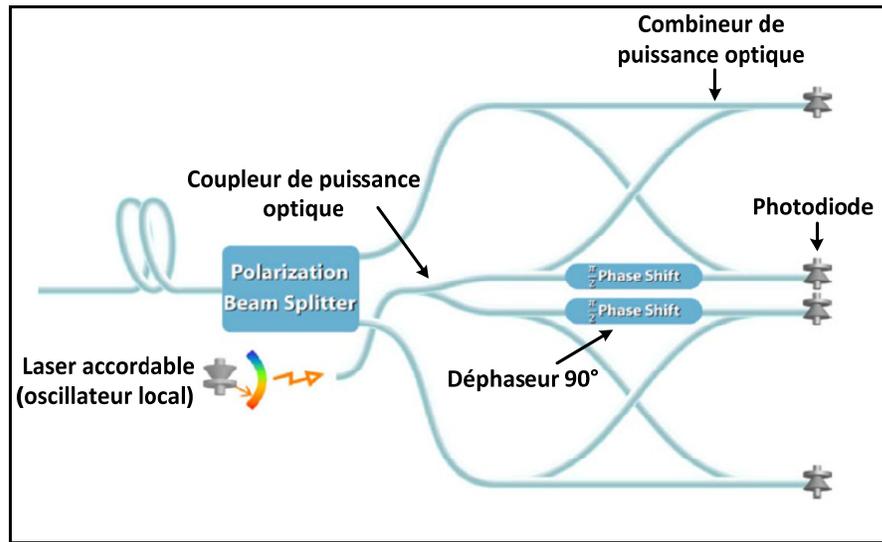


Figure 2.5 Schéma de l'architecture générale du récepteur cohérent
Adaptée de Roberts *et al.* (2009, p.3)

Les modes de polarisation orthogonaux du signal optique DP-QPSK de la liaison sont tout d'abord démultiplexés par un diviseur des modes de polarisation, appelé *Polarization Beam Splitter* (PBS) (Voir Figure 2.5). Les signaux en phase et en quadrature de chaque mode de polarisation X et Y sont divisés puis combinés avec chacune des phases mutuellement orthogonales de la source polarisée de l'oscillateur locale. La fréquence du laser accordable est ajustée de telle sorte que celle-ci se situe à quelques centaines de MHz de la fréquence du signal optique DP-QPSK pour la démodulation et l'application de fonctions de traitement sur les flux de données XI, XQ, YI, YQ par le module DSP d'après Roberts *et al.* (2009). Les signaux XI, XQ, YI, YQ sont démultiplexés à travers une architecture photonique disposant de coupleurs optiques de puissance, de déphaseurs 90° appelé $\pi/2$ *phase shift* et de combineurs optiques de puissance (Voir Figure 2.5). Ces signaux optiques sont ensuite détectés et convertis dans le domaine électrique par quatre photodiodes sur chacune des phases orthogonales des deux modes de polarisation du signal. Chaque signal électrique a par la suite été amplifié et converti en signal numérique par l'application d'un convertisseur analogique numérique (ADC). Un amplificateur à contrôle automatique de gain est intégré entre le récepteur cohérent et le module de traitement de signal numérique (module DSP) pour la bonne mise en forme des signaux électriques XI, XQ, YI, YQ. La section suivante

décrit les spécifications, les fonctions de traitement et les paramètres de *monitoring* calculés par le module DSP.

2.4.3.2 Module de traitement du signal numérique

La porte maximale du système de transmission optique DP-QPSK à 40 Gbit/s est spécifiée à 2200 km de fibre SMF (*Single Mode Fiber*). Le module de traitement de signal numérique (module DSP) est ainsi capable de compenser jusqu'à 50 000 ps/nm de dispersion chromatique, jusqu'à 25 ps de PMD équivalent à une compensation maximale de DGD instantanée de 75 ps – 100 ps d'après Roberts *et al.* (2009). Le seuil limite de performance du système de transmission 40G DP-QPSK est égale à 3.8×10^{-3} , ce qui correspond au taux d'erreur binaire limite avant application du code de correction d'erreur (Pre-FEC BER). Le taux d'erreur binaire (BER) est corrigé jusqu'à 10^{-15} après l'application du code de correction d'erreur (FEC). La figure 2.6 montre le schéma des principales fonctions du module de traitement du signal numérique appliquées sur les signaux XI, XQ, YI, YQ du signal DP-QPSK transmis sur la liaison du réseau CANARIE et sur la liaison du réseau Verizon.

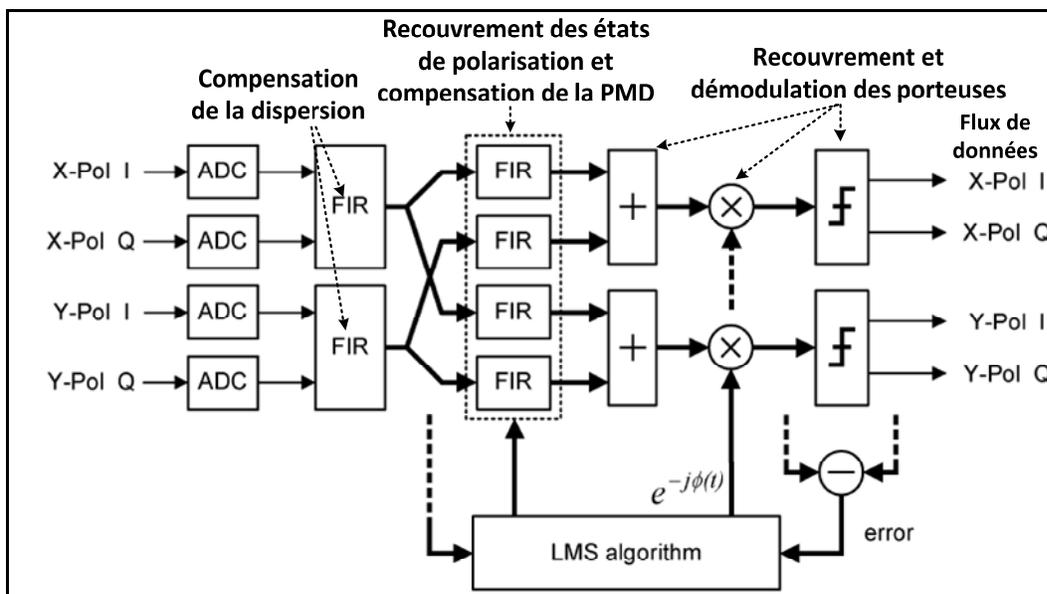


Figure 2.6 Schéma fonctionnel du module de traitement de signal électronique du transpondeur cohérent à 40 Gbit/s
Adaptée de Roberts *et al.* (2009, p.5)

Le module de traitement de signal numérique est implémenté par une puce électronique capable d'effectuer 12×10^{12} opérations de calcul par seconde d'après Roberts *et al.* (2009). La première fonction implémentée par le module DSP est la compensation de la dispersion chromatique du signal en combinant dans un filtre numérique à réponse impulsionnelle finie (FIR) le signal en phase et le signal en quadrature de chaque polarisation orthogonale (*Voir* Figure 2.6). Le filtre numérique à réponse impulsionnelle finie calcule et applique la fonction inverse des effets de propagation du signal DP-QPSK pour la compensation de la valeur totale de la dispersion chromatique. La seconde fonction implémentée dans le module DSP a pour objectif le suivi des états de polarisation du signal DP-QPSK et la compensation de la dispersion des modes de polarisation (PMD). La figure 2.6 montre quatre filtres numériques imbriqués pour la combinaison des signaux en phases et des signaux en quadrature des polarisations orthogonales X et Y. Ce type de configuration, disposant de quatre entrées et de quatre sorties, permet de suivre rapidement les principaux états de polarisation du signal DP-QPSK et de mesurer le retard de propagation entre les modes de polarisation (*Voir* Figure 2.6) d'après Roberts *et al.* (2009). Le démultiplexage des signaux XI, XQ, YI, YQ, c'est-à-dire la démodulation des porteuses (*Carrier recovery*) du signal DP-QPSK, constitue la troisième principale fonction du module DSP. Les porteuses du signal optique sont récupérées à travers un circuit de prédiction électronique (*Feedforward circuit*) implémentant l'algorithme des moindres carrés (*Least Mean Square Algorithm*) qui a pour objectif la minimisation des écarts de fréquence et de phase entre le signal optique DP-QPSK et la source de l'oscillateur local d'après Roberts *et al.* (2009) (*Voir* Figure 2.6). Le résultat des fonctions du module DSP se présente sous la forme de coefficients des filtres numériques. Les paramètres de *monitoring* calculés à partir des coefficients de filtres numériques sont selon Roberts *et al.* (2009) :

- les valeurs de dispersion chromatique (CD) du signal optique DP-QPSK en ps/nm;
- les valeurs de PMD exprimées en ps qui désignent la moyenne des valeurs de DGD;
- les valeurs de DGD exprimées en ps;
- les valeurs de PDL exprimées en dB;

- les valeurs de vecteurs Stokes (S_1 , S_2 , S_3) qui désignent la représentation des états de polarisation du signal sur la sphère de Poincaré;
- les valeurs de rapport signal électrique sur bruit désignées par SNR_x et par SNR_y spécifiques à chaque mode de polarisation orthogonal, exprimées en dB;
- les valeurs du taux d'erreur binaire avant l'application du code de correction d'erreur (pre-FEC BER).

Les valeurs de DGD, de PDL, de pre-FEC BER et de vecteurs Stokes (S_1 , S_2 , S_3) sont calculées avec une fréquence d'échantillonnage égale à 1 kHz alors que la fréquence d'échantillonnage du calcul de valeurs de CD, de PMD, de SNR_x et SNR_y fluctue entre 1 Hz de 10 Hz. Le *monitoring* des effets de polarisation (PMD, PDL, SOP) ou des paramètres de performance (SNR_x , SNR_y , pre-FEC BER) peut ainsi être effectué à l'aide des capacités de calcul du module DSP. Des paramètres diagnostiques du système de transmission optique DP-QPSK à 40 Gbit/s (Ex : coefficients des amplificateurs à contrôle automatique de gain, etc.) sont aussi extraits du module DSP lors des expériences de *monitoring* réalisées sur la liaison du réseau CANARIE et sur la liaison du réseau Verizon.

2.5 Fonction « polarimètre » du synthétiseur de polarisation Agilent N7786B

Le synthétiseur de polarisation Agilent N7786B a été déployé au site du terminal en Californie pour effectuer des mesures indépendantes des états de polarisation parallèlement aux mesures des paramètres de *monitoring* réalisées avec le transpondeur cohérent sur la liaison du réseau Verizon (*Voir* Figure 2.2).

Cet instrument peut être configuré en mode autonome pour mesurer avec précision durant de longues périodes (plusieurs heures à plusieurs mois) les vecteurs Stokes (S_1 , S_2 , S_3), ce qui en fait un instrument de choix pour la caractérisation des fluctuations des états de polarisation. Ce synthétiseur de polarisation est constitué d'un contrôleur de polarisation en Niobate de Lithium et d'un analyseur de polarisation selon la fiche technique d'Agilent (2013). La figure

2.7 présente le schéma à haut niveau de l'analyseur de polarisation d'après la fiche technique d'Agilent (2013).

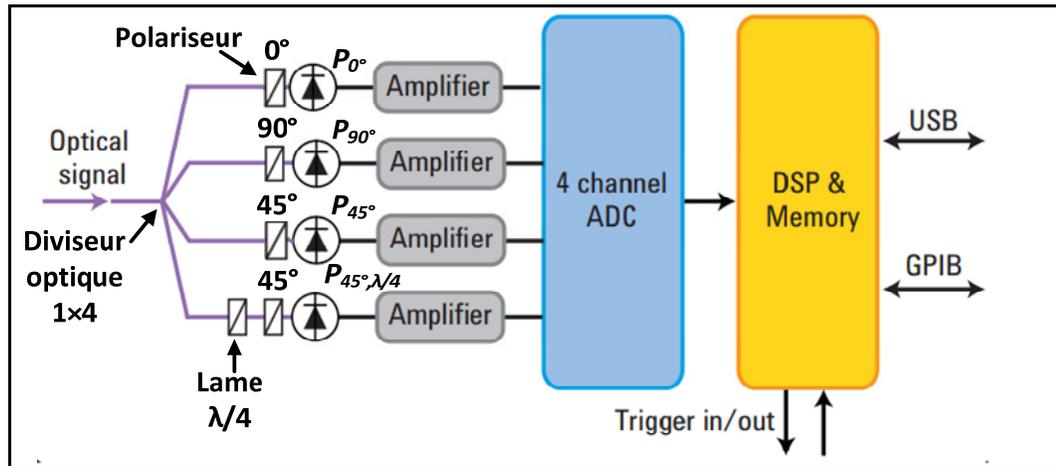


Figure 2.7 Schéma à haut niveau de l'analyseur de polarisation intégré au synthétiseur de polarisation
Adaptée de la fiche technique d'Agilent (2013)

L'analyseur de polarisation, commandé par la carte microcontrôleur définit la fonction « polarimètre » intégré au synthétiseur de polarisation (*Voir* Figure 2.7). Le polarimètre a pour objectif la mesure des puissances de plusieurs modes de polarisation linéaire du signal optique pour que le module DSP puisse calculer les valeurs de vecteurs Stokes. Hui *et al.* (2006, p.193) mentionnent que seule la mesure des puissances déterminées par quatre états de polarisation linéaire du signal optique suffisent pour le calcul des composantes du vecteur Stokes (S_1 , S_2 , S_3). Hui *et al.* (2006, p.193) décrit les puissances des quatre états de polarisations linéaires mesurées du signal optique nécessaires au calcul des vecteurs Stokes :

- P_{0° : puissance optique du mode de polarisation linéaire horizontale;
- P_{90° : puissance optique du mode de polarisation linéaire verticale;
- P_{+45° : puissance optique du mode de polarisation linéaire $+45^\circ$;
- $P_{+45^\circ, \lambda/4}$: puissance optique du mode de polarisation linéaire $+45^\circ$ après que l'axe et la direction de la polarisation du signal optique a été déphasé de 90° .

Le signal optique émis par le modem à 10 Gbit/s à la longueur d'onde λ_{10G} après propagation dans la liaison est préalablement séparé en quatre par un diviseur optique de puissance 1×4 disposé à l'entrée du polarimètre. La figure 2.7 montre que les puissances des quatre états de polarisation linéaire sont mesurées en parallèle et de façon indépendante à travers des dispositifs optiques sensibles à la polarisation constitués de polariseurs linéaires et d'une lame quart d'onde ($\lambda/4$). Le polariseur sélectionne un mode de polarisation linéaire particulier du signal alors que la lame quart d'onde modifie la polarisation de la lumière la traversant en appliquant un retard de 90° sur un des principaux états de polarisation du signal optique conduisant à modifier la nature et la direction de polarisation du signal à la sortie de ce composant photonique. Les trois premières sorties optiques du diviseur sont constituées d'un polariseur déterminé par un angle d'orientation de 0° , de 90° et de 45° respectivement. La quatrième sortie est constituée d'une lame quart d'onde $\lambda/4$ et d'un polariseur $+45^\circ$ (Voir Figure 2.7). Les puissances optiques P_{0° , P_{90° , P_{+45° et $P_{+45^\circ, \lambda/4}$ des différentes polarisations linéaires sont mesurées par des photodétecteurs disposés sur chacune des sorties (Voir Figure 2.7). Les quatre signaux électriques obtenus sont ensuite amplifiés puis convertis numériquement avant d'être analysés par le module de traitement de signal numérique (module DSP) (Voir Figure 2.7). L'équation (2.1) formule le calcul appliqué par le module de traitement de signal numérique (module DSP) intégré à la réception de l'analyseur de polarisation tirée de Hui *et al.* (2006, p.193).

$$\begin{pmatrix} S_0 \\ S_1 \\ S_2 \\ S_3 \end{pmatrix} = \begin{pmatrix} P_{0^\circ} + P_{90^\circ} \\ P_{0^\circ} - P_{90^\circ} \\ P_{+45^\circ} - P_{-45^\circ} \\ P_{+45^\circ, \lambda/4} - P_{-45^\circ, \lambda/4} \end{pmatrix} = \begin{pmatrix} P_{0^\circ} + P_{90^\circ} \\ P_{0^\circ} - P_{90^\circ} \\ 2P_{+45^\circ} - S_0 \\ 2P_{+45^\circ, \lambda/4} - S_0 \end{pmatrix} \quad (2.1)$$

L'équation (2.2) formule le calcul du degré de polarisation réalisé par le module DSP selon Hui *et al.* (2006, p.191).

$$DOP = \frac{\sqrt{S_1^2 + S_2^2 + S_3^2}}{S_0} \quad (2.2)$$

2.6 Méthode d'acquisition des données

Cette section de chapitre a pour objectif la description d'une méthode d'acquisition de données sous la forme de scripts et de programmes d'acquisition développés lors des expériences de *monitoring* réalisées sur la liaison du réseau CANARIE et sur la liaison du réseau Verizon. Chaque cycle d'acquisition comporte une période d'acquisition des données et une période de transfert des données vers le disque dur.

2.6.1 Méthode d'acquisition des données sur la liaison du réseau CANARIE

Les paramètres de *monitoring* décrits précédemment dans la section 2.4.3.2 sont mesurés avec le transpondeur cohérent DP-QPSK à 40 Gbit/s configuré par une fréquence d'échantillonnage égale à 1 Hz qui désigne la fréquence des mesures paramétrée par défaut par le système de transmission. Les données sont sauvegardées sur le disque dur du serveur lors des périodes d'acquisition des données après qu'une connexion Telnet a été établie avec le modem cohérent pour accéder aux interfaces d'acquisition (*Voir ANNEXE I, p.125*). Celles-ci indiquent les valeurs des paramètres de *monitoring* mesurés en temps réel par le transpondeur cohérent. Ces paramètres de *monitoring* sont ensuite capturés par l'application de la fonction *TCPdump* offerte dans l'environnement linux installé sur le serveur. Le résultat de l'application de cette fonction se présente sous la forme de fichiers au format binaire qui comportent les valeurs des paramètres de *monitoring* issues des interfaces d'acquisition proposés par le transpondeur cohérent. Un analyseur de texte (*Parser*) PHP (*Hypertext PreProcesseur*) a été développé par Tremblay *et al.* (2012) pour la conversion des fichiers au format binaire vers des fichiers textes structurés déterminés par l'extension CSV (*Comma-separated values*) (*Voir ANNEXE II, p.127*). Le script développé a pour objectif la détection des valeurs de paramètres de *monitoring* et de la mise en forme des données dans un fichier qui structure les valeurs des paramètres de *monitoring* (*Voir ANNEXE III, p.129*). Le résultat de chaque période d'acquisition des données se présente sous la forme d'un fichier au format CSV. 80 000 valeurs de paramètres de *monitoring* peuvent ainsi être mesurées et enregistrées par le transpondeur cohérent pour une période de 24 h. Près de 1.8 million de valeurs de paramètres énoncés dans la section 2.4.3.2 ont été mesurées du 14 octobre 2011 au 17

novembre 2011 par le transpondeur cohérent. Des périodes de maintenance et des périodes de réinitialisation du transpondeur cohérent se sont produites durant l'expérience de *monitoring* sur la liaison du réseau CANRIE, ce qui a provoqué des interruptions pouvant durer quelques heures. Les périodes de maintenance avaient pour objectif la validation des données mesurées par le transpondeur cohérent alors que les périodes de réinitialisation étaient causées par des arrêts spontanés du transpondeur cohérent.

2.6.2 Méthode d'acquisition des données sur la liaison du réseau Verizon

Un programme d'acquisition a été préalablement développé par Crawford *et al.* (2013) pour le pilotage du commutateur optique, du transpondeur cohérent et du polarimètre ainsi que pour configurer la mesure des vecteurs Stokes (S_1 , S_2 , S_3) et du degré de polarisation (DOP) à toutes les 50 μ s (correspondant à une fréquence d'échantillonnage égale à 20 kHz) avec le polarimètre.

La période d'acquisition des données et la période de sauvegarde des données sont déterminées par deux fonctions distinctes du programme d'acquisition lors de l'expérience de *monitoring* réalisée sur la liaison du réseau Verizon. La première fonction initialise et active les mesures des paramètres de *monitoring* avec le transpondeur cohérent et les mesures des vecteurs Stokes avec le polarimètre. La deuxième fonction a pour rôle l'interruption de la mesure pour le transfert des données stockées dans la mémoire du transpondeur cohérent et du polarimètre vers le disque dur.

2.6.2.1 Programme d'acquisition des données avec le transpondeur cohérent

Le transpondeur cohérent a été configuré de manière à ce que les coefficients des filtres numériques et les composantes du vecteur Stokes soient acquis à une fréquence d'échantillonnage égale à 1 kHz (*Voir* section 2.4.3.2). La figure 2.8 décrit le diagramme de fonctions du programme d'acquisition de données exécuté au commutateur optique en lien avec le transpondeur cohérent du 25 au 30 octobre 2012.

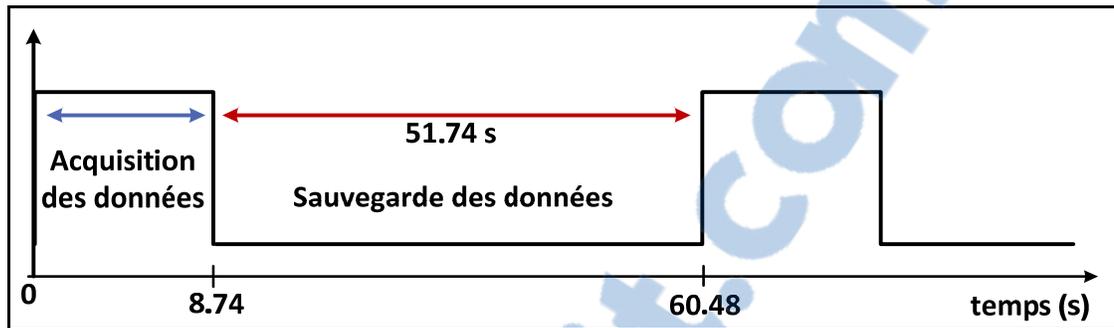


Figure 2.8 Diagramme de fonctions du programme d'acquisition des données avec le transpondeur cohérent du 25 au 30 octobre 2012

La durée d'acquisition des données est égale à 8.74 s et le temps de transfert des données de la mémoire du transpondeur cohérent vers le disque dur est égal à 51.74 s. La durée du cycle d'acquisition et du cycle de sauvegarde des données est donc égale à 60.48 s (*Voir* Figure 2.8). De cette manière, 8738 valeurs de chaque paramètre de *monitoring* sont mesurées et sauvegardées durant chaque cycle de mesure du 25 au 30 octobre 2012. La figure 2.9 décrit le diagramme de fonctions du programme d'acquisition exécuté au commutateur optique en lien avec le transpondeur cohérent du 5 au 11 novembre 2012.

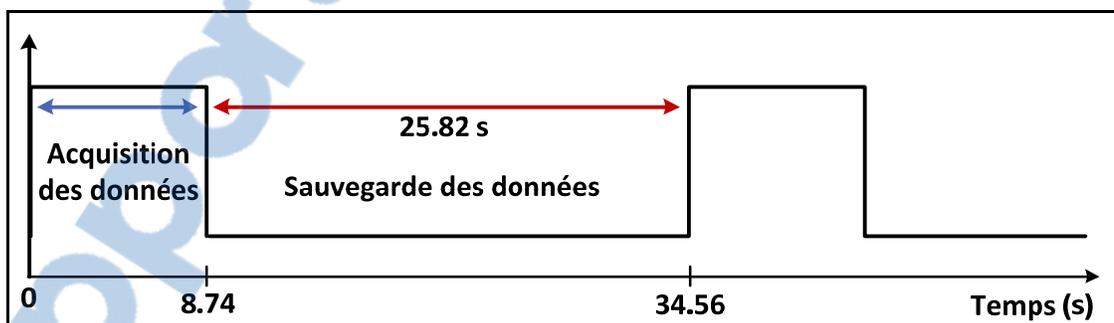


Figure 2.9 Diagramme de fonctions du programme d'acquisition des données avec le transpondeur cohérent du 5 au 11 novembre 2012

Durant la période de mesure du 5 au 11 novembre 2012, la durée de la sauvegarde des données a été réduite par deux et est à présent égale à 25.82 s (*Voir* Figure 2.9). Le programme d'application exécuté sur le transpondeur cohérent et sur le commutateur optique génère 7536 fichiers de données structurées au format CSV du 25 au 30 octobre 2012 et 14356 fichiers CSV du 5 au 11 novembre 2012 (*Voir* ANNEXE III, p.128). Plus de 170

millions de valeurs de paramètres de *monitoring* ont été mesurées et sauvegardées par le transpondeur cohérent pour les deux périodes de mesure sur la liaison du réseau Verizon.

2.6.2.2 Programme d'acquisition des données avec le polarimètre

La figure 2.10 décrit le diagramme des fonctions du programme d'acquisition des données exécuté au commutateur optique avec le polarimètre sur toutes les périodes de mesure (*Voir* section 2.3.2).

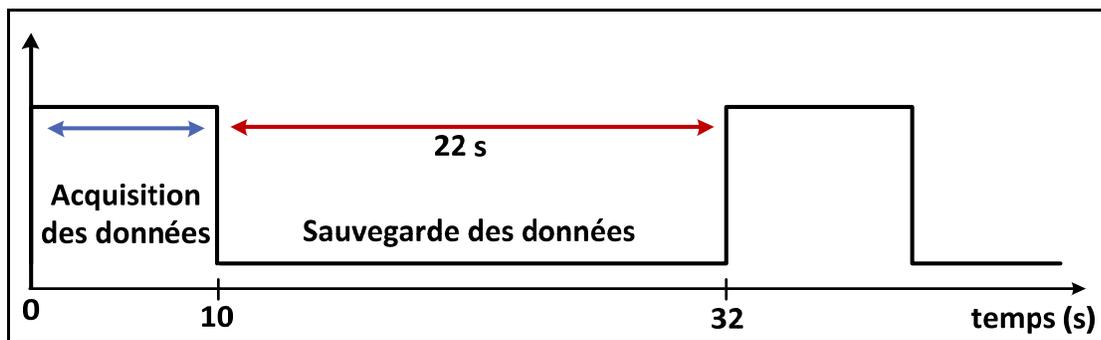


Figure 2.10 Diagramme de fonctions du programme d'acquisition des données enregistrées avec le polarimètre

Chaque période d'acquisition des données dure 10 s. La fréquence d'échantillonnage des mesures du vecteurs Stokes (S_1 , S_2 , S_3) et du degré de polarisation (DOP) est égale à 20 kHz par le polarimètre, ce qui permet la collecte de 200 000 données lors des périodes d'acquisition. Le temps de transfert des données de la carte mémoire du polarimètre vers le disque dur est égal à 22 s. Le polarimètre génère et transmet les valeurs de vecteurs Stokes, les valeurs du degré de polarisation et les valeurs de puissance optique à la réception dans trois matrices distinctes (*Voir* ANNEXE IV, p.131), elles-mêmes mises en structure et sauvegardées dans un fichier Matlab à la fin de chaque période d'acquisition des données (*Voir* Figure 2.10). 39 000 fichiers Matlab ont été sauvegardés du 25 au 30 octobre 2012, du 5 au 11 novembre 2012 et du 4 au 19 septembre 2013 désignant plus de 7.8 milliards de valeurs enregistrées avec le polarimètre lors de l'expérience de *monitoring* réalisée sur la liaison du réseau Verizon.

CHAPITRE 3

MONITORING ET CARACTERISATION DES VALEURS DE DGD ET DE PDL AVEC UN TRANSPONDEUR COHERENT DP-QPSK À 40 GBIT/S

3.1 Introduction

Ce chapitre décrit le traitement, les méthodes de calcul et l'analyse des mesures réalisées avec un transpondeur cohérent Ciena DP-QPSK à 40 Gbit/s sur une liaison optique majoritairement aérienne du réseau de Verizon et sur une liaison optique majoritairement enterrée du réseau de CANARIE.

Le transpondeur cohérent, c'est-à-dire la carte OCLD, a été utilisé en tant qu'instrument de mesures des paramètres et des performances de *monitoring* pour apporter une meilleure compréhension des effets de polarisation dégradant la qualité de la transmission du signal optique sur différentes installations. Les méthodes de caractérisation décrites dans ce chapitre s'inscrivent dans la lignée des travaux de *monitoring* précédemment réalisés par Waddy *et al.* (2005), par Kawasaki *et al.* (2007), par Woodward *et al.* (2012), par Woodward *et al.* (2014).

Ma contribution porte sur le développement de quatre outils d'analyse, sous la forme de scripts Matlab, appliqués sur les valeurs de DGD et les valeurs de PDL mesurées à chaque seconde sur le lien du réseau Canarie et mesurées à chaque milliseconde sur lien du réseau Verizon. Ces outils d'analyse intègrent des méthodes de calcul et d'analyse des résultats décrites de la section 3.3 jusqu'à la section 3.6.

Une méthode de traitement préliminaire a été développée pour la conversion des données stockées dans les fichiers CSV dans une structure de données indexée conçue dans l'environnement Matlab pour faciliter le développement et l'évolution de chacune des méthodes de calcul appliquées sur les données enregistrées avec le transpondeur cohérent.

3.2 Traitement des données enregistrées par le transpondeur cohérent

Le traitement des données a pour objectif final la création d'une structure de données indexée par « clé » dans l'environnement Matlab constituée de 21 592 matrices de paramètres de *monitoring* sur une période de 10 jours de mesures réalisées par le transpondeur cohérent sur la liaison du réseau de Verizon (*Voir* section 2.2.2). Chaque « clé » désigne la matrice spécifique d'un paramètre de *monitoring* mesuré par le transpondeur cohérent (*Voir* section 2.6). Un script Matlab a été développé pour la conception de la structure de données indexée et comporte plusieurs processus pour l'automatisation du traitement des fichiers CSV (*comma-separated values*). La figure 3.1 décrit la séquence de traitement appliquée sur les fichiers CSV constitués chacun de 8738 valeurs pour chaque paramètre de *monitoring* mesuré avec le transpondeur cohérent.

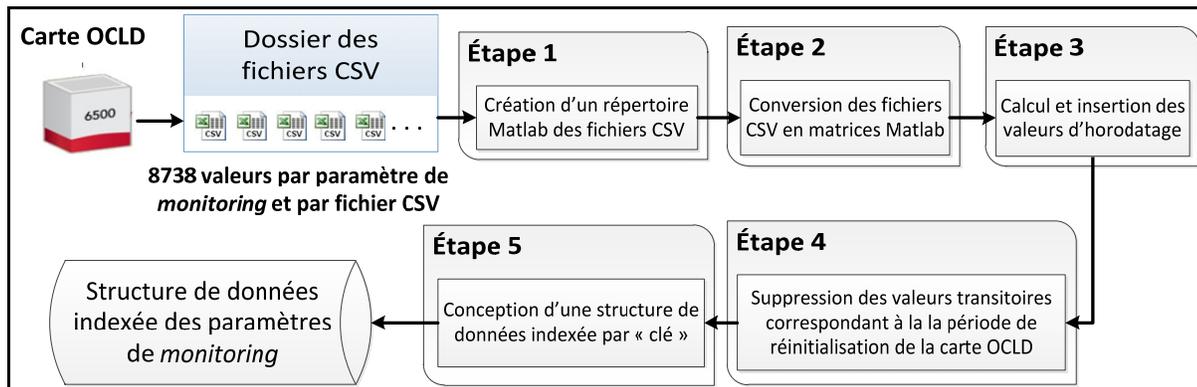


Figure 3.1 Séquence de traitement des fichiers de mesure enregistrés par la carte OCLD

Un script Matlab a été développé pour l'automatisation du traitement des fichiers CSV enregistrés par le transpondeur cohérent vers une structure de donnée indexée unique (*Voir* ANNEXE V, p.133). L'exécution de ce script est identique à un programme qui exécute toutes les étapes de traitement décrites par la figure 3.1 sur chaque fichier CSV.

La première étape consiste à concevoir un répertoire d'adresses des fichiers CSV (*Voir* Figure 3.1). Ce répertoire d'adresses est une structure de données qui indexe la liste des chemins d'accès des paramètres de *monitoring* enregistrés dans chaque fichier CSV (*Voir*

ANNEXE VI, p. 137). La lecture directe des fichiers CSV est lente et génère beaucoup d'erreurs dans l'environnement Matlab. Le répertoire Matlab des fichiers CSV facilite l'accès aux données enregistrées par le transpondeur cohérent et l'automatisation de la séquence de traitement décrites à la figure 3.1.

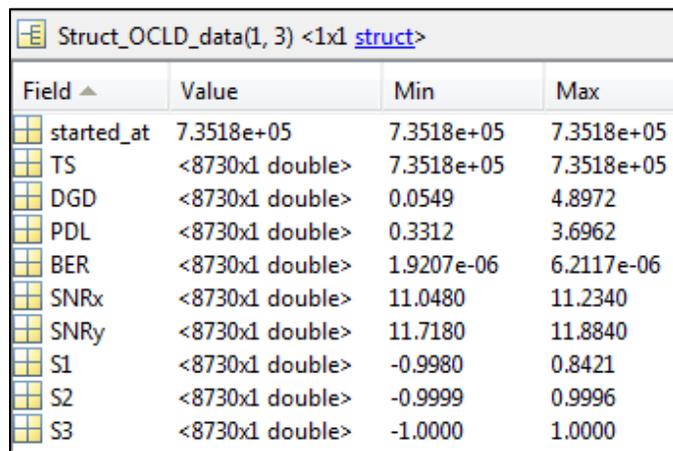
La seconde étape consiste à convertir les paramètres de *monitoring* de chaque fichier CSV en matrices Matlab. Deux étapes de traitement comportent l'automatisation de la conversion des fichiers CSV en matrices Matlab. Les fichiers CSV sont tout d'abord convertis dans des fichiers textes numériques lors de la première étape de traitement (*Voir* ANNEXE VII, p. 139) en utilisant le mode de lecture binaire. Les fichiers textes numériques sont par la suite convertis en matrices Matlab lors de la seconde partie de cette étape de traitement. Tous les paramètres de *monitoring* ont été capturés à partir du fichier texte numérique par l'application de la fonction *textscan* (*Voir* ANNEXE VIII, p. 141). Les valeurs de chronomètre *time_ms*, de rapport signal sur bruit électrique SNR_X du mode de polarisation X, de rapport signal sur bruit SNR_Y du mode de polarisation Y, de PMD instantanée désignant la valeur de DGD (*Voir* section 1.3), de PDL instantanée et de vecteurs Stokes (S_1 , S_2 , S_3) ont été sélectionnées parmi les paramètres de *monitoring* mesurés par le transpondeur cohérent (*Voir* section 2.6.2.1). Le résultat de cette seconde étape de traitement se présente sous la forme d'une matrice Matlab constituée de ces paramètres de *monitoring*.

Les valeurs d'horodatage ont été créées et associées à chaque paramètre mesuré lors de la troisième étape de traitement. Cette étape de traitement consiste à former et à insérer les valeurs d'horodatage pour chaque matrice Matlab constituée des paramètres de *monitoring* sélectionnés lors de l'étape de traitement précédente. Le résultat de cette troisième étape se présente sous la forme d'une matrice constituée des valeurs de paramètres de *monitoring* et des valeurs d'horodatage synchronisées (*Voir* ANNEXE IX, p. 143).

La quatrième étape de traitement des données est la suppression des valeurs transitoires mesurées durant la période de réinitialisation du transpondeur cohérent pour chaque période d'acquisition (*Voir* Figure 3.1). Le temps de réinitialisation de la carte OCLD peut durer

quelques secondes lors de la période d'acquisition des données. La période de réinitialisation du transpondeur cohérent est identifiée lorsque la valeur de DGD mesurée et la valeur de PDL mesurée fluctuent de 30 à 160 ps et de 9 à 30 dB respectivement. Durant la période de réinitialisation, les valeurs de DGD et de PDL sont égales à -1 alors que les valeurs de pre-FEC BER sont égales à +1 ce qui signifie la perte du signal optique par la carte OCLD. Le résultat se présente sous la forme d'une matrice réduite constituée seulement de mesures valides des paramètres de *monitoring* dont le nombre de valeurs fluctue entre 2000 et 8738 valeurs (*Voir ANNEXE X, p. 145*).

La dernière étape de traitement consiste à sauvegarder chacun des paramètres de *monitoring* stockés dans les matrices précédemment obtenues sous la forme d'une structure de données indexée par « clé » (*Voir Figure 3.1*). La structure de données conçue dispose de plusieurs « compartiments ». Chaque « compartiment » correspond à la clé d'accès spécifique à un paramètre de *monitoring*, aux valeurs d'horodatage associées et au temps de début de chaque mesure désigné par la variable *started_at* (*Voir ANNEXE XI, p. 147*). La figure 3.2 présente la structure de donnée indexée par « clé » des valeurs de paramètres de *monitoring* horodatées et des valeurs de temps de début *started_at* lors d'une période d'acquisition des données avec le transpondeur cohérent.



Field ▲	Value	Min	Max
started_at	7.3518e+05	7.3518e+05	7.3518e+05
TS	<8730x1 double>	7.3518e+05	7.3518e+05
DGD	<8730x1 double>	0.0549	4.8972
PDL	<8730x1 double>	0.3312	3.6962
BER	<8730x1 double>	1.9207e-06	6.2117e-06
SNRx	<8730x1 double>	11.0480	11.2340
SNRy	<8730x1 double>	11.7180	11.8840
S1	<8730x1 double>	-0.9980	0.8421
S2	<8730x1 double>	-0.9999	0.9996
S3	<8730x1 double>	-1.0000	1.0000

Figure 3.2 Structure de données indexée par « clé » des paramètres de *monitoring* lors d'une période d'acquisition réalisée avec le transpondeur cohérent

Les « clés » de la structure de données indexée sont des champs pour l'accès aux valeurs d'un paramètre en particulier (p. ex. valeurs de DGD). Les « clés » sont définies par la première colonne à gauche et les matrices associées à chacune des clés sont présentées dans la seconde colonne sur la figure 3.2.

3.3 Évolution temporelle des valeurs de DGD et des valeurs de PDL

La première analyse spécifique de données a porté sur la comparaison des courbes d'évolution temporelle des valeurs de DGD et des valeurs de PDL mesurées par le transpondeur cohérent sur le lien du réseau Canarie et sur le lien du réseau Verizon. Kawasaki *et al.* (2007) et Woodward *et al.* (2014) proposent de caractériser l'activité en polarisation d'un lien optique cohérent par le calcul de la dérivée temporelle, c'est-à-dire des fluctuations des valeurs de DGD et de PDL en fonction du temps. Cette étude a pour objectif de connaître les principales caractéristiques des valeurs d'évolution temporelle et des valeurs de dérivée temporelle de DGD et de PDL mesurées sur la liaison du réseau Canarie et sur la liaison du réseau Verizon.

3.3.1 Méthode de calcul et d'analyse de l'évolution temporelle des valeurs de DGD et de PDL

La figure 3.3 décrit la méthode de calcul et d'analyse appliquée sur la structure de données indexée des paramètres de *monitoring* (Voir Figure 3.2) pour la caractérisation des valeurs de DGD et de PDL en fonction du temps sur la liaison du réseau CANARIE et sur la liaison du réseau Verizon.

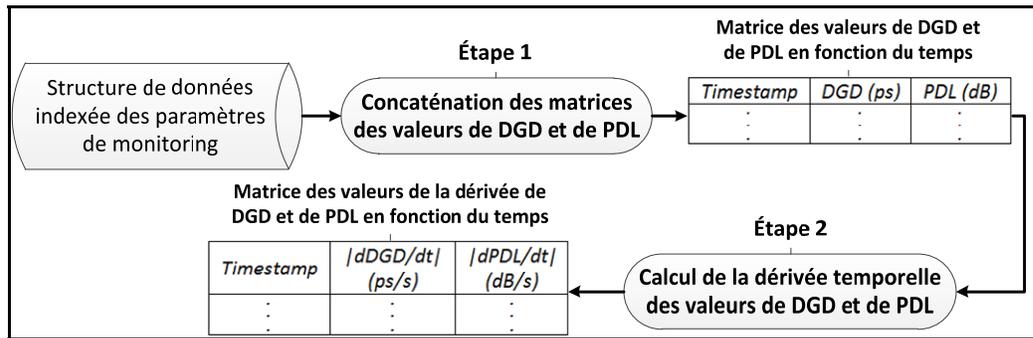


Figure 3.3 Méthode de calcul et d'analyse des valeurs de DGD et de PDL horodatées mesurées par le transpondeur cohérent

Ma contribution porte sur le développement d'un script Matlab qui inclut deux actions distinctes sur les valeurs de DGD et de PDL mesurées avec le transpondeur cohérent. La première action est la concaténation des valeurs de DGD et des valeurs de PDL en fonction du temps sur une période de 5 jours sur le lien du réseau Canarie et sur le lien du réseau Verizon. Le résultat de cette action se présente sous la forme d'une matrice qui regroupe l'ensemble des valeurs de DGD et de PDL en fonction du temps. Ce script Matlab est un programme qui permet de sélectionner les index des matrices en effectuant la comparaison du temps de début et du temps de fin de la période sélectionnée par l'utilisateur avec le temps de début désignée par la variable *started_at* de toutes les périodes d'acquisition réalisées avec le transpondeur cohérent (Voir ANNEXE XII, p.149). Lorsque tous les index de la structure de données ont été sélectionnés, alors les valeurs de DGD, de PDL et des temps de mesure correspondant sont chargées successivement dans une unique matrice. Cette matrice est constituée de près de 2 millions de valeurs de DGD et de PDL horodatées pour les mesures réalisées sur le lien du réseau Canarie et de près de 117 millions de valeurs pour les mesures réalisées sur le lien du réseau Verizon pour une période de 5 jours.

Les valeurs absolues de la dérivée temporelle sont ensuite calculées sur les valeurs de DGD et de PDL concaténées lors de la seconde étape de traitement pour la caractérisation des fluctuations de DGD et de PDL sur le lien du réseau Canarie et sur le lien du réseau Verizon. Le script Matlab développé permet de soumettre ce calcul à des contraintes particulières. En effet, la seconde partie du script Matlab assure le calcul de la dérivée temporelle des valeurs

de DGD et de PDL sur une durée strictement égale à la période d'échantillonnage séparant deux mesures adjacentes réalisées par le transpondeur cohérent (*Voir* ANNEXE XIII, p. 151). L'équation (3.1) formule le calcul de la dérivée temporelle appliqué sur les valeurs de DGD ou sur les valeurs de PDL mesurées par le transpondeur cohérent.

$$\frac{dx}{dt} = \left| \frac{x(t_2) - x(t_1)}{t_2 - t_1} \right| \quad (3.1)$$

Les valeurs de dérivée temporelle de DGD et de PDL sont désignées par la variable $\left| \frac{dDGD}{dt} \right|$ et par la variable $\left| \frac{dPDL}{dt} \right|$ respectivement. Une seconde matrice est créée après l'application du calcul de la dérivée temporelle par le script Matlab selon l'équation (3.1) sur les mesures réalisées par le transpondeur cohérent sur chacune des liaisons optiques. Cette matrice sauvegarde toutes les valeurs de $\left| \frac{dDGD}{dt} \right|$ et $\left| \frac{dPDL}{dt} \right|$ (*Voir* Figure 3.3).

3.3.2 Analyse des courbes d'évolution temporelle de DGD et de PDL

Un second script Matlab a été développé pour l'optimisation de l'affichage des courbes d'évolution temporelle des valeurs de DGD et de PDL mesurées sur le lien aérien du réseau Verizon et sur le lien enterrée du réseau Canarie (*Voir* ANNEXE XIV, p. 150). La figure 3.4 décrit les courbes d'évolution temporelle des valeurs de DGD et de PDL mesurées avec le transpondeur cohérent du 26 au 30 octobre 2011 sur le lien enterré du réseau Canarie.

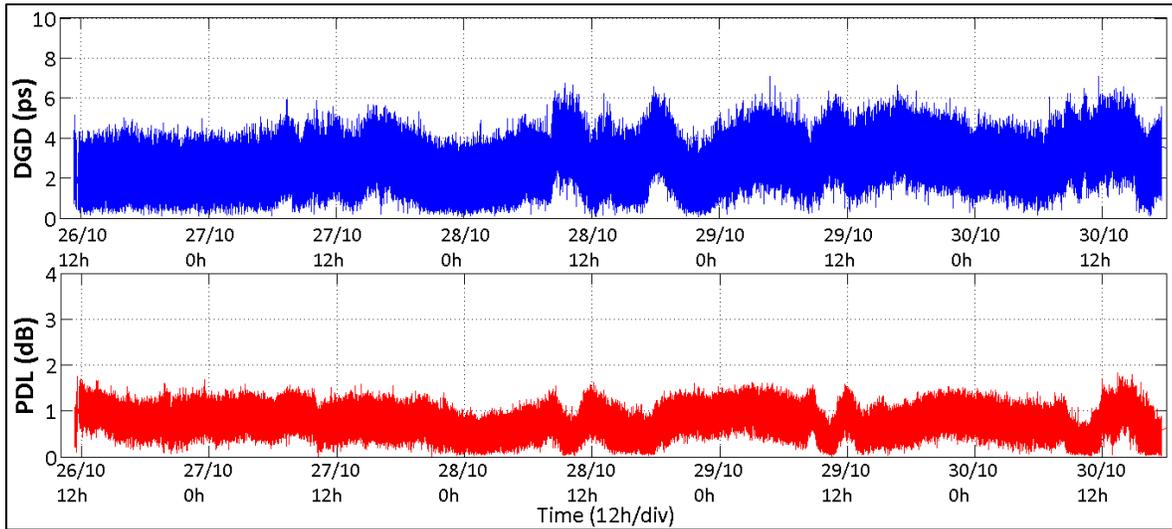


Figure 3.4 Courbes d'évolution temporelle des valeurs de DGD et des valeurs de PDL mesurées à chaque seconde par le transpondeur cohérent

La figure 3.5 décrit les courbes d'évolution temporelle des valeurs de DGD et de PDL mesurées du 5 au 11 novembre 2012 avec le transpondeur cohérent sur le lien du réseau Verizon.

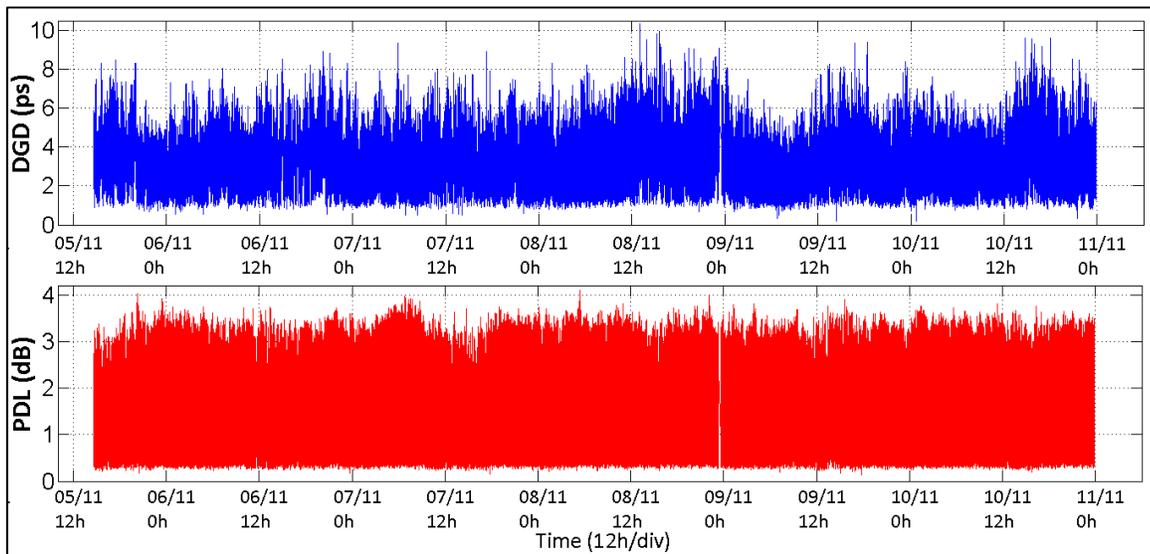


Figure 3.5 Courbes d'évolution temporelle des valeurs de DGD et des valeurs de PDL mesurées à chaque milliseconde par le transpondeur cohérent

Le tableau 3.1 présente la moyenne, l'écart type et la valeur maximale des courbes d'évolution temporelle de DGD et de PDL ainsi que des valeurs de dérivée temporelle respectives pour la comparaison des mesures réalisées sur la liaison majoritairement enterrée du réseau Canarie et sur la liaison majoritairement aérienne du réseau Verizon.

Tableau 3.1 Caractérisation des mesures de DGD, de $|dDGD/dt|$, de PDL et de $|dPDL/dt|$ par le calcul de la moyenne, de l'écart type et de la valeur maximale

Liaison	DGD (ps)		$ dDGD/dt $ (ps/s)		PDL (dB)		$ dPDL/dt $ (dB/s)	
	Canarie	Verizon	Canarie	Verizon	Canarie	Verizon	Canarie	Verizon
Moyenne	2.761	3.149	0.785	1.008	0.761	1.907	0.191	0.211
Écart type	0.933	1.374	0.592	0.779	0.251	0.68	0.144	0.16
Maximum	7.111	13.24	4.509	8.97	1.822	4.47	1.68	3.25

Les valeurs de DGD, de $\left|\frac{dDGD}{dt}\right|$, de PDL et de $\left|\frac{dPDL}{dt}\right|$ sont supérieures sur la liaison du réseau Verizon que celles mesurées sur la liaison du réseau Canarie. La moyenne des valeurs de DGD et la moyenne des valeurs de PDL sont supérieures sur la liaison du réseau Verizon que celles calculées sur la liaison du réseau Canarie (*Voir* Tableau 3.1) en raison d'une distance de liaison supérieure pour la liaison du réseau de Verizon (*Voir* section 2.2). L'écart type des valeurs de DGD et l'écart type des valeurs de PDL sur la liaison du réseau Verizon sont aussi nettement supérieurs que celles calculées sur la liaison du réseau Canarie (*Voir* Tableau 3.1). Par ailleurs, on observe que la moyenne et l'écart type des dérivées temporelles $\left|\frac{dDGD}{dt}\right|$ et $\left|\frac{dPDL}{dt}\right|$ sont supérieures sur la liaison aérienne du réseau Verizon que sur la liaison du réseau Canarie. En effet, seulement 17 valeurs de $\left|\frac{dDGD}{dt}\right|$ sont supérieures à 4 ps lors des mesures réalisées sur la liaison du réseau Canarie alors qu'il existe 326 632 valeurs de $\left|\frac{dDGD}{dt}\right|$ supérieures à 4 ps sur la liaison du réseau Verizon. La probabilité qu'une valeur de $\left|\frac{dDGD}{dt}\right|$ soit supérieure à 4 ps est de l'ordre de 10^{-5} sur la liaison du réseau Canarie alors que cette

probabilité s'élève à 0.28 % pour la liaison du réseau Verizon. 8 valeurs de $\left|\frac{dPDL}{dt}\right|$ sont supérieures à 1 dB pour les mesures réalisées sur la liaison du réseau Canarie alors que l'on observe plus de 29241 valeurs de $\left|\frac{dPDL}{dt}\right|$ supérieures à 1 dB sur la liaison du réseau Verizon. La probabilité qu'il se produit une valeur de $\left|\frac{dPDL}{dt}\right|$ supérieure à 1 dB est égale à 5×10^{-6} sur la liaison du réseau Canarie alors que cette valeur de probabilité est égale à 2×10^{-4} sur la liaison du réseau Verizon (*Voir Figures 3.4 – 3.5*). Après analyse de ces résultats, les valeurs de DGD fluctuent plus significativement sur la liaison aérienne car ce type de liaison est soumis à de nombreuses perturbations externes (p. ex. variations de la température, vitesses de vent, etc.) comme cela est présenté dans les travaux de *monitoring* de Waddy *et al.* (2001), de Waddy *et al.* (2005) et de Kawasaki *et al.* (2007).

3.4 Moyenne et écart type des valeurs de DGD et de PDL

Woodward *et al.* (2012) et Woodward *et al.* (2014) ont calculé la moyenne des valeurs de DGD et des valeurs de PDL sur chaque minute de mesures pour la réduction du bruit généré par le système de transmission optique. Kawasaki *et al.* (2007) évalue les fluctuations caractéristiques des valeurs de DGD par le calcul de l'écart type pour chaque minute et pour chaque heure de mesures réalisées par le transpondeur cohérent. Cette section propose de caractériser avec approfondissement les fluctuations des valeurs de DGD et de PDL par le calcul de la moyenne et de l'écart type à chaque minute de mesures sur la liaison du réseau CANARIE et sur la liaison du réseau Verizon. Cette caractérisation s'inscrit ainsi dans la lignée des travaux précédemment réalisés par Kawasaki *et al.* (2007), Woodward *et al.* (2012) et Woodward *et al.* (2014). Afin que les valeurs de DGD et de PDL puissent être caractérisées avec approfondissement, un outil d'analyse a été développé pour le calcul des moyennes et des écarts types des valeurs de DGD et de PDL mesurées. La méthode de calcul de cet outil est décrit de façon synthétique dans la première partie de cette section. La seconde partie de cette section est l'analyse des résultats obtenus par l'application de l'outil d'analyse sur les même périodes de mesures que celles choisies dans la section précédente (*Voir section 3.3.2*).

3.4.1 Méthode de calcul des moyennes et des écarts types des valeurs de DGD et de PDL

L'outil d'analyse développé se présente sous la forme d'un script Matlab constitué de deux étapes de calcul. La figure 3.6 décrit la méthode de calcul de la moyenne et de l'écart type pour la caractérisation et pour la comparaison avec précision des mesures de DGD et de PDL sur le lien du réseau Canarie et sur le lien du réseau Verizon sur une période de 5 jours.

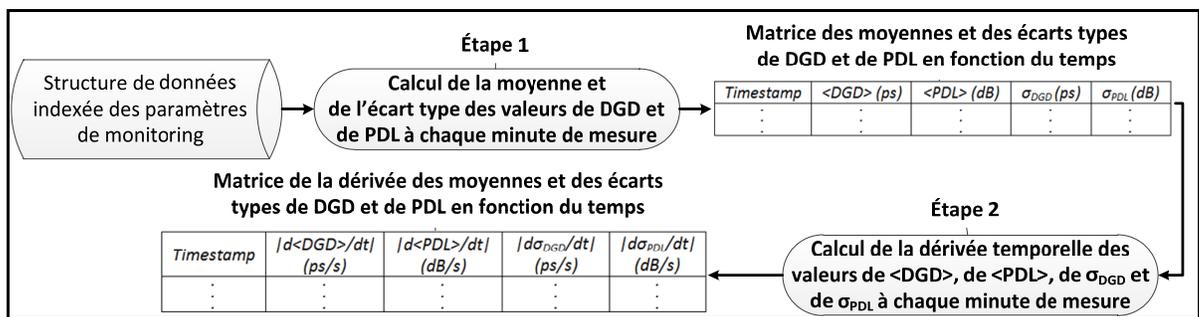


Figure 3.6 Méthode de calcul de la moyenne et de l'écart type des valeurs de DGD et de PDL ainsi que de leur dérivée temporelle respective pour chaque minute de mesure

La première étape de calcul porte sur la moyenne des valeurs de DGD et des valeurs de PDL représentées selon les variables <DGD> et <PDL> respectivement et sur l'écart type des valeurs de DGD et de PDL désignées par les variables σ_{DGD} et σ_{PDL} respectivement (Voir Figure 3.6). Le script développé sélectionne les index de la structure de données correspondant à la période de mesures sélectionnée par l'utilisateur sur la liaison du réseau CANARIE et sur la liaison du réseau Verizon. La moyenne et l'écart type des valeurs de DGD et de PDL sont calculés sur une période de mesures strictement égale à 1 min telle que réalisée par Woodward *et al.* (2012) (Voir ANNEXE XV, p. 155). Soit n le nombre de valeurs de DGD ou de PDL mesurées sur une minute de mesure, x_i la valeur de DGD ou la valeur de PDL correspondant à un instant de mesure précis et i la valeur de l'indice correspondant à la valeur de DGD ou de PDL, alors la moyenne $\langle x \rangle$ et l'écart type σ_x des valeurs de DGD ou de PDL sont calculées selon les équations (3.2 – 3.3) d'après Navidi (2011, p.14 -15).

$$\langle x \rangle = \frac{1}{n} \sum_{i=1}^n x_i \quad (3.2)$$

$$\sigma_x = \sqrt{\frac{1}{n} \sum_{i=1}^n (x_i - \langle x \rangle)^2} \quad (3.3)$$

Les moyennes et les écarts types pour chaque minute de mesures des valeurs de DGD et de PDL sont calculées par l'application des fonctions *mean* et *std* offertes par le module *Descriptive Statistics* de Matlab. Le résultat de cette première étape de calcul se présente sous la forme d'une matrice constituée des valeurs de $\langle DGD \rangle$, de $\langle PDL \rangle$, de σ_{DGD} et de σ_{PDL} en fonction du temps (*Voir* Figure 3.6).

Le second calcul est la dérivée temporelle des valeurs de $\langle DGD \rangle$, de $\langle PDL \rangle$, de σ_{DGD} et de σ_{PDL} ayant pour résultat les variables $\left| \frac{d\langle DGD \rangle}{dt} \right|$, $\left| \frac{d\langle PDL \rangle}{dt} \right|$, $\left| \frac{d\sigma_{DGD}}{dt} \right|$ et $\left| \frac{d\sigma_{PDL}}{dt} \right|$ respectivement. La seconde partie du script Matlab assure le calcul de la dérivée temporelle sur les deux liaisons optiques cohérentes en respectant strictement la durée égale à 1 min séparant deux mesures adjacentes (*Voir* ANNEXE XVI, p. 157). Le résultat de cette seconde étape de calcul se présente sous la forme d'une matrice constituée des valeurs de $\left| \frac{d\langle DGD \rangle}{dt} \right|$, de $\left| \frac{d\langle PDL \rangle}{dt} \right|$, de $\left| \frac{d\sigma_{DGD}}{dt} \right|$ et de $\left| \frac{d\sigma_{PDL}}{dt} \right|$ en fonction du temps (*Voir* Figure 3.6). 6306 valeurs et 7741 valeurs de $\langle DGD \rangle$, de $\langle PDL \rangle$, de σ_{DGD} et de σ_{PDL} ont été calculées et enregistrées dans les matrices présentées sur la figure 3.6 pour 5 jours de mesure sur la liaison optique du réseau CANARIE et sur la liaison optique du réseau Verizon respectivement.

3.4.2 Analyse des courbes d'évolution temporelle des moyennes et des écarts types des valeurs de DGD et de PDL pour chaque minute de mesure

Un script Matlab a été développé pour l'affichage des courbes d'évolution temporelle des valeurs de $\langle DGD \rangle$, $\langle PDL \rangle$, σ_{DGD} et σ_{PDL} ainsi que des valeurs de leurs dérivées temporelles respectives superposées sur le même graphique telle que présenté dans les travaux de Woodward *et al.* (2012) et de Woodward *et al.* (2014) (*Voir* ANNEXE XVII, p. 159). La

figure 3.7 décrit les courbes d'évolution temporelle des moyennes de DGD de PDL pour chaque minute de mesure, appelées $\langle DGD \rangle$ et $\langle PDL \rangle$, et de leurs dérivées temporelles respectives, appelées $\left| \frac{d\langle DGD \rangle}{dt} \right|$ et $\left| \frac{d\langle PDL \rangle}{dt} \right|$ pour chaque minute de mesures du 26 au 30 octobre 2011 sur le lien optique cohérent du réseau CANARIE.

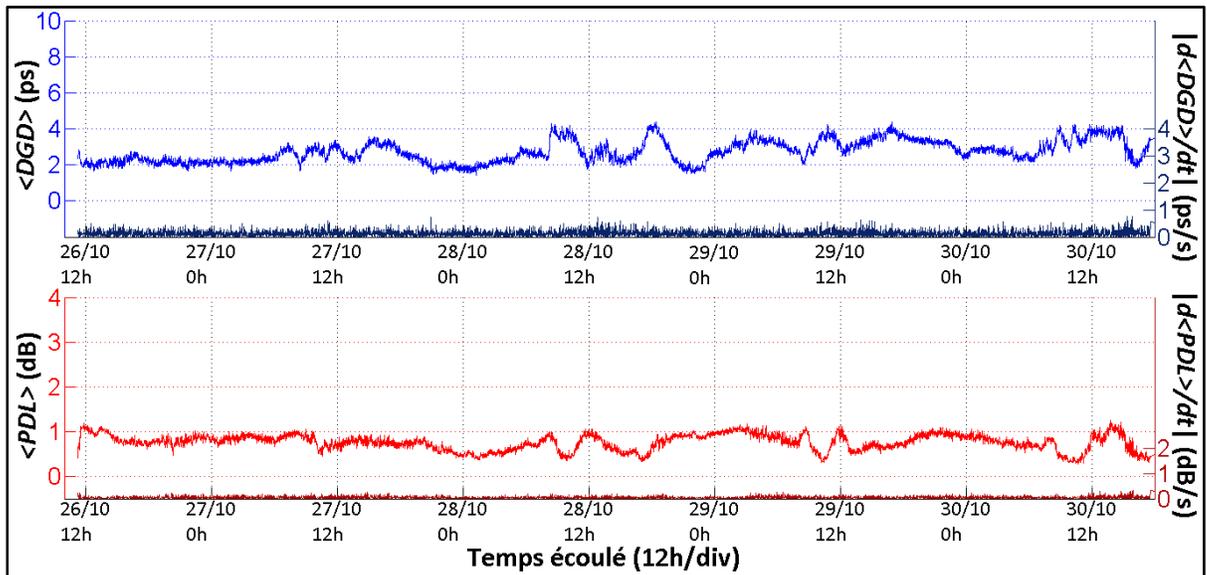


Figure 3.7 Évolution temporelle des valeurs de $\langle DGD \rangle$, de $|d\langle DGD \rangle/dt|$, de $\langle PDL \rangle$ et de $|d\langle PDL \rangle/dt|$ à chaque minute sur le lien optique cohérent du réseau Canarie

La figure 3.8 décrit les courbes d'évolution temporelle de $\langle DGD \rangle$, de $\langle PDL \rangle$, de leurs dérivées temporelles $\left| \frac{d\langle DGD \rangle}{dt} \right|$ et $\left| \frac{d\langle PDL \rangle}{dt} \right|$ pour chaque minute de mesures du 5 au 11 novembre 2012 sur le lien optique du réseau Verizon.

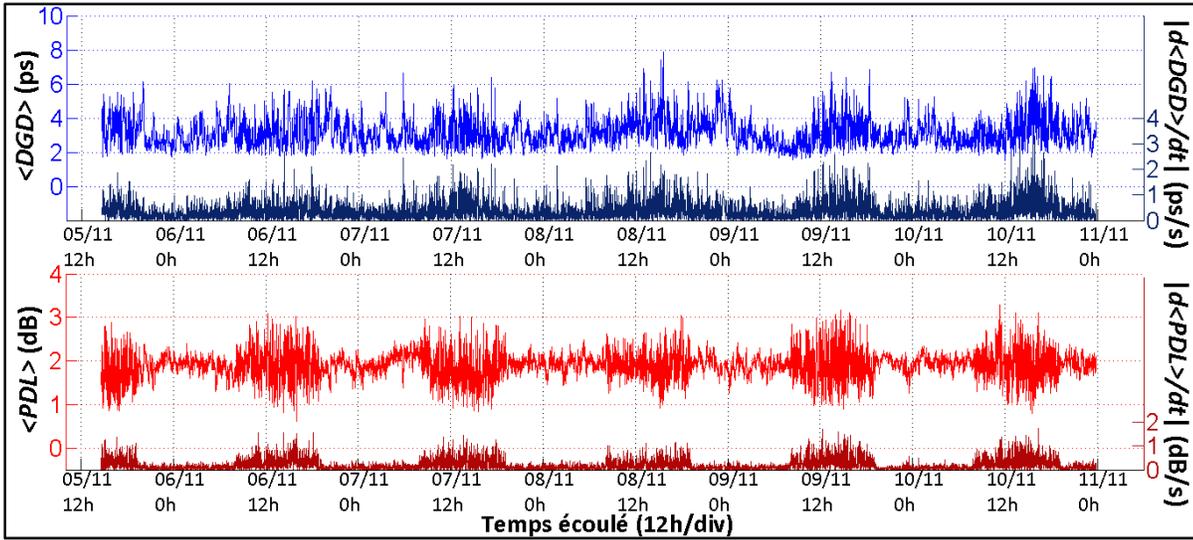


Figure 3.8 Évolution temporelle des valeurs de $\langle DGD \rangle$, de $\langle PDL \rangle$, de $|d\langle DGD \rangle/dt|$ et de $|d\langle PDL \rangle/dt|$ pour chaque minute sur le lien optique cohérent du réseau Verizon

Les grandeurs des axes de la figure 3.7 et de la figure 3.8 sont identiques en vue de la comparaison efficace des mesures réalisées sur le lien optique du réseau Canarie et sur le lien optique du réseau Verizon. Les valeurs de $\langle DGD \rangle$ fluctuent entre 2 et 4 ps et entre 2 et 8 ps sur la liaison enterrée du réseau Canarie et sur la liaison du réseau Verizon respectivement (Voir Figures 3.7 –3.8). Les valeurs de $\langle PDL \rangle$ mesurées se situent entre 0.5 et 1 dB sur la liaison du réseau Canarie et se situent entre 1 et 3 dB sur la liaison du réseau Verizon. Les valeurs maximales de $\left| \frac{d\langle DGD \rangle}{dt} \right|$ et $\left| \frac{d\langle PDL \rangle}{dt} \right|$ sont égales à 0.775 ps et à 0.346 dB pour les mesures réalisées sur la liaison du réseau Canarie et celles-ci sont égales à 3.31 ps et à 1.743 dB sur la liaison du réseau Verizon (Voir Figures 3.7 –3.8). Les valeurs de $\left| \frac{d\langle DGD \rangle}{dt} \right|$ et de $\left| \frac{d\langle PDL \rangle}{dt} \right|$ sont 2.84 fois et 4.24 fois supérieures sur la liaison du réseau Verizon que sur la liaison du réseau Canarie. De plus, deux types de fluctuation caractérisent l'évolution temporelle des variables de $\langle DGD \rangle$, de $\langle PDL \rangle$, de $\left| \frac{d\langle DGD \rangle}{dt} \right|$ et de $\left| \frac{d\langle PDL \rangle}{dt} \right|$ et apparaissent de manière cyclique lors des mesures réalisées sur le lien aérien du réseau Verizon (Voir Figure 3.8). La première période définit une demi-journée de mesures débutant à 8h et prenant fin à 20h. La seconde période de mesures débute à 20h et se termine à 8h le

lendemain. Ces périodes définissent une journée et une nuit de mesures respectivement. La moyenne des valeurs de $\left| \frac{d\langle DGD \rangle}{dt} \right|$ et de $\left| \frac{d\langle PDL \rangle}{dt} \right|$ sont égales à 0.60 ps et à 0.11 dB lors des journées de mesures alors que celles-ci s'élèvent à 0.24 ps et à 0.027 dB lors des nuits de mesure. Les amplitudes des valeurs de $\left| \frac{d\langle DGD \rangle}{dt} \right|$ et de $\left| \frac{d\langle PDL \rangle}{dt} \right|$ sont plus importantes durant la journée de mesures que durant la nuit de mesures. En effet, les moyennes de $\left| \frac{d\langle DGD \rangle}{dt} \right|$ et de $\left| \frac{d\langle PDL \rangle}{dt} \right|$ sont 2.5 fois et 4.1 fois supérieures lors des journées de mesures que celles calculées durant les nuits de mesures. La figure 3.9 décrit les courbes d'évolution temporelle des valeurs de σ_{DGD} et σ_{PDL} , ainsi que des valeurs de $\left| \frac{d\sigma_{DGD}}{dt} \right|$ et de $\left| \frac{d\sigma_{PDL}}{dt} \right|$ sur le lien optique cohérent du réseau CANARIE pour la période de mesure du 26 au 30 octobre 2011.

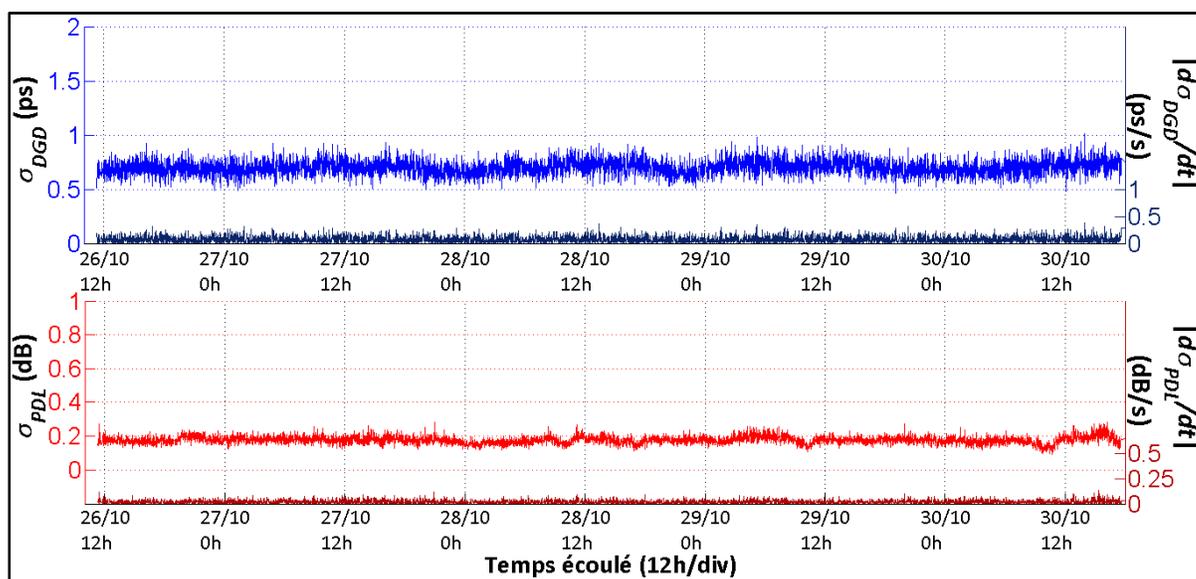


Figure 3.9 Évolution temporelle des valeurs de σ_{DGD} , de σ_{PDL} , de $|d\langle\sigma_{DGD}\rangle/dt|$ et de $|d\langle\sigma_{PDL}\rangle/dt|$ à chaque minute sur le lien optique cohérent du réseau CANARIE

On observe que les valeurs de σ_{DGD} et σ_{PDL} varient peu pour chaque minute de mesures sur la figure 3.9, ce qui démontre que seules des faibles valeurs de $\left| \frac{d\sigma_{DGD}}{dt} \right|$ et $\left| \frac{d\sigma_{PDL}}{dt} \right|$ se produisent sur la liaison enterrée du réseau Canarie. En effet, les moyennes des valeurs de σ_{DGD} et σ_{PDL} calculées pour chaque minute de mesure sont égales à 0.7 ps et à 0.18 dB. Les moyennes des

valeurs de $\left| \frac{d\sigma_{DGD}}{dt} \right|$ et de $\left| \frac{d\sigma_{PDL}}{dt} \right|$ sont égales à 0.07 ps et à 0.02 dB respectivement. La figure 3.10 décrit les courbes d'évolution temporelle des écarts types de DGD et des écarts types de PDL pour chaque minute de mesures du 5 au 11 novembre 2012 sur le lien optique du réseau Verizon.

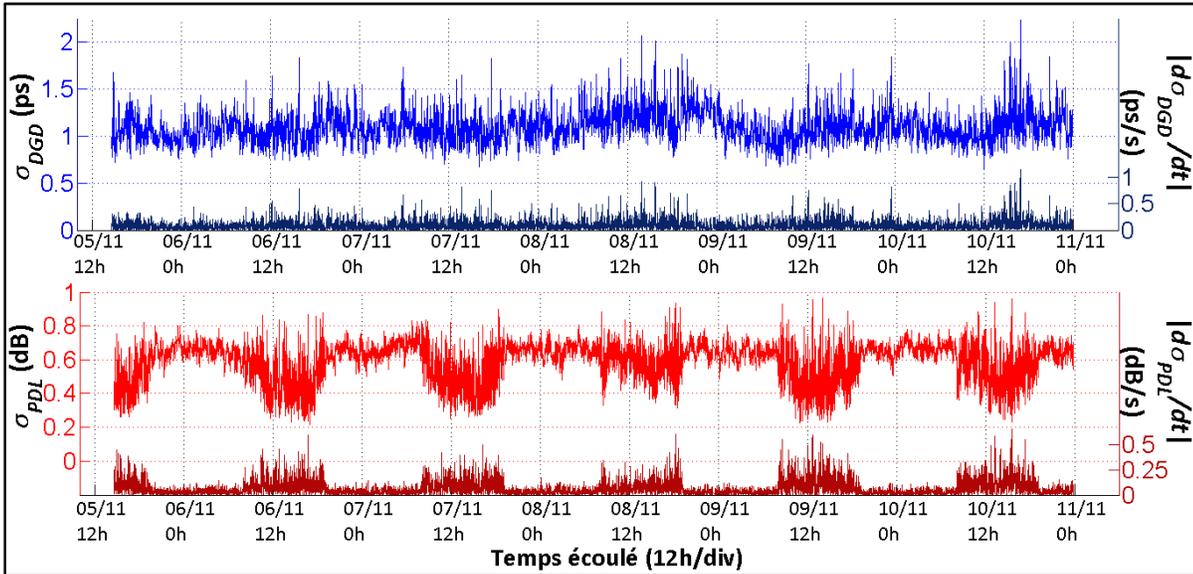


Figure 3.10 Évolution temporelle des valeurs de σ_{DGD} , de σ_{PDL} , de $|d\langle\sigma_{DGD}\rangle/dt|$ et de $|d\langle\sigma_{PDL}\rangle/dt|$ à chaque minute sur le lien optique cohérent du réseau Verizon

Les valeurs de σ_{DGD} et de σ_{PDL} sont supérieures sur la liaison du réseau Verizon que celles mesurées sur la liaison du réseau CANARIE (Voir Figures 3.9 -3.10). En effet, la moyenne des valeurs de σ_{DGD} et de σ_{PDL} calculés pour chaque minute de mesures sur le lien du réseau Verizon sont égales à 1.09 ps et à 0.59 dB respectivement (Voir Figure 3.10). Les moyennes des valeurs de $\left| \frac{d\sigma_{DGD}}{dt} \right|$ et de $\left| \frac{d\sigma_{PDL}}{dt} \right|$ sont égales aussi à 0.1 ps et 0.07 dB. On peut déduire que les fluctuations des valeurs de $\left| \frac{d\sigma_{DGD}}{dt} \right|$ et de $\left| \frac{d\sigma_{PDL}}{dt} \right|$ sont 1.42 fois et 3.5 fois supérieures lors des mesures réalisées sur le lien du réseau Verizon que celles mesurées sur le lien du réseau Canarie. De plus, deux types de fluctuations des valeurs de σ_{DGD} , de σ_{PDL} , $\left| \frac{d\sigma_{DGD}}{dt} \right|$ et de $\left| \frac{d\sigma_{PDL}}{dt} \right|$ sont aussi observés durant les journées et les nuits de mesures (Voir Figure 3.13). Les

moyennes des valeurs de $\left| \frac{d\sigma_{DGD}}{dt} \right|$ et de $\left| \frac{d\sigma_{PDL}}{dt} \right|$ pour une journée de mesures sont égales respectivement à 0.88 ps et à 0.4105 dB alors que celles-ci s'élèvent à 0.66 ps et à 0.178 dB lors d'une nuit de mesures (Voir Figure 3.10). Waddy *et al.* (2001), Kawasaki *et al.* (2007) proposent que les variations des températures et les variations des vitesses de vent sont les phénomènes physiques à l'origine des différentes amplitudes de fluctuation de toutes les valeurs calculées sur le lien aérien du réseau Verizon (Voir Figures 3.9 - 3.10). On peut donc déduire de toutes ces observations que les valeurs de $\left| \frac{d\langle DGD \rangle}{dt} \right|$, de $\left| \frac{d\langle PDL \rangle}{dt} \right|$, de $\left| \frac{d\sigma_{DGD}}{dt} \right|$ et de $\left| \frac{d\sigma_{PDL}}{dt} \right|$ sont quasi stables et constants sur le lien enterrée du réseau CANARIE que celles-ci sont nettement supérieures sur le lien aérien du réseau Verizon en raison du type d'installation et du très grand nombre de valeurs de DGD et de PDL mesurées et enregistrées par le transpondeur cohérent sur la liaison du réseau Verizon.

3.5 Autocorrélation des valeurs de DGD et de PDL

Woodward *et al.* (2014) et Kawasaki *et al.* (2007) proposent de caractériser les fluctuations des valeurs de DGD et de PDL, c'est-à-dire des valeurs de $\left| \frac{d\langle DGD \rangle}{dt} \right|$ et de $\left| \frac{d\langle PDL \rangle}{dt} \right|$ par le calcul de la fonction d'autocorrélation. Le résultat de la fonction d'autocorrélation se présente sous la forme d'un signal corrélé, c'est-à-dire multiplié par lui-même, en vue de connaître la dépendance entre une mesure au temps t et une mesure au temps $t+k$. Waddy *et al.* (2001) ont calculé la fonction d'autocorrélation des valeurs de DGD et de PDL sur une liaison aérienne. Les résultats des travaux de Waddy *et al.* (2001) et de Waddy *et al.* (2005) ont permis de démontrer que plus l'amplitude des fluctuations de $\left| \frac{d\langle DGD \rangle}{dt} \right|$ et de $\left| \frac{d\langle PDL \rangle}{dt} \right|$ sont élevées, plus le temps de corrélation des valeurs de DGD et de PDL mesurées par le transpondeur cohérent est de courte durée. La fonction d'autocorrélation est par conséquent un outil d'analyse efficace pour la caractérisation des valeurs de DGD et de PDL mesurées sur tout type d'installation optique. Celle-ci a donc été appliquée sur toutes les valeurs de DGD et de PDL mesurées avec le transpondeur cohérent sur la liaison du réseau Canarie et sur la liaison du réseau Verizon en vue de connaître et de comparer la durée selon laquelle il n'existe plus de relation temporelle entre la mesure réalisée au temps t et celle réalisée au

temps $t+\tau$ comme présentée dans les travaux de Kawasaki *et al.* (2007). Cette durée est appelée le temps de corrélation τ des valeurs de DGD ou de PDL mesurées.

3.5.1 Méthode de calcul de la fonction d'autocorrélation des valeurs de DGD et de PDL

Ma principale contribution porte sur le développement d'un script Matlab qui comporte la méthode de calcul de la fonction d'autocorrélation appliquée sur les valeurs de DGD et de PDL. 5 jours de mesures ont été choisis pour chacune des liaisons optiques cohérentes. La période de mesures sélectionnée pour le lien du réseau Canarie débute le 26 octobre 2011 jusqu'au 30 octobre 2011 alors que celle-ci commence le 5 novembre 2012 et prend fin le 10 novembre 2012 pour le lien du réseau Verizon. La figure 3.11 décrit la méthode de calcul de la fonction d'autocorrélation et des temps de corrélation appliquée sur les valeurs de DGD et sur les valeurs de PDL pour chacune des périodes de mesures précédemment décrites.

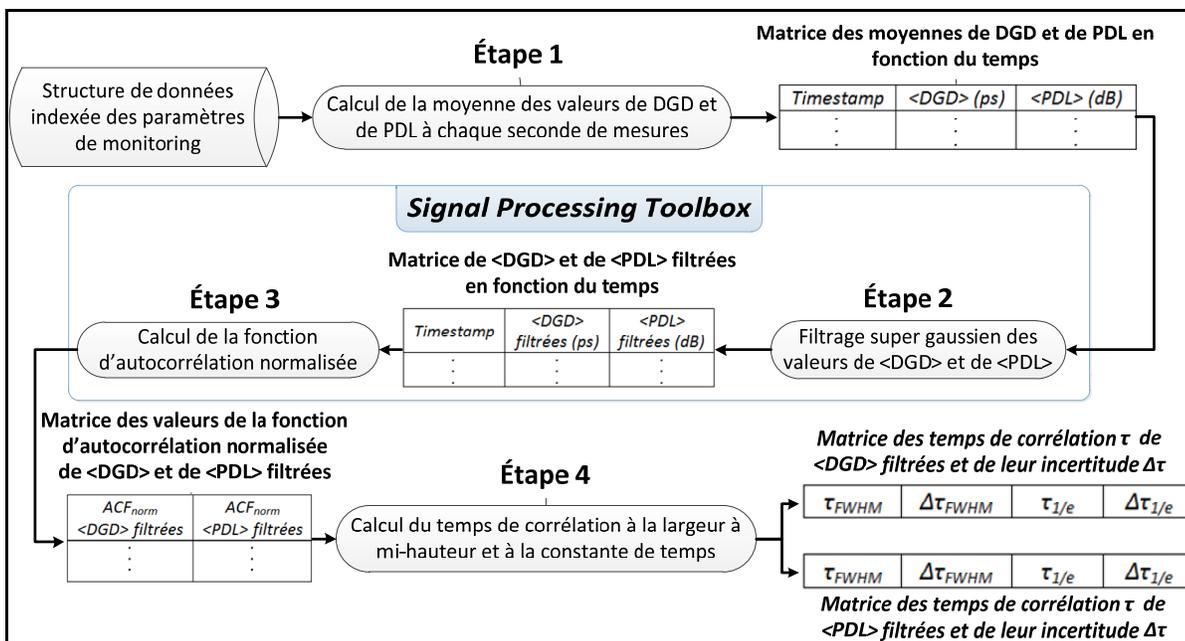


Figure 3.11 Méthode de calcul de la fonction d'autocorrélation et des temps de corrélation τ_{FWHM} et $\tau_{1/e}$ sur les valeurs de DGD et sur les valeurs de PDL

La méthode de calcul du temps de corrélation correspondant aux valeurs de DGD et aux valeurs de PDL mesurées par le transpondeur cohérent comporte quatre étapes. Les sections suivantes décrivent de façon détaillée chacune des étapes de la méthode de calcul de la fonction d'autocorrélation des valeurs de DGD et de PDL présentée par la figure 3.11.

3.5.1.1 Étape 1 : calcul de la moyenne des valeurs de DGD et de PDL pour chaque seconde de mesure

Afin que les résultats des fonctions d'autocorrélation calculées sur la liaison du réseau CANARIE et sur la liaison du réseau Verizon puissent être comparés efficacement, la moyenne des valeurs de DGD et la moyenne des valeurs de PDL, désignées par la variable $\langle DGD \rangle$ et par la variable $\langle PDL \rangle$ respectivement, ont été calculées pour chaque seconde de mesure. Ce premier calcul développé dans le script Matlab est appliqué sur la structure de données indexée telle que présentée par la figure 3.5 (Voir ANNEXE XVIII, p. 165). Le résultat de cette étape se présente sous la forme d'une matrice constituée des valeurs de $\langle DGD \rangle$ et de $\langle PDL \rangle$ en fonction du temps pour chacune des périodes de mesures (Voir Figure 3.11).

3.5.1.2 Étape 2 : filtrage super gaussien des valeurs de $\langle DGD \rangle$ et des valeurs de $\langle PDL \rangle$ pour chaque seconde de mesure

Maurice O'sullivan (2012) a proposé trois méthodes de filtrage pour la réduction du bruit généré par le système de transmission optique. La première méthode de filtrage porte sur le lissage des mesures qui désigne une technique de filtrage temporel. Une moyenne glissante est appliquée sur les valeurs de $\langle DGD \rangle$ et de $\langle PDL \rangle$ pour réaliser cette première méthode de filtrage. La seconde méthode proposée est le filtrage super gaussien du spectre des valeurs de DGD et de PDL. La méthode de Welch définit la troisième méthode de filtrage proposée. Les résultats les plus concluants ont été obtenus par l'application du filtre super gaussien d'ordre 4. La méthode de filtrage sélectionnée définit ainsi le calcul de la convolution du filtre super gaussien d'ordre 4 appliqué sur la transformée de Fourier, c'est-à-dire le spectre, des valeurs de $\langle DGD \rangle$ et de $\langle PDL \rangle$.

La figure 3.12 décrit la séquence de calcul du filtrage super gaussien appliqué sur les moyennes des valeurs de DGD et des valeurs de PDL mesurées avec le transpondeur cohérent sur le lien du réseau CANARIE et sur le lien du réseau Verizon pour chaque seconde de mesure.

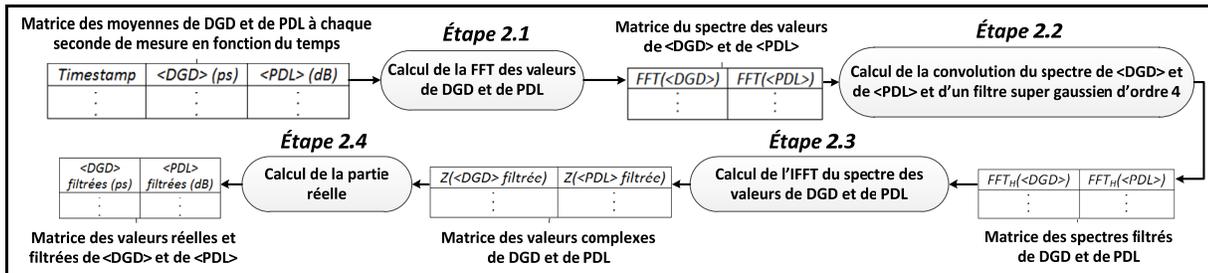


Figure 3.12 Séquence de calcul du filtrage super gaussien de la transformée de Fourier des valeurs de <DGD> et de <PDL>

Des fonctions offertes par le module *Signal Processing Toolbox* ont été configurées pour chacune des étapes 2.1 à 2.4 décrites par la figure 3.12 dans le script Matlab (*Voir ANNEXE XIX, p. 167*).

La transformée de Fourier rapide a tout d'abord été calculée sur les valeurs de <DGD> et de <PDL> lors de l'étape 2.1 par l'application de la fonction *fft* dans Matlab. Soit x_n la variable désignant la valeur de DGD ou de PDL à l'indice n et le nombre total N de valeurs mesurées, alors l'équation (3.4) décrit la formule de transformée de Fourier rapide selon Kreyszig *et al.* (2011, p.529).

$$X(n) = \sum_{n=1}^N x_n \times e^{-j2\pi \frac{(N-1)(n-1)}{N}} = \sum_{n=1}^N x_n \times w^{Nn} \quad \text{avec } 1 \leq n \leq N \quad (3.4)$$

L'équation (3.4) formule la somme de la multiplication d'une série de valeurs numériques x_n par leurs coefficients w^{Nn} respectifs à l'indice n selon l'équation (3.4). Les valeurs $X(n)$ sont

les valeurs du spectre des valeurs de $\langle DGD \rangle$ et de $\langle PDL \rangle$ résultant du calcul de la transformée de Fourier et sont enregistrées dans une matrice complexe de N valeurs lors de l'étape de calcul 2.1 (Voir Figure 3.12).

La méthode de filtrage réalisée dans l'étape 2.2 consiste à sélectionner seulement les fréquences du spectre des valeurs de $\langle DGD \rangle$ et de $\langle PDL \rangle$ pour lesquelles leurs amplitudes sont les plus élevées. Les fréquences harmoniques les plus éloignées de la fréquence fondamentale ont été supprimées lors du filtrage super gaussien appliqué sur le spectre des valeurs de $\langle DGD \rangle$ et de $\langle PDL \rangle$. La figure 3.13 présente la partie imaginaire du spectre des valeurs de $\langle DGD \rangle$ sur le lien du réseau Verizon après l'étape de calcul 2.1 (Voir Figure 3.12).

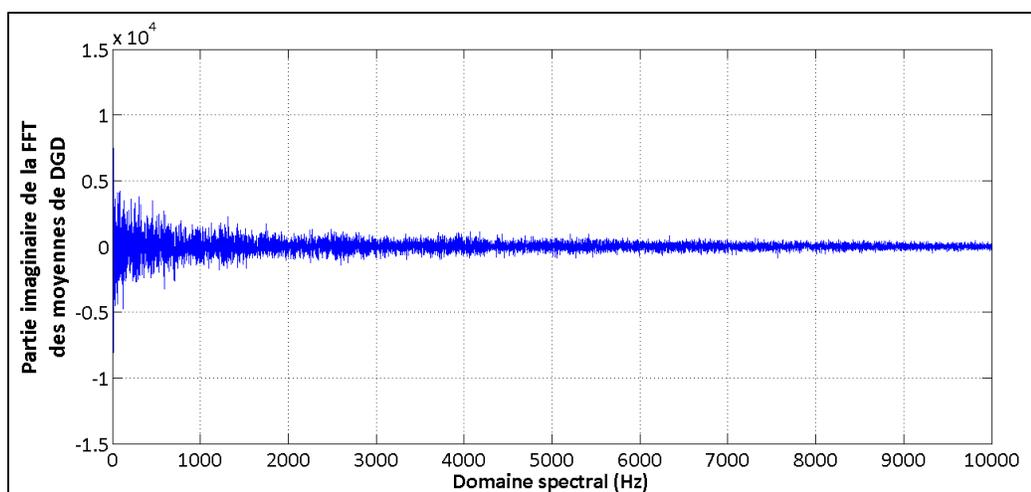


Figure 3.13 Partie imaginaire des valeurs de $\langle DGD \rangle$ sur le lien optique cohérent du réseau Verizon

La figure 3.13 montre que la limite à 500 Hz permet de sélectionner plus de 90 % des fréquences caractéristiques du spectre des valeurs de $\langle DGD \rangle$. La largeur du filtre doit être suffisamment grande pour permettre la sélection de toutes les fréquences jusqu'à 500 Hz. Le filtre super gaussien d'ordre 4 permet aisément de réaliser cette technique de filtrage sur le spectre $X(n)$ des valeurs de $\langle DGD \rangle$ et de $\langle PDL \rangle$ mesurées par le transpondeur cohérent. L'équation (3.5) présente la formule mathématique du filtre super gaussien d'ordre k .

$$H(n) = e^{-\frac{1}{2}\left(\alpha \times \frac{n}{N/2}\right)^k} \text{ avec } \sigma = \frac{N}{2\alpha} \quad (3.5)$$

Tirée de Hui *et al.* (1991)

σ définit la bande passante du filtre super gaussien. Celle-ci définit la plage de sélection des fréquences du spectre $X(n)$ des valeurs de $\langle DGD \rangle$ et de $\langle PDL \rangle$. Afin que le bruit associé aux fréquences harmoniques puisse être quasi supprimé, un filtre super gaussien d'ordre 4 a été choisi. En effet, le filtre super gaussien d'ordre 4 sélectionne toutes les fréquences situées entre la fréquence fondamentale indiquée par la valeur $X(0)$ jusqu'à la fréquence de coupure correspondant à la valeur du spectre $X(\alpha)$. Ce filtre définit donc le moyen de cibler avec précision les basses fréquences du spectre $X(n)$ des valeurs de $\langle DGD \rangle$ et de $\langle PDL \rangle$ mesurées par le transpondeur cohérent. L'équation (3.6) formule l'équation du filtre super gaussien d'ordre 4 déduite de l'équation (3.5).

$$H(n) = e^{-4\left(\alpha \times \frac{n}{N}\right)^4} \quad (3.6)$$

La valeur caractéristique α est égale à la valeur de la bande passante σ du filtre super gaussien d'ordre 4 selon l'équation (3.6). Pour cette raison, il est facile de configurer avec précision la fréquence de coupure du filtre $H(n)$ super gaussien d'ordre 4. Le choix de la valeur de l'écart type σ du filtre super gaussien est une décision importante en vue de la réalisation d'un filtrage de qualité. Plus la valeur α est faible, plus la bande passante du filtre dans le domaine spectrale est étroite (*Voir* Équation 3.5). Ce filtre doit lisser correctement les mesures tout en ne supprimant aucune des fluctuations caractéristiques des valeurs de $\langle DGD \rangle$ et de $\langle PDL \rangle$. La valeur du paramètre α est égale à 500 pour l'application d'un filtrage adapté du spectre des valeurs de $\langle DGD \rangle$ et du spectre des valeurs de $\langle PDL \rangle$ d'après l'équation 3.6. La figure 3.14 décrit le filtre super gaussien $H(n)$ d'ordre 4 appliqué au spectre des valeurs de $\langle DGD \rangle$ et au spectre des valeurs de $\langle PDL \rangle$.

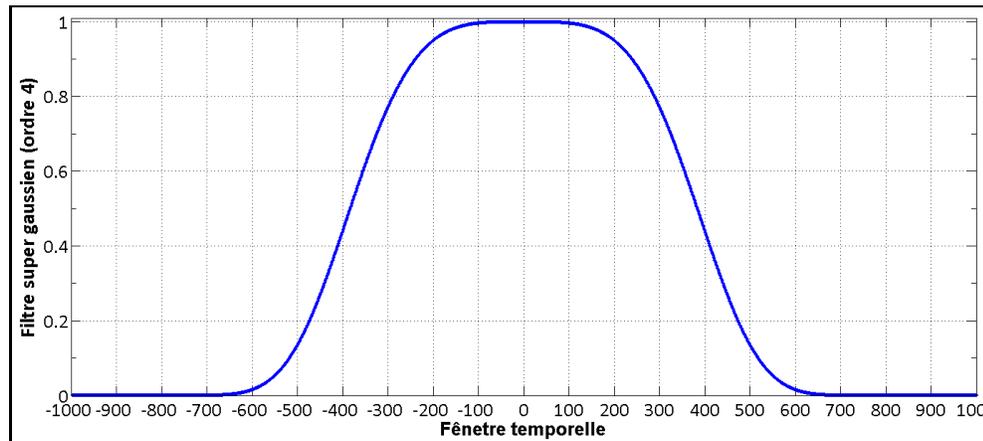


Figure 3.14 Réponse spectrale du filtre super gaussien d'ordre 4

La convolution du filtre super gaussien d'ordre 4 et des valeurs du spectre des valeurs de $\langle DGD \rangle$ ou du spectre des valeurs de $\langle PDL \rangle$ est calculée en vue de la réduction du bruit des mesures selon Smith (1999) et telle qu'indiquée sur la figure 3.15. L'équation (3.7) formule le calcul de convolution $X_h(n)$ des valeurs du spectre $X(n)$ de $\langle DGD \rangle$ et de $\langle PDL \rangle$ et du filtre super gaussien $H(n)$ d'ordre 4.

$$X_h(n) = X(n) * H(n) \quad (3.7)$$

Les valeurs $X_h(n)$ sont les valeurs du spectre filtré de $\langle DGD \rangle$ et de $\langle PDL \rangle$. Le résultat de cette étape de calcul se présente sous la forme d'une matrice constituée des valeurs du spectre filtré $X_h(n)$ de $\langle DGD \rangle$ et de $\langle PDL \rangle$, désignées par la variable $FFT_h(\langle DGD \rangle)$ et par la variable $FFT_h(\langle PDL \rangle)$ respectivement (Voir Figure 3.12).

L'inverse de la transformée de Fourier rapide est ensuite calculé sur les valeurs filtrées du spectre $X_h(n)$ lors de l'étape de calcul 2.3. L'équation (3.8) formule le calcul de l'inverse de la transformée de Fourier rapide sur le spectre $X_h(n)$ des valeurs de $\langle DGD \rangle$ et des valeurs de $\langle PDL \rangle$ filtrées adaptée de Kreyszig *et al.* (2011, p.530).

$$x_h(n) = \left| \frac{1}{N} \sum_{n=1}^N X_h(n) e^{j2\pi \frac{(N-1)(n-1)}{N}} \right| = \left| \frac{1}{N} \sum_{n=1}^N X_h(n) W^{nN} \right| \quad (3.8)$$

Les valeurs $x_h(n)$ sont les valeurs de $\langle DGD \rangle$ filtrées et les valeurs de $\langle PDL \rangle$ filtrées et sont enregistrées dans une matrice complexe lors de l'étape de calcul 2.3 (*Voir* figure 3.12). La dernière étape de calcul consiste à extraire la partie réelle des valeurs de $\langle DGD \rangle$ et de $\langle PDL \rangle$ filtrées et à sauvegarder ces valeurs à présent réelles et filtrées dans le domaine temporel (*Voir* figure 3.12).

3.5.1.3 Étape 3 : calcul de la fonction d'autocorrélation des valeurs de $\langle DGD \rangle$ filtrées et des valeurs de $\langle PDL \rangle$ filtrées

La troisième étape de la méthode de calcul consiste à calculer la fonction d'autocorrélation normalisée en soustrayant la composante continue c'est-à-dire la valeur moyenne des valeurs de $\langle DGD \rangle$ et de $\langle PDL \rangle$ filtrées en vue du calcul d'un temps de corrélation significatif aux variations temporelles des valeurs de $\langle DGD \rangle$ et de $\langle PDL \rangle$ (*Voir* Figure 3.12). La soustraction de la composante continue des valeurs de $\langle DGD \rangle$ et de $\langle PDL \rangle$ filtrées permet de visualiser efficacement les dépendances des valeurs mesurées entre les différents temps de mesure t et $t+k$.

Kawasaki *et al.* (2007) et Woodward *et al.* (2014) appliquent cette fonction d'autocorrélation dans leurs travaux de *monitoring* respectifs. Le module *Signal Processing Toolbox* est une seconde fois utilisé en vue du calcul de la fonction d'autocorrélation normalisée sur les valeurs de $\langle DGD \rangle$ et de $\langle PDL \rangle$ calculées par le script Matlab lors de cette troisième étape de calcul (*Voir* ANNEXE XX, p. 169). La figure 3.15 décrit la séquence de calcul de la fonction d'autocorrélation normalisée appliquée sur les valeurs de $\langle DGD \rangle$ filtrées et sur les valeurs de $\langle PDL \rangle$ filtrées.

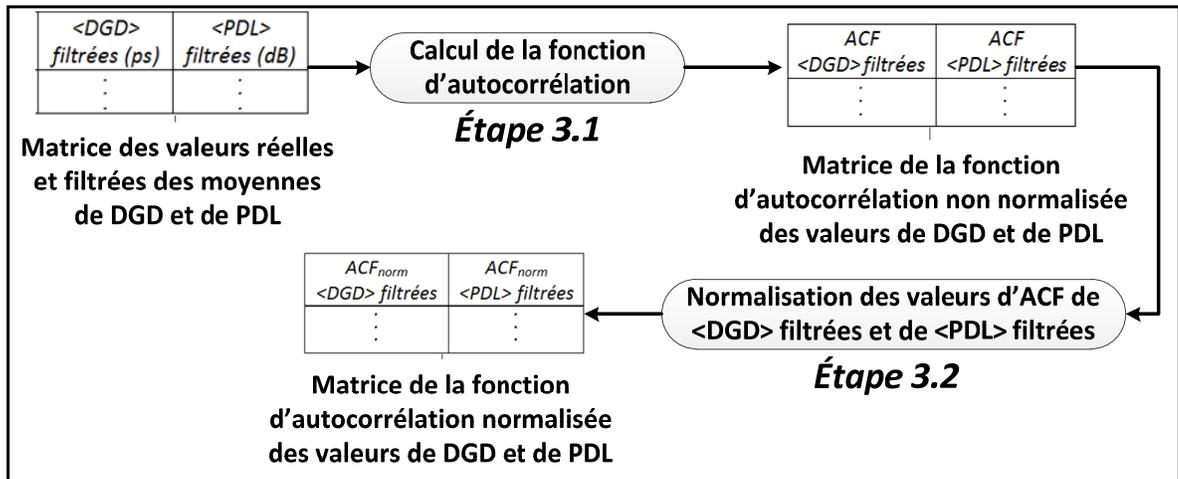


Figure 3.15 Séquence de calcul de la fonction d'autocorrélation normalisée sur les valeurs de $\langle DGD \rangle$ filtrées et sur les valeurs de $\langle PDL \rangle$ filtrées

L'équation (3.9) formule la fonction d'autocorrélation appliquée lors de l'étape de calcul 3.1 sur les valeurs de $\langle DGD \rangle$ filtrées et sur les valeurs de $\langle PDL \rangle$ filtrées adaptée de Moon *et al.* (1999, p.13).

$$\begin{aligned}
 ACF(k) &= Cov(x_n, x_{n-k}) = E[(x(n) - \bar{x}(n))(x(n) - \bar{x}(n - k))] \quad (3.9) \\
 &= \frac{1}{N} \sum_{n=0}^{N-k-1} (x(n) - \bar{x}(n)) (x(n+k) - \bar{x}(n))
 \end{aligned}$$

Cette fonction est aussi appelée fonction d'auto-covariance car celle-ci soustrait la moyenne des valeurs de $\langle DGD \rangle$ et de $\langle PDL \rangle$ lorsque ces valeurs sont multipliées à l'indice n et à l'indice $n+k$. La fonction d'autocorrélation formulée selon l'équation (3.9) désignée par la variable $ACF(k)$ présente deux limites. La valeur maximale de la fonction d'autocorrélation est égale à la variance σ_x^2 des valeurs de $\langle DGD \rangle$ filtrées ou des valeurs de $\langle PDL \rangle$ filtrées lorsque la valeur k est égale à 0. Lorsque k tend vers l'infini, alors le résultat de la fonction d'autocorrélation tend vers 0. Afin que les fonctions d'autocorrélation $ACF(k)$ puissent être comparées et superposées sur les mesures réalisées sur la liaison du réseau Canarie et sur la liaison du réseau Verizon, la fonction d'autocorrélation a été normalisée par la variance σ_x^2 (Voir Figure 3.15). La fonction d'autocorrélation normalisée est représentée par la variable ACF_{norm} et est formulée par l'équation (3.10) adaptée de Kawasaki *et al.* (2007).

$$ACF_{norm}(k) = \frac{Cov(x_n, x_{n-k})}{\sigma_x^2} = \frac{\sum_{n=0}^{N-k-1} (x(n) - \bar{x})(x(n+k) - \bar{x})}{\sum_{n=0}^N (x(n) - \bar{x})^2} \quad (3.10)$$

La valeur de la fonction d'autocorrélation normalisée ACF_{norm} est égale à 1 lorsque k est égal à 0. Cette fonction tend vers 0 lorsque k tend vers l'infini. Le résultat de cette étape de calcul 3.2 se présente sous la forme d'une matrice constituée des valeurs de la fonction d'autocorrélation normalisée ACF_{norm} de $\langle DGD \rangle$ filtrées et des valeurs de la fonction d'autocorrélation normalisée ACF_{norm} de $\langle PDL \rangle$ filtrées (Voir Figure 3.15).

3.5.1.4 Étape 4 : calcul du temps de corrélation des valeurs de $\langle DGD \rangle$ et de $\langle PDL \rangle$ filtrées

Deux seuils ont été définis pour le calcul du temps de corrélation appliqué sur les valeurs de la fonction d'autocorrélation normalisée ACF_{norm} des valeurs $\langle DGD \rangle$ filtrées et de $\langle PDL \rangle$ filtrées (Voir Figure 3.11). Le premier seuil de la fonction ACF_{norm} est égal à la largeur à mi-hauteur, ce qui correspond à la moitié de la valeur maximale de la fonction d'autocorrélation normalisée ACF_{norm} . Le second seuil définit la constante de temps $1/e$ qui est égale à 36 % de la valeur maximale de la fonction d'autocorrélation normalisée ACF_{norm} . Sachant que la valeur maximale de la fonction ACF_{norm} est égale à 1, alors le temps de corrélation estimé défini à la largeur à mi-hauteur $\hat{\tau}_{HWHM}$ et celui défini à la constante de temps $\hat{\tau}_{1/e}$ sont égaux à 0.5 et à 0.37 respectivement telles que définies par Woodward *et al.* (2014) et par Kawasaki *et al.* (2007). Les valeurs d'incertitudes sont par la suite calculées. Celles-ci définissent le rapport entre le temps de corrélation estimé $\hat{\tau}$ et la durée totale de la période de mesures. L'équation (3.11) formule le calcul de l'incertitude relative Δ_τ relatif au temps de corrélation estimé $\hat{\tau}$ sur la durée totale de la mesure T .

$$\Delta_\tau = \frac{\hat{\tau}}{T} \quad (3.11)$$

La valeur du temps de corrélation τ formulé selon l'équation (3.12) est ainsi encadrée par la valeur du temps de corrélation estimé $\hat{\tau}$ et par les valeurs d'incertitude Δ_τ .

$$\hat{\tau} - \Delta_{\tau} \times \hat{\tau} \leq \tau \leq \hat{\tau} + \Delta_{\tau} \times \hat{\tau} \quad (3.12)$$

Le calcul des temps de corrélation $\hat{\tau}_{HWHM}$ et $\hat{\tau}_{1/e}$ ainsi que de leur incertitude Δ_{τ} sur les valeurs de $\langle DGD \rangle$ filtrées et de $\langle PDL \rangle$ filtrées est la dernière étape de calcul développé dans le script Matlab (*Voir* ANNEXE XXI, p. 171). Les valeurs des temps de corrélation $\hat{\tau}_{HWHM}$ et $\hat{\tau}_{1/e}$ et leur valeur d'incertitude ERR_{τ} respective sont par la suite enregistrés dans deux matrices distinctes constituées de temps de corrélations $\hat{\tau}_{HWHM}$ et $\hat{\tau}_{1/e}$ de la fonction ACF_{norm} des valeurs de $\langle DGD \rangle$ filtrées et des valeurs de $\langle PDL \rangle$ filtrées (*Voir* figure 3.12).

3.5.2 Analyse des temps de corrélation des valeurs de DGD et de PDL

Un script Matlab a été développé pour optimiser l'affichage et la superposition des fonctions d'autocorrélation normalisées (*Voir* ANNEXE XXII, p. 173). La figure 3.16 décrit les résultats de la fonction d'autocorrélation ACF_{norm} sur les valeurs de $\langle DGD \rangle$ filtrées lors des mesures réalisées sur le lien optique du réseau CANARIE et sur le lien optique du réseau Verizon pour une période de 5 jours de mesures réalisées par le transpondeur cohérent.

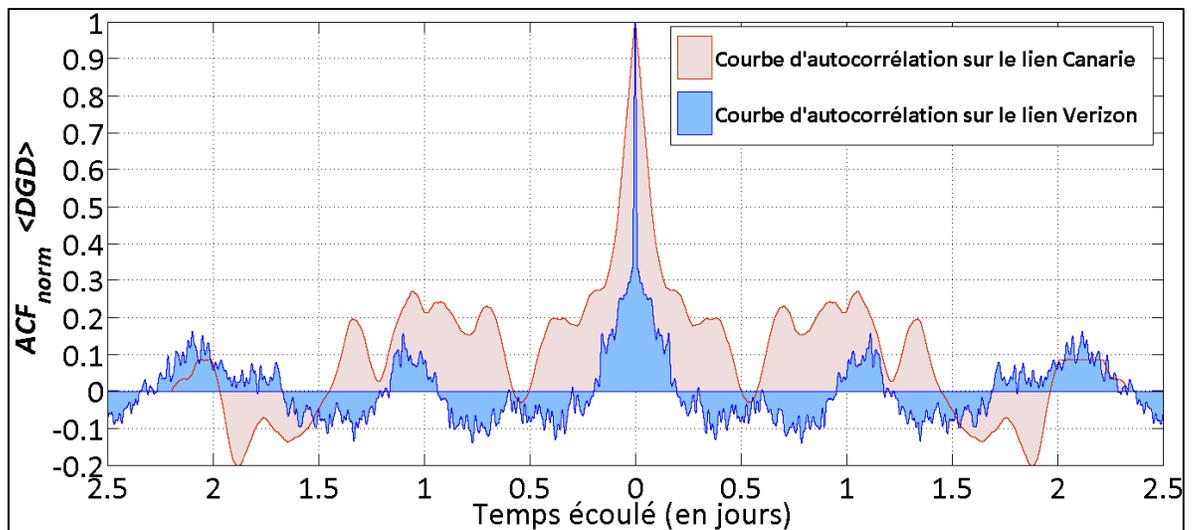


Figure 3.16 Fonctions d'autocorrélation ACF_{norm} des valeurs de $\langle DGD \rangle$ filtrées sur la liaison du réseau Canarie et sur la liaison du réseau Verizon durant 5 jours

Les valeurs des temps de corrélation $\hat{\tau}_{HWHM}$ et $\hat{\tau}_{1/e}$ des valeurs de $\langle DGD \rangle$ filtrées sont égales à 221 min et à 237 min respectivement lors des mesures réalisées sur le lien optique du réseau CANARIE alors que les temps de corrélation $\hat{\tau}_{HWHM}$ et $\hat{\tau}_{1/e}$ sont égales à 22 min et à 24 min respectivement lors de l'expérience de *monitoring* menée sur le lien du réseau Verizon pour une période de 5 jours. Les valeurs de temps de corrélation $\hat{\tau}_{HWHM}$ et $\hat{\tau}_{1/e}$ sont près de 10 fois plus court sur le lien optique du réseau Verizon que sur le lien optique du réseau Canarie (*Voir* Figure 3.16). Des résultats quasi identiques sont obtenus par le calcul de la fonction d'autocorrélation normalisée ACF_{norm} sur les valeurs de $\langle PDL \rangle$ filtrées durant 5 jours de mesures réalisées sur la liaison du réseau Canarie et sur la liaison dur réseau Verizon (*Voir* ANNEXE XXIII, p. 177). La relation temporelle entre deux mesures de DGD ou de PDL ne dure donc que quelques minutes sur la liaison du réseau Verizon alors que celle-ci dure quelques heures sur la liaison du réseau Canarie. Le script Matlab qui intègre la méthode de calcul de la fonction d'autocorrélation normalisée ACF_{norm} (*Voir* section 3.5.1) est par la suite appliqué sur chaque journée de mesures pour analyser l'évolution temporelle des temps de corrélation $\hat{\tau}_{HWHM}$ et $\hat{\tau}_{1/e}$ ainsi que de leur incertitude relative Δ_{HWHM} et $\Delta_{1/e}$. Le tableau 3.2 présente la valeur du temps de corrélation τ_{HWHM} et la valeur du temps de corrélation $\tau_{1/e}$ après le calcul des fonctions d'autocorrélation normalisées sur les valeurs de $\langle DGD \rangle$ filtrées et de $\langle PDL \rangle$ filtrées.

Tableau 3.2 Temps de corrélation τ_{HWHM} et $\tau_{1/e}$ et de leur incertitude Δ_{HWHM} et $\Delta_{1/e}$ des valeurs de $\langle DGD \rangle$ filtrées et des valeurs de $\langle PDL \rangle$ filtrées sur le lien du réseau CANARIE

Jour	ACF_{DGD}		ACF_{PDL}	
	$\tau_{HWHM} \pm \Delta_{HWHM}$	$\tau_{1/e} \pm \Delta_{1/e}$	$\tau_{HWHM} \pm \Delta_{HWHM}$	$\tau_{1/e} \pm \Delta_{1/e}$
26/10/2011	132 min \pm 29 min	200 min \pm 66 min	133 min \pm 14 min	201 min \pm 26 min
27/10/2011	162 min \pm 18 min	268 min \pm 50 min	141 min \pm 16 min	176 min \pm 23 min
28/10/2011	136 min \pm 13 min	165 min \pm 19 min	179 min \pm 12 min	231 min \pm 28 min
29/10/2011	101 min \pm 7 min	137 min \pm 13 min	112 min \pm 22 min	158 min \pm 37 min
30/10/2011	125 min \pm 11 min	170 min \pm 20 min	110 min \pm 28 min	137 min \pm 43 min

La valeur moyenne du temps de corrélation $\hat{\tau}_{HWHM}$ et la valeur moyenne du temps de corrélation $\hat{\tau}_{1/e}$ sont égales à 131 min et à 188 min respectivement pour les valeurs de $\langle DGD \rangle$ filtrées (Voir Figure 3.16) et sont égales à 135 min et à 180 min pour les valeurs de $\langle PDL \rangle$ filtrées (Voir Tableau 3.2). La valeur du temps de corrélation $\hat{\tau}_{HWHM}$ fluctue entre 101 min et 179 min et la valeur du temps de corrélation $\hat{\tau}_{1/e}$ varie entre 137 min et 268 min (Voir Tableau 3.3). L'erreur relative ERR_{τ} aux temps de corrélation $\hat{\tau}_{HWHM}$ et $\hat{\tau}_{1/e}$ est rarement supérieure à 60 min. La moyenne des valeurs d'incertitude relative Δ_{HWHM} et $\Delta_{1/e}$ est égale à 15min37s. L'écart relatif entre la moyenne du temps de corrélation $\hat{\tau}_{HWHM}$ et de son incertitude Δ_{HWHM} est égal à 11,9 % pour les valeurs de $\langle DGD \rangle$ et est égale à 13,7 % pour les valeurs de $\langle PDL \rangle$, ce qui signifie qu'une caractérisation précise du temps de corrélation des valeurs de $\langle DGD \rangle$ ou des valeurs de $\langle PDL \rangle$ sur ce lien est difficile à estimer. Le tableau 3.3 présente les valeurs des temps de corrélation $\hat{\tau}_{HWHM}$ et $\hat{\tau}_{1/e}$ ainsi que de leur valeur d'incertitude relative Δ_{HWHM} et $\Delta_{1/e}$ calculées sur les valeurs de $\langle DGD \rangle$ filtrées et sur les valeurs de $\langle PDL \rangle$ filtrées sur le lien du réseau Verizon.

Tableau 3.3 Temps de corrélation τ_{HWHM} et $\tau_{1/e}$ et de leur incertitude Δ_{HWHM} et $\Delta_{1/e}$ des valeurs de $\langle DGD \rangle$ filtrées et des valeurs de $\langle PDL \rangle$ filtrées sur le lien du réseau Verizon

Jour	ACF_{DGD}		ACF_{PDL}	
	$\tau_{HWHM} \pm \Delta_{HWHM}$	$\tau_{1/e} \pm \Delta_{1/e}$	$\tau_{HWHM} \pm \Delta_{HWHM}$	$\tau_{1/e} \pm \Delta_{1/e}$
05/11/2012	7 min \pm 5 s	10 min \pm 11 s	2 min \pm 0.5 s	3 min \pm 0.825 s
06/11/2012	7 min 31 s \pm 2 s	9 min 33 s \pm 3 s	3 min 21 s \pm 0.468 s	3 min 48 s \pm 0.602 s
07/11/2012	5 min 55 s \pm 1 s	8 min 38 s \pm 3 s	3 min 44 s \pm 0.613 s	5 min 19 s \pm 1 s
08/11/2012	7 min 29 s \pm 2 s	11 min 26 s \pm 5 s	3 min 41 s \pm 0.581s	4 min 49 s \pm 0.967s
09/11/2012	7 min 27 s \pm 2 s	19 min 03 s \pm 15 s	3 min 41 s \pm 0.581 s	4 min 23 s \pm 0.801 s
10/11/2012	6 min 27 s \pm 1 s	14 min 37 s \pm 8 s	3 min 42 s \pm 0.552 s	4 min 20 s \pm 0.786 s

La moyenne des valeurs du temps de corrélation $\hat{\tau}_{HWHM}$ et la moyenne des valeurs du temps de corrélation $\hat{\tau}_{1/e}$ sont égales à 7 min et à 12 min respectivement pour la courbe d'autocorrélation des valeurs de $\langle DGD \rangle$ filtrées et sont égales à 3 min 25 s et à 4 min 25 s pour la courbe d'autocorrélation des valeurs de $\langle PDL \rangle$ filtrées (Voir Tableau 3.3). Les temps de corrélation τ_{HWHM} et $\tau_{1/e}$ déterminés pour une ou plusieurs journées de mesure ont donc une durée très courte (Voir Figure 3.16). Les valeurs d'incertitude relative Δ_{HWHM} et $\Delta_{1/e}$ fluctuent entre 468 ms et 5 s (Voir Tableau 3.2). Les écarts relatifs entre les moyennes des temps de corrélation $\hat{\tau}_{HWHM}$ et $\hat{\tau}_{1/e}$ et de leur incertitude respective sont égaux à 0,52 % et à 1,33 % sur les valeurs de $\langle DGD \rangle$ filtrées et sur les valeurs de $\langle PDL \rangle$ filtrées. Ces résultats montrent que les valeurs des temps de corrélation sont caractérisées très précisément sur la liaison du réseau Verizon en raison de la très courte durée de dépendance entre une mesure de $\langle DGD \rangle$ et de $\langle PDL \rangle$ au temps t et au temps $t+k$. Il en résulte de ces nombreuses observations que le temps de corrélation dépend du type d'installation de la liaison optique. Les temps de corrélation calculés provenant des mesures réalisées sur la liaison presque aérienne du réseau Verizon sont caractérisés par des durées de 10 à 20 fois plus courte que les temps de corrélation calculées sur la liaison presque enterrée du réseau CANARIE.

3.6 Étude de la fonction de densité des valeurs de DGD mesurées sur le lien du réseau Verizon

Peterson (2004, p.38), Reimer (2007, p.9) et Woodward *et al* (2014) montrent que la fonction de densité de probabilité des valeurs de DGD suit une fonction de densité de Maxwell. Cette étude a pour objectif de valider l'ajustement des valeurs de DGD par la fonction de densité de Maxwell sur le lien optique cohérent du réseau Verizon.

3.6.1 Méthode de calcul et d'ajustement de la fonction de densité de probabilité des valeurs de DGD

Ma principale contribution dans cette étude porte sur le développement et sur l'intégration d'une méthode de calcul et d'ajustement de l'histogramme normalisé des valeurs de DGD

mesurées du 5 au 11 novembre 2012 par la fonction de densité de Maxwell. Près de 117 millions de valeurs de DGD ont été mesurées par le transpondeur cohérent sur la liaison optique cohérente du réseau Verizon durant cette période. La figure 3.17 décrit la méthode de calcul et d'ajustement de l'histogramme normalisée des valeurs de DGD appliquée sur la structure de données indexée (Voir Figure 3.5).

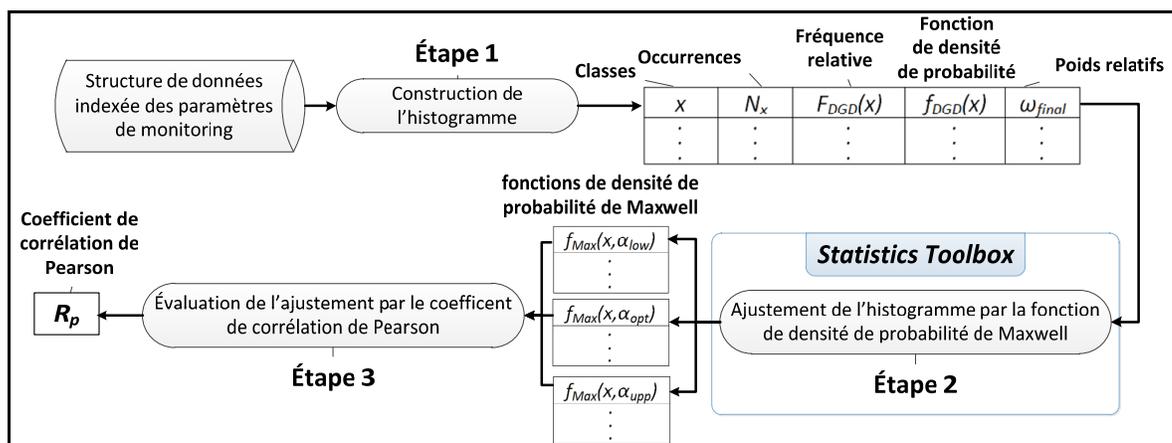


Figure 3.17 Méthode de calcul et d'ajustement de l'histogramme normalisé des valeurs de DGD mesurées sur le lien du réseau Verizon

La méthode de calcul et d'ajustement de l'histogramme normalisé des valeurs de DGD par la fonction de densité de Maxwell comportent trois étapes décrites par la figure 3.17 et a été développée dans deux scripts Matlab. L'histogramme normalisé des valeurs de DGD mesurées est tout d'abord calculé lors de la première étape de calcul (Voir Figure 3.17) par l'application d'un premier script Matlab (Voir ANNEXE XXIV, p. 179).

Les ajustements de l'histogramme des valeurs de DGD par la fonction de densité de Maxwell sont calculés lors de la seconde étape de calcul. Les fonctions de densité de Maxwell obtenues ont été évaluées par le calcul du coefficient de corrélation de Pearson lors de la troisième étape de calcul (Voir Figure 3.17). La méthode de calcul et d'évaluation de l'ajustement de l'histogramme des valeurs de DGD par la fonction de densité de Maxwell est développée dans un second script Matlab (Voir ANNEXE XXV, p. 183).

3.6.1.1 Étape 1 : calcul de l'histogramme normalisé des valeurs de DGD

Les valeurs d'occurrences N_x c'est-à-dire le nombre de répétitions des valeurs de DGD comprises entre deux classes x_1 et x_2 de l'histogramme sont tout d'abord calculées. La résolution Δx , c'est-à-dire la taille de chaque classe x de l'histogramme des valeurs de DGD est égale à 0.01 ps. Les valeurs de la fonction de probabilité désignée par la variable $F_{DGD}(x)$ sont par la suite calculées. Ces valeurs indiquent les fréquences relatives aux valeurs d'occurrence N_x et représentent les valeurs de probabilité estimée \hat{p}_x des valeurs de DGD mesurées. Navidi (2010) montre que la valeur d'une probabilité \hat{p}_x est calculée par la division du nombre d'occurrence N_x sur le nombre total de valeurs N_{TOT} des valeurs de DGD mesurées. L'équation (3.13) formule le calcul de la fonction de probabilité $F_{DGD}(x)$ des valeurs de DGD mesurées adaptée de Navidi (2010, p.27).

$$F_{DGD}(x) = \hat{p}(DGD = x) = \frac{N_x}{N_{TOT}} \quad (3.13)$$

L'histogramme constitué des valeurs de la fonction $F_{DGD}(x)$ est ensuite normalisé par le calcul de l'aire pour l'obtention de la fonction de densité de probabilité $f_{DGD}(x)$ des valeurs de DGD mesurées par le transpondeur cohérent. Kreyszig *et al.* (2011, p.828) propose le calcul de l'aire de l'histogramme des valeurs de DGD mesurées par l'application de la méthode des trapèzes (*Voir ANNEXE XXVI*, p. 187). Le calcul de normalisation de l'histogramme consiste à multiplier la fonction de probabilité $F_{DGD}(x)$ par l'inverse de l'aire précédemment calculée. L'équation (3.14) décrit le calcul de l'histogramme normalisé de la fonction de densité de probabilité $f_{DGD}(x)$ adaptée de Navidi (2010, p.27).

$$f_{DGD}(x) = F_{DGD}(x) \times 1/Aire \cong \frac{dF_{DGD}(x)}{dx} \quad (3.14)$$

Les valeurs de la fonction de densité de probabilité $f_{DGD}(x)$ désignent les valeurs de l'histogramme normalisé des valeurs de DGD mesurées par le transpondeur cohérent. L'équation (3.15) formule les propriétés relatives de la fonction de densité de probabilité

$f_{DGD}(x)$ des valeurs de DGD mesurées sur le lien du réseau Verizon adaptée de Navidi (2010, p.103).

$$f_{DGD}(x) \geq 0 \quad \text{et} \quad \int_{x_0}^{x_n} f_{DGD}(x) dx = 1 \quad (3.15)$$

Dans la mesure où chaque valeur de l'histogramme normalisé possède une valeur d'incertitude propre d'après Navidi (2010, p.209), la valeur de poids de chaque classe x de l'histogramme est par la suite calculée pour accroître la précision de l'ajustement de la fonction de densité de probabilité $f_{DGD}(x)$. La valeur de l'incertitude σ_{px} porte sur le calcul de la marge d'erreur de chaque valeur de l'histogramme $F_{ARR}(x)$ des valeurs de DGD mesurées. La marge d'erreur de chaque valeur de l'histogramme peut être approchée par une loi statistique binomiale selon Navidi (2010, p.210). L'équation (3.16) formule le calcul des valeurs d'incertitude σ_{px} d'un histogramme normalisé tirée de Navidi (2010, p.210).

$$\sigma_{px} = \sqrt{\frac{\hat{p}_x \times (1 - \hat{p}_x)}{n}} \quad (3.16)$$

n indique le nombre de classe x de l'histogramme et \hat{p}_x est égale à la valeur de la probabilité de la fonction de probabilité $F_{DGD}(x)$ des valeurs de DGD mesurées. L'équation (3.17) formule le calcul des valeurs de probabilité P_x en fonction de la valeur de la probabilité estimée \hat{p}_x et de la valeur d'incertitude σ_{px} .

$$P_x = F_{DGD}(x) \pm \sigma_{px} \Leftrightarrow P_x = \hat{p}_x \pm \sigma_{px} \quad (3.17)$$

Wolberg (2005, p.49) montre que les poids sont inversement proportionnels aux valeurs d'incertitude σ_{px}^2 lorsque ceux-ci sont calculés au moyen d'une échelle linéaire. Plus la valeur de la probabilité estimée \hat{p}_x d'une classe de DGD est faible, plus la valeur du poids ω_{px} de

cette même classe est grande. L'équation (3.18) formule le calcul des valeurs de poids ω_{px} égales à l'inverse des valeurs d'incertitudes $\sigma_{p_x}^2$ tirée de Wolberg (2005, p.49).

$$\omega_{px} = \frac{1}{\sigma_{p_x}^2} \quad (3.18)$$

Le résultat du premier script Matlab pour le calcul de l'histogramme normalisé des valeurs de DGD se présente sous la forme d'une matrice constituée des valeurs de classe x , des valeurs d'occurrences N_x , des valeurs de la fonction de probabilité $F_{DGD}(x)$, des valeurs de la fonction de densité $f_{DGD}(x)$ des valeurs de DGD mesurées et des valeurs de poids ω_{px} (Voir Figure 3.17).

3.6.1.2 Étape 2 : ajustement de l'histogramme des valeurs de DGD

Cette étape a pour objectif l'ajustement de la fonction de densité de probabilité $f_{DGD}(x)$ des valeurs de DGD mesurées par la fonction de densité de Maxwell formulée selon l'équation (3.19).

$$f_{Max}(x, \alpha) = \frac{1}{\alpha^3} \sqrt{\frac{2}{\pi}} x^2 e^{-\left(\frac{x^2}{2\alpha^2}\right)} \text{ avec } x \geq 0 \text{ et } \alpha \geq 0 \quad (3.19)$$

Tirée de Walck (2007)

La valeur α est le paramètre caractéristique de la fonction de densité de Maxwell $f_{Max}(x, \alpha)$. L'histogramme normalisé $f_{DGD}(x)$ des valeurs de DGD mesurées est ajusté par la fonction de densité de Maxwell en configurant des fonctions offertes par le module *Statistics Toolbox* dans le second script Matlab dédié à l'application de la méthode de calcul et d'évaluation de l'ajustement.

La fonction *nlinfit* est tout d'abord configurée et appliquée sur la fonction de densité $f_{DGD}(x)$ des valeurs de DGD mesurées pour le calcul du paramètre optimisé α_{opt} de la fonction de

densité de Maxwell $f_{Max}(x, \alpha_{opt})$. Cette fonction minimise la méthode pondérée des moindres carrés $S_w(\alpha)$ par l'application de l'algorithme Levenberg-Marquardt (*Voir ANNEXE XXVII, p. 189*). La méthode pondérée des moindres carrés $S_w(\alpha)$ calcule les écarts au carré, c'est-à-dire les résidus $r_i^2(\alpha)$, entre les valeurs de la fonction de densité $f_{DGD}(x)$ et les valeurs de la fonction de densité de Maxwell $f_{Max}(x, \alpha)$. L'équation (3.20) formule la minimisation de la méthode pondérée des moindres carrés $S_w(\alpha_{opt})$ appliquée sur la fonction de densité de probabilité $f_{DGD}(x)$ des valeurs de DGD et sur la fonction de distribution de Maxwell $f_{Max}(x, \alpha)$ pour le calcul du paramètre optimisé α_{opt} d'après Walpole *et al.* (2007).

$$S_w(\alpha) = \min_{\alpha > 0} \left(\sum_{i=1}^N w_i (f_{DGD}(x_i) - f_M(x_i, \alpha))^2 \right) = \min_{\alpha > 0} \left(\sum_{i=1}^N w_i \times r_i^2(\alpha) \right) \quad (3.20)$$

L'approximation de l'erreur standard du paramètre optimisé α_{opt} est la méthode de calcul de l'incertitude Δ implémentée par la fonction *nlparci*. Le résultat de cette fonction se présente sous la forme de deux paramètres, α_{low} et α_{upp} . Ces paramètres désignent la limite inférieure et la limite supérieure à un intervalle d'incertitude de 95% (*Voir ANNEXE XXVIII, p. 193*). La valeur d'incertitude Δ des paramètres α_{low} et α_{upp} est calculée par la multiplication de l'erreur standard du paramètre $SE_{\sigma_{opt}}$ et du coefficient τ de la table de la loi de *Student* à un intervalle de confiance θ (*Voir ANNEXE XXIX, p. 195*). L'équation (3.21) formalise le calcul de l'incertitude Δ du paramètre optimisée α_{opt} de la fonction de densité de Maxwell $f_{Max}(x, \sigma_{opt})$ d'après Walpole *et al.* (2007).

$$\Delta = \pm \tau_{1-\theta/2}^{n-1} \times SE_{\alpha_{opt}} \quad (3.21)$$

L'équation (3.22) formule les limites inférieures et supérieures du paramètre optimisé σ_{opt} de la fonction de densité de Maxwell à un intervalle de confiance θ selon Walpole *et al.* (2007).

$$\alpha_{opt} - \tau_{1-\theta/2}^{n-1} \times SE_{\alpha_{opt}} \leq \alpha_{opt} \leq \alpha_{opt} + \tau_{1-\theta/2}^{n-1} \times SE_{\alpha_{opt}} \quad (3.22)$$

Les paramètres α_{low} et α_{upp} qui désignent les limites à un intervalle de confiance de 95% ont été calculés selon l'équation (3.23) au moyen du coefficient τ de la table de la loi de *Student*.

$$\begin{aligned} \alpha_{opt} - 1,96 \times SE_{\alpha_{opt}} &\leq \alpha_{opt} \leq \alpha_{opt} + 1,96 \times SE_{\alpha_{opt}} \\ \alpha_{low} &\leq \alpha_{opt} \leq \alpha_{upp} \end{aligned} \quad (3.23)$$

Les fonctions de densité de Maxwell $f_{Max}(x, \alpha_{low})$, $f_{Max}(x, \alpha_{opt})$, $f_{Max}(x, \alpha_{upp})$ sont par la suite calculées par l'application de la fonction *nlpredci*.

3.6.1.3 Étape 3 : évaluation de l'ajustement par le coefficient de corrélation de Pearson

Walpole *et al.* (2005, p.432) propose le calcul du coefficient de corrélation de Pearson pour évaluer avec qualité l'ajustement d'un histogramme normalisé par tout type de fonction de densité de probabilité. Pour cette raison, l'ajustement de la fonction de densité $f_{DGD}(x)$ des valeurs de DGD mesurées par les fonctions de densité de Maxwell $f_{Max}(x, \alpha_{low})$, $f_{Max}(x, \alpha_{opt})$, $f_{Max}(x, \alpha_{upp})$ est évalué par le calcul du coefficient de corrélation de Pearson. Cette méthode d'évaluation calcule la covariance $COV(f_{DGD}(x), f_{Max}(x, \sigma))$ de la fonction de densité $f_{DGD}(x)$ des valeurs de DGD avec les fonction de densité de Maxwell $f_{Max}(x, \alpha_{opt})$, $f_{Max}(x, \alpha_{low})$, $f_{Max}(x, \alpha_{upp})$ calculées lors de la seconde étape de calcul. Les équations (3.25- 3.26) formulent le calcul du coefficient de corrélation de Pearson R_p sur la fonction densité $f_{DGD}(x)$ et sur les fonctions de densité $f_{Max}(x, \alpha_{opt})$, $f_{Max}(x, \alpha_{low})$, $f_{Max}(x, \alpha_{upp})$ selon Walpole *et al.* (2011, p.432 - 433).

$$R_p = \frac{COV(f_{DGD}(x), f_{Max}(x, \alpha))}{\sigma_{f_{DGD}(x)} \sigma_{f_{Max}(x, \sigma)}} \quad (3.24)$$

$$R_p = \frac{\sum_{i=1}^N (f_{Max}(x_i, \alpha_{opt}) - \overline{f_{Max}(x, \alpha_{opt})}) \times (f_{DGD}(x_i) - \overline{f_{DGD}(x)})}{\sqrt{\sum_{i=1}^N (f_{Max}(x_i, \alpha_{opt}) - \overline{f_{Max}(x, \alpha_{opt})})^2} \times \sqrt{\sum_{i=1}^N (f_{DGD}(x_i) - \overline{f_{DGD}(x)})^2}} \quad (3.25)$$

L'évaluation de l'ajustement par le coefficient de corrélation de Pearson s'établit sur une échelle de valeurs comprises entre 0 et 1. Plus la valeur du coefficient de corrélation de Pearson est proche de 1, plus les fonctions de densité de Maxwell sont proches de la fonction de densité $f_{DGD}(x)$ des valeurs de DGD mesurées. Le coefficient de corrélation de Pearson R_p a été calculé par l'application de la fonction *corrcoef* du module *Descriptive Statistics* offert par l'environnement Matlab.

3.6.2 Ajustement de l'histogramme des valeurs de DGD par la fonction de densité de Maxwell

La figure 3.18 décrit l'ajustement de l'histogramme normalisé $f_{DGD}(x)$ des valeurs de DGD par la fonction de densité de Maxwell sur la période de mesures du 5 au 11 novembre 2012 réalisées par le transpondeur cohérent sur la liaison du réseau Verizon.

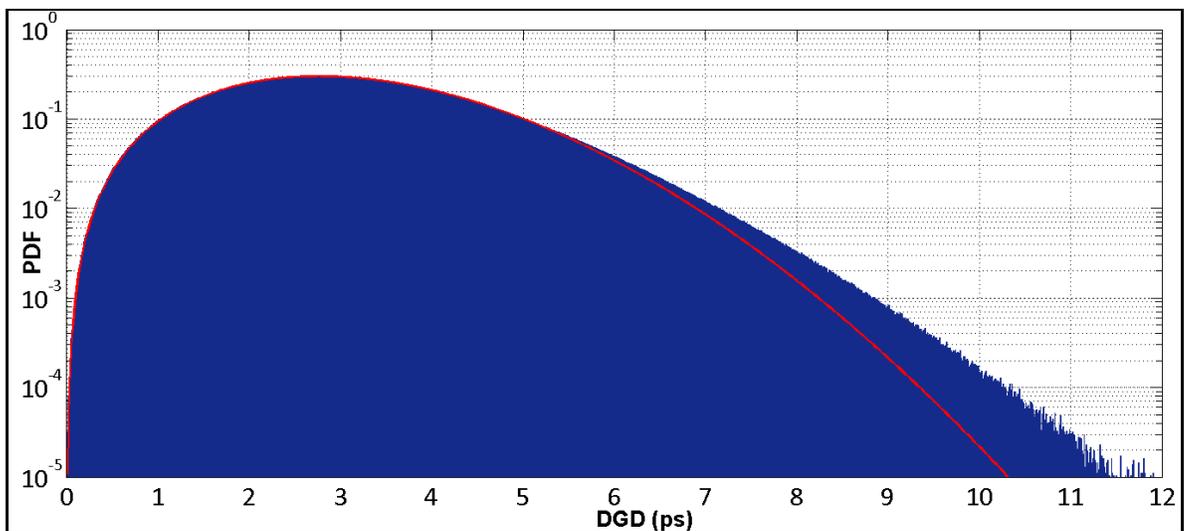


Figure 3.18 Ajustement de l'histogramme normalisé des valeurs de DGD par la fonction de densité de Maxwell

L'histogramme normalisé $f_{DGD}(x)$, présenté par la figure 3.18 décrit la fonction de densité de près de 117 millions de valeurs de DGD valides correspondantes à une période de 5 jours de mesures. La moyenne, l'écart type et la valeur maximale des valeurs de DGD mesurées par le transpondeur cohérent pour une période de 5 jours sont égaux à 3.148 ps, à 1.063 ps et à 11.8 ps respectivement.

Le paramètre optimisé α_{opt} et les paramètres α_{low} et α_{upp} , à un intervalle de confiance de 95 % sont égaux à 1.9528, à 1.9514 et à 1.9542 respectivement. Ceci démontre que les paramètres calculés α_{low} et α_{upp} à une incertitude de 95 % sont très proches du paramètre optimisé α_{opt} . L'écart relatif est égale à 0.07 % entre les paramètres α_{low} et α_{upp} , et le paramètre optimisé α_{opt} . 1187 classes sont créées lors du calcul de la fonction de densité $f_{DGD}(x)$ des valeurs de DGD mesurées. Plus le nombre total de classes n est grand, plus la confiance du paramètre optimisé est importante. Les écarts entre les fonctions de densité de Maxwell $f_{Max}(x, \alpha_{opt})$, $f_{Max}(x, \alpha_{low})$, $f_{Max}(x, \alpha_{upp})$ sont très faibles (Voir Figure 3.18). Les valeurs des fonctions de densité de Maxwell sont très proches de la fonction de densité $f_{DGD}(x)$ des valeurs de DGD jusqu'à 5.5 ps (Voir Figure 3.18). Des écarts significatifs sont observés entre la fonction de densité de Maxwell et la fonction de densité $f_{DGD}(x)$ des valeurs de DGD lorsque celles-ci sont supérieures à 5.5 ps.

Le coefficient de corrélation de Pearson a été calculé pour évaluer l'ajustement des fonctions de densité de Maxwell calculées sur différentes régions de l'histogramme normalisé $f_{DGD}(x)$ des valeurs de DGD. La méthode d'évaluation de l'ajustement a été appliquée sur trois plages de valeurs de l'histogramme normalisé $f_{DGD}(x)$. La première plage définit les valeurs de DGD inférieures à 5.5 ps. La deuxième plage désigne les valeurs de DGD supérieures à 5.5 ps et la dernière plage regroupe toutes les valeurs de l'histogramme normalisé $f_{DGD}(x)$. Le tableau 3.4 présente les coefficients de corrélation de Pearson calculés pour chacune des fonctions de densité de Maxwell $f_{Max}(x, \alpha_{opt})$, $f_{Max}(x, \alpha_{low})$, $f_{Max}(x, \alpha_{upp})$ sur les différentes régions des valeurs de DGD sélectionnées.

Tableau 3.4 Coefficients de corrélation de Pearson sur plusieurs régions de l'histogramme normalisé $f_{DGD}(x)$ des valeurs de DGD par la fonction de densité de Maxwell

valeurs de DGD sélectionnées	Coefficient de corrélation de Pearson		
	$f_M(x, \alpha_{low})$	$f_M(x, \alpha_{opt})$	$f_M(x, \alpha_{upp})$
DGD \leq 5.5 ps	0,9453	0,9428	0,9401
DGD $>$ 5.5 ps	0,8291	0,8384	0,8474
$0 \leq$ DGD \leq 12 ps	0,9671	0,9658	0,9644

Le coefficient de corrélation de Pearson est proche de 1 lorsque les valeurs de DGD sont inférieures à 5.5 ps. En effet, celui-ci varie de 0.9401 jusqu'à 0.9453 en fonction du paramètre α (Voir Tableau 3.4). Ceci démontre que la fonction de densité de Maxwell décrit efficacement la fonction de densité de probabilité des valeurs de DGD inférieures à 5.5 ps. Le coefficient de corrélation de Pearson fluctue entre 0.8291 et 0.8474 pour la région de valeurs de DGD supérieures à 5.5 ps (Voir Tableau 3.4). Ceci montre que la fonction de densité de Maxwell ne suit pas avec précision la fonction de densité $f_{DGD}(x)$ pour des valeurs de DGD supérieures à 5.5 ps. Le coefficient de corrélation de Pearson est très proche de 1 lorsque ce calcul est appliqué sur l'ensemble des valeurs de l'histogramme normalisé des valeurs de DGD. La fonction de densité de Maxwell apporte un modèle statistique proche de la fonction de densité de probabilité des valeurs de DGD mesurées avec le transpondeur cohérent. Pourtant, la fonction de densité de Maxwell n'apporte pas un ajustement précis sur la région de l'histogramme caractérisée par des valeurs de DGD supérieures à 5.5 ps.

Les travaux futurs pour cette étude seront menés sur l'algorithme de correction des valeurs de DGD pour accroître la qualité de l'ajustement sur tout l'histogramme des valeurs de DGD mesurées par le transpondeur cohérent. Si la courbe de correction n'aboutit pas à une plus grande précision de l'ajustement de la fonction de densité de Maxwell, alors une fonction de densité de probabilité sera développée en utilisant des méthodes mathématiques avancées afin qu'un nouveau modèle statistique des valeurs de DGD puisse être défini sur la liaison optique majoritairement aérienne du réseau Verizon.

3.7 Conclusion du chapitre

La structure de données indexée par « clé » est le résultat du traitement de données et a facilité le développement et l'application de quatre méthodes de caractérisation différentes appliquées sur les valeurs de DGD et de PDL décrites de la section 3.3 jusqu'à la section 3.6 de ce chapitre.

Le calcul de la moyenne et de l'écart type des valeurs de DGD et de PDL mesurées à chaque minute de mesures a permis la caractérisation des amplitudes de fluctuations, c'est-à-dire des valeurs de dérivée temporelle, sur le lien majoritairement aérien du réseau Verizon et sur le lien majoritairement enterrée du réseau CANARIE. Il a été possible d'identifier deux types de fluctuations observées sur le lien du réseau Verizon lors des jours et des nuits de mesures.

Le filtrage super gaussien d'ordre 4 définit un moyen très efficace de réduire le bruit généré par le système de transmission optique sur les mesures de $\langle DGD \rangle$ et de $\langle PDL \rangle$. La méthode de calcul de la fonction d'autocorrélation normalisée ACF_{norm} a permis d'approfondir l'analyse des variations des valeurs de $\langle DGD \rangle$ filtrées et des valeurs de $\langle PDL \rangle$ filtrées. L'application de cette méthode de calcul apporte de nouvelles connaissances et confirme les caractérisations réalisées précédemment par Woodward *et al.* (2014), Kawasaki *et al.* (2007) et Waddy *et al.* (2005).

L'ajustement de l'histogramme des valeurs de DGD par la fonction de densité de Maxwell n'est valide que pour les valeurs de DGD inférieures à 5.5 ps. Ceci démontre qu'il est nécessaire d'étudier avec approfondissement la fonction de densité $f_{DGD}(x)$ des valeurs de DGD en vue d'identifier les phénomènes à l'origine des fluctuations des valeurs de DGD mesurés par le transpondeur cohérent.

Le «savoir-faire» ressortant de cette expérience constitue une solide base pour les travaux de caractérisation futurs des effets de polarisation mesurés avec le transpondeur cohérent.

CHAPITRE 4

MONITORING ET CARACTÉRISATION DES VITESSES DE ROTATION ANGULAIRE DES ÉTATS DE POLARISATION AVEC UN POLARIMÈTRE

4.1 Introduction

Ce chapitre présente le traitement, les méthodes de calcul, les méthodes d'analyse et les résultats de caractérisation des fluctuations des états de polarisation mesurées sur une liaison optique majoritairement aérienne, composée de 900 kilomètres de câbles OPGW (*Optical ground wire*), appartenant au réseau Verizon.

Crawford *et al.* (2013) énonce que ce type d'installation est plus sensible aux mesures d'activités intenses en polarisation que celles subsistant sur d'autres types d'installations (p. ex. liaison optique enterrée). Afin que des fluctuations rapides des états de polarisation soient mesurées, la fonction « polarimètre » du synthétiseur de polarisation a été utilisée pour la mesure rapide et avec une résolution temporelle élevée (20 kHz) des paramètres de Stokes (S_1 , S_2 , S_3). Pour cette raison, cette étude approfondit les travaux précédents de *monitoring* des fluctuations des états de polarisation de Peterson, *et al.* (2004), de Matsuda, *et al.* (2007) et de Nelson (2012).

Tout au long de cette étude, ma contribution a porté sur le traitement des données enregistrées par le polarimètre, sur le développement des méthodes de calcul et des méthodes d'analyse sous la forme de scripts Matlab. Les méthodes d'analyse ont été développées par la configuration et l'application de modules Matlab pour la caractérisation avec précision des fluctuations des états de polarisation.

4.2 Traitement des données enregistrées par le polarimètre

Le traitement de plusieurs milliards de données enregistrées au polarimètre sur de longues périodes de mesure (totalisant plus de 14 jours) a représenté un véritable défi au cours de cette expérience. Le traitement des fichiers de mesure enregistrés par le polarimètre comporte plusieurs processus. Le résultat de ma contribution pour le traitement des données se présente sous la forme d'un seul script Matlab en vue de l'automatisation des calculs appliqués sur les paramètres de Stokes (S_1 , S_2 , S_3). La figure 4.1 présente la séquence de traitement appliquée sur les 39 000 matrices constituées de 200 000 vecteurs Stokes pour toutes les périodes d'acquisition de mesures réalisées avec le polarimètre (*Voir* section 2.2.2).

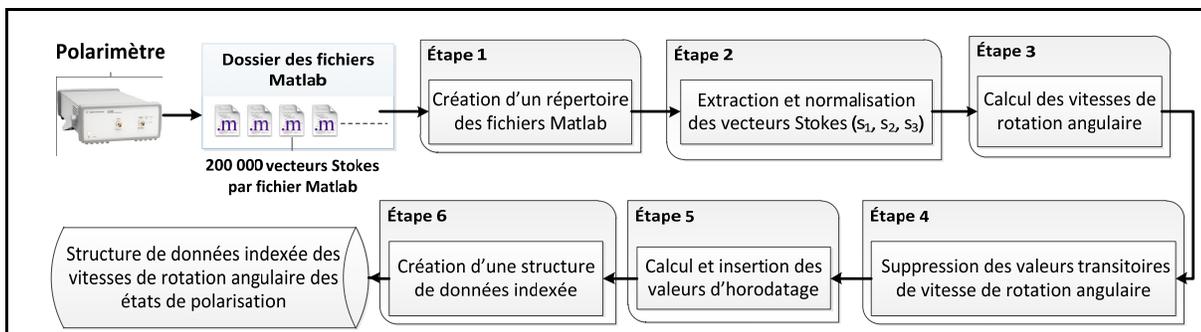


Figure 4.1 Séquence de traitement des fichiers de mesure enregistrés avec le polarimètre

Le script Matlab dédié au traitement des données comporte chacune des étapes de traitement présentées à la figure 4.1 (*Voir* ANNEXE XXX, p. 198). Chaque étape de traitement intègre des fonctions offertes par le module *Mathematics Toolbox* qui optimisent les opérations et les calculs appliqués sur les matrices constituées des vecteurs Stokes (S_1 , S_2 , S_3) afin que le délai de traitement soit le plus court possible. L'objectif final du traitement des fichiers de mesures est la création d'une structure de données indexée permettant d'enregistrer les matrices Matlab constituées des valeurs de vitesse de rotation angulaire des états de polarisation en fonction du temps.

La première étape a pour objectif la création d'un répertoire qui désigne les adresses des fichiers Matlab enregistrés avec le polarimètre (*Voir* Figure 4.1). Ce répertoire d'adresses se

présente sous la forme d'une structure de données où chaque adresse correspond un numéro d'identification unique *ID* désignant un fichier de mesure (*Voir* ANNEXE XXXI, p. 201). Le passage par une structure de données indexée a pour objectif de charger rapidement les fichiers Matlab et de faciliter l'automatisation des étapes de traitement des données enregistrées avec le synthétiseur de polarisation (*Voir* Figure 4.1).

Les paramètres de Stokes (S_1, S_2, S_3) sont ensuite extraits et normalisés lors de la seconde étape de traitement (*Voir* Figure 4.1). Les matrices constituées de 200 000 vecteurs Stokes sont lues à partir des adresses du répertoire des fichiers Matlab. Les vecteurs Stokes ont ensuite été normalisés pour que les calculs appliqués puissent caractériser avec précision la partie polarisée du signal optique DP-QPSK. Hui *et al.* (2006, p.191) indiquent la nécessité de normaliser les vecteurs Stokes afin de ne pas inclure la partie non polarisée du signal optique et de caractériser avec précision les états de polarisation mesurés. Les équations (4.1-4.3) formulent le calcul de normalisation des vecteurs Stokes selon Hui *et al.* (2006, p.191).

$$s_1(t) = \frac{S_1(t)}{\sqrt{S_1^2(t) + S_2^2(t) + S_3^2(t)}} \quad (4.1)$$

$$s_2(t) = \frac{S_2(t)}{\sqrt{S_1^2(t) + S_2^2(t) + S_3^2(t)}} \quad (4.2)$$

$$s_3(t) = \frac{S_3(t)}{\sqrt{S_1^2(t) + S_2^2(t) + S_3^2(t)}} \quad (4.3)$$

Les résultats des équations (4.1 -4.3) se présentent sous la forme de matrices constituées de 200 000 vecteurs Stokes normalisées pour une période de 14 jours de mesures réalisées avec le polarimètre (*Voir* ANNEXE XXXII, p. 203).

Les vitesses de rotation angulaire des états de polarisation ont été calculées lors de la troisième étape de traitement des données (*Voir* Figure 4.1). Le calcul des vitesses de rotation angulaire entre deux vecteurs Stokes adjacents définit un moyen efficace de caractériser les fluctuations des états de polarisation du signal optique transmis selon Nelson (2012),

Matsuda *et al.* (2009) et Noutsios (2007). Reimer *et al.* (2013) montrent que les vitesses de rotation angulaire des états de polarisation définissent l'angle de l'arc sous tendu entre deux vecteurs Stokes normalisés $\vec{s}(t_1)$ et $\vec{s}(t_2)$ sur la sphère de Poincaré tel qu'indiqué dans le premier chapitre (*Voir* section 1.2.3). Le produit scalaire entre deux vecteurs Stokes normalisés adjacents est le calcul des vitesses de rotation angulaire exprimées en rad/ms et désignées par la variable $\frac{d\vec{S}}{dt}$. (*Voir* ANNEXE XXXIII, p. 205). L'équation (4.4) décrit le développement du produit scalaire $\vec{s}(t_1) \cdot \vec{s}(t_2)$ de chacun des paramètres de Stokes normalisés (s_1, s_2, s_3) pour le calcul des vitesses de rotation angulaire des états de polarisation $\frac{d\vec{S}}{dt}$ à partir de l'équation (1.12) adapté de Bao *et al.* (2004).

$$\frac{d\vec{S}}{dt} = \frac{\text{acos}(s_1(t_1) \cdot s_1(t_2) + s_2(t_1) \cdot s_2(t_2) + s_3(t_1) \cdot s_3(t_2))}{t_2 - t_1} \quad (4.4)$$

Le résultat de cette troisième étape de traitement se présente sous la forme de 39 000 matrices constituées des vitesses de rotation angulaire $\frac{d\vec{S}}{dt}$ des états de polarisation sur une période totalisant 14 jours de mesures (*Voir* section 2.2.2). Les valeurs de vitesse de rotation angulaire des états de polarisation sont aussi représentées par l'acronyme *ARR* (signifiant « *Angular rotation rate* ») dans toutes les équations et dans toutes les figures y référant tout au long de ce chapitre.

La quatrième étape a pour objectif la suppression des valeurs transitoires des vitesses de rotation angulaire des états de polarisation causées par le montage de mesure (*Voir* Figure 2.2). Ces valeurs transitoires sont très supérieures aux effets transitoires de polarisation physiques mesurés par le polarimètre et peuvent se produire jusqu'à 70 ms du temps de début de chaque période d'acquisition. Seulement 30 vecteurs Stokes adjacents (soit une durée inférieure à 1.5 ms) constituent la période transitoire qui se produit au début de chaque mesure. Pour cette raison, celles-ci ne sont pas valides et doivent être supprimées avant de caractériser les vitesses de rotation angulaire des états de polarisation mesurés par le polarimètre sur la liaison du réseau Verizon. Une fonction Matlab a été développée et a été

insérée dans le script Matlab dédié au traitement des données enregistrées au polarimètre pour l'automatisation de la localisation et de la suppression de la période transitoire (*Voir ANNEXE XXXIV, p. 207*). Le résultat final de cette quatrième étape de traitement se présente sous la forme d'une matrice réduite qui sauvegarde seulement les valeurs de vitesse de rotation angulaire mesurées après la période transitoire.

La cinquième étape consiste à former et à insérer les valeurs d'horodatage, appelées aussi *timestamp*, sur chaque matrice constituée des valeurs valides de vitesses de rotation angulaire $\frac{d\vec{S}}{dt}$ des états de polarisation (*Voir Figure 4.1*). Le titre de chaque fichier définit la valeur d'horodatage (*Voir section 2.6.2.2*). Les valeurs qui constituent les étiquettes de temps (p. ex. année, mois, jour, etc.) ont été extraites par l'application d'un processus de traitement qui associe une valeur d'horodatage pour chaque valeur de vitesse de rotation angulaire calculée. Une fonction Matlab a été développée en vue de l'automatisation de l'insertion des temps de mesure aux matrices constituées des valeurs de vitesse de rotation angulaire valides $\frac{d\vec{S}}{dt}$ (*Voir ANNEXE XXXV, p. 209*). Les résultats de cette étape de traitement se présentent sous la forme de 39 000 matrices constituées des valeurs valides de vitesse de rotation angulaire et du temps de mesure correspondant pour une période totalisant 14 jours de mesures réalisées au polarimètre (*Voir section 2.2.2*).

Cette dernière étape consiste à enregistrer dans une structure de données indexée les matrices précédemment calculées et constituées des vitesses de rotation angulaire en fonction du temps (*Voir Figure 4.1*). Cette structure de données associe un numéro d'identifiant unique indiqué par la variable *idx* pour chaque matrice constituée des valeurs de vitesse de rotation angulaire horodatées (*Voir ANNEXE XXXVI, p. 211*). Le numéro d'identifiant *idx* est la « clé » qui désigne uniquement le numéro de la matrice constituée des valeurs de vitesse de rotation angulaire horodatées. Cette structure de données indexée ne comporte pas de « clé » désignant un paramètre de *monitoring* spécifique tel que réalisé lors du traitement des données enregistrées avec le transpondeur cohérent car la vitesse de rotation angulaire des états de polarisation $\frac{d\vec{S}}{dt}$ est la seule variable étudiée dans ce chapitre. La figure 4.3 présente la

structure de données indexée qui résulte de l'application du script Matlab dédié au traitement des données enregistrées au polarimètre.

Identifiant de chaque matrice	Matrices des vitesses de rotation angulaire et des valeurs d'horodatage		Temps de début de chaque mesure
Field	Value	Value	
id1	<199757x2 single>	7.3518e+05	
id2	<199763x2 single>	7.3518e+05	
id3	<199758x2 single>	7.3518e+05	
id4	<199755x2 single>	7.3518e+05	
id5	<199760x2 single>	7.3518e+05	
id6	<199761x2 single>	7.3518e+05	
id7	<199761x2 single>	7.3518e+05	
id8	<199758x2 single>	7.3518e+05	
id9	<199764x2 single>	7.3518e+05	
id10	<199742x2 single>	7.3518e+05	

Figure 4.2 Structure de données indexée obtenue après l'exécution du script Matlab dédié au traitement de données enregistrées au polarimètre

Cette structure de données indexée comporte 39 000 matrices, elles-mêmes constituées des valeurs de vitesse de rotation angulaire des états de polarisation horodatés, de la valeur du temps de début de chaque mesure et de l'identifiant unique idx (Voir Figure 4.3). La taille de chaque matrice varie entre 198 400 et 199 999 valeurs dépendamment du temps de mesure auquel se produit la période transitoire. Cette structure de données est constituée de près de 7.79 milliards de valeurs de vitesse de rotation angulaire en fonction du temps pour une période de 14 jours de mesures sur la liaison du réseau Verizon.

4.3 Étude de la fonction de densité des vitesses de rotation angulaire des états de polarisation

Cette section propose la caractérisation et l'ajustement de la fonction de densité des vitesses de rotation angulaire des états de polarisation. La méthode de calcul et la méthode d'ajustement de la fonction de densité des vitesses de rotation angulaire sont tout d'abord

décrites de façon détaillée dans la première partie. La seconde partie de cette étude a pour objectif de vérifier l'hypothèse selon laquelle les vitesses de rotation angulaire des états de polarisation peuvent être ajustées par la fonction de densité de Rayleigh telle que démontrée dans les travaux de Peterson *et al.* (2004) et de Matsuda *et al.* (2009). La seconde partie de cette section est aussi constituée de l'analyse d'autres fonctions de densité de probabilité ajustées sur l'histogramme normalisé des vitesses de rotation angulaire des états de polarisation.

4.3.1 Méthode de calcul et d'ajustement de l'histogramme des vitesses de rotation angulaire des états de polarisation

La figure 4.3 décrit les étapes de calcul et d'ajustement de la fonction de densité des vitesses de rotation angulaire des états de polarisation mesurés sur le lien aérien du réseau Verizon.

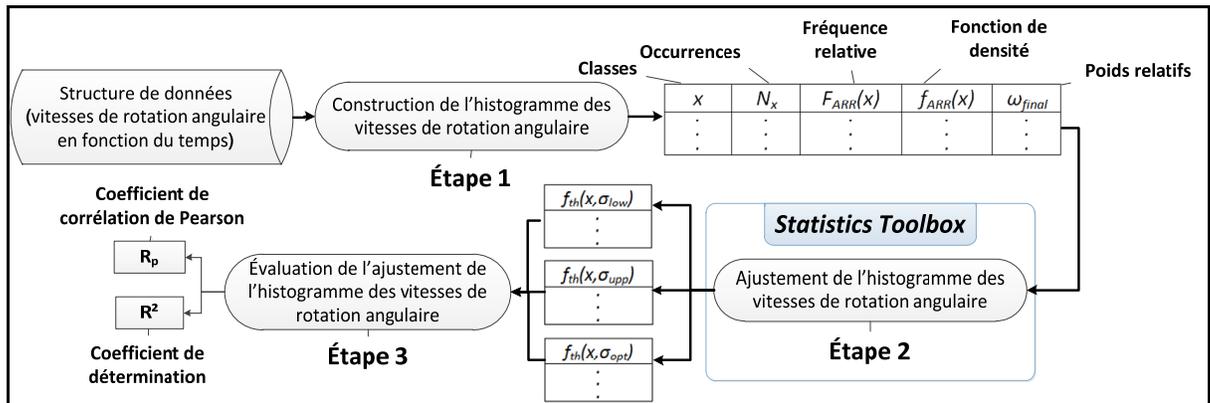


Figure 4.3 Méthode de calcul et d'ajustement de la fonction de densité des vitesses de rotation angulaire des états de polarisation

Un script Matlab a été développé et appliqué sur la structure de données indexée en vue de l'automatisation du calcul de l'histogramme normalisé des vitesses de rotation angulaire des états de polarisation lors de la première étape de calcul sur la figure 4.3 (Voir ANNEXE XXXVII, p. 213).

La seconde étape de calcul porte sur la méthode d'ajustement de l'histogramme normalisé des vitesses de rotation angulaire $\frac{d\vec{s}}{dt}$ des états de polarisation dans un second script Matlab (*Voir* ANNEXE XXXVIII, p. 217) qui désigne un outil d'analyse adapté pour le test de plusieurs fonctions de densité de probabilité (p. ex. fonction de densité de Rayleigh). Le résultat de l'ajustement se présente sous la forme de trois matrices. Ces matrices contiennent respectivement les valeurs de la fonction de densité théorique optimisée $f_{th}(x, \sigma_{opt})$, et les valeurs des fonctions de densité $f_{th}(x, \sigma_{low})$ et $f_{th}(x, \sigma_{upp})$ à un intervalle d'incertitude de 95% lors de la seconde étape de calcul (*Voir* Figure 4.3). La troisième étape de calcul consiste à évaluer les ajustements par le calcul du coefficient de corrélation de Pearson et par le calcul du coefficient de détermination (*Voir* Figure 4.3).

Les trois étapes de calcul décrites par la figure 4.3 sont présentées de façon détaillée de la section 4.3.1.1 jusqu'à la section 4.3.1.3.

4.3.1.1 Étape 1 : calcul de l'histogramme normalisé des vitesses de rotation angulaire

Les valeurs d'occurrence N_x c'est-à-dire le nombre de répétitions des valeurs de vitesse de rotation angulaire $\frac{d\vec{s}}{dt}$ comprises entre deux classes x_1 et x_2 sont tout d'abord calculées. La résolution, appelée aussi la taille de l'histogramme, désignée par la variable Δx est égale à 0.01 rad/ms. La fonction de probabilité appelée $F_{ARR}(x)$ est par la suite calculée et comporte les fréquences relatives aux valeurs d'occurrence N_x . Ces valeurs définissent les probabilités estimées \hat{p}_x des vitesses de rotation angulaire pour chaque classe de l'histogramme. Navidi (2010) montre que la valeur d'une probabilité estimée \hat{p}_x est calculée par la division du nombre d'occurrence N_x sur le nombre total de valeurs N_{TOT} . L'équation (4.5) formule le calcul de la fonction de probabilité $F_{ARR}(x)$ sur les valeurs de vitesse de rotation angulaire $\frac{d\vec{s}}{dt}$ adaptée de Navidi (2010, p.27).

$$F_{ARR}(x) = \hat{p}(ARR = x) = \frac{N_x}{N_{TOT}} \quad (4.5)$$

L'histogramme constitué des valeurs de la fonction $F_{ARR}(x)$ est ensuite normalisé par le calcul de l'aire pour l'obtention de la fonction de densité de probabilité $f_{ARR}(x)$ des vitesses de rotation angulaire. Kreyszig *et al.* (2011, p.828) propose le calcul de l'aire de l'histogramme des vitesses de rotation angulaire des états de polarisation par l'application de la méthode des trapèzes (Voir ANNEXE XXVI, p. 187). Le calcul de normalisation de l'histogramme consiste à multiplier la fonction de probabilité $F_{ARR}(x)$ par l'inverse de l'aire précédemment calculée. L'équation (4.6) décrit le calcul de la fonction de densité de probabilité $f_{ARR}(x)$ sur les vitesses de rotation angulaire des états de polarisation.

$$f_{ARR}(x) = F_{ARR}(x) \times 1/Aire \cong \frac{dF_{ARR}(x)}{dx} \quad (4.6)$$

Adaptée de Navidi (2010, p.27)

L'histogramme normalisé des vitesses de rotation angulaire calculées désigne les valeurs de la fonction de densité de probabilité $f_{ARR}(x)$. L'équation (4.7) formule les propriétés relatives de la fonction de densité de probabilité $f_{ARR}(x)$ des vitesses de rotation angulaire des états de polarisation mesurés par le polarimètre.

$$f_{ARR}(x_i) \geq 0 \quad \text{et} \quad \int_{x_0}^{x_n} f_{ARR}(x_i) dx_i = 1 \quad (4.7)$$

Adaptée de Navidi (2010, p.103)

L'aire de la fonction de densité de probabilité $f_{ARR}(x)$ des vitesses de rotation angulaire est égale à 1 sur l'intervalle] x_0 ; x_n]. L'équation (4.8) décrit le calcul de la valeur de probabilité selon laquelle une vitesse de rotation angulaire soit comprise entre les classes x_1 et x_2 .

$$p(x_1 < ARR \leq x_2) = \int_{x_1}^{x_2} f_{ARR}(x) dx \quad (4.8)$$

Les valeurs de poids de chaque classe x de l'histogramme sont par la suite calculées pour accroître la précision de l'ajustement de la fonction de densité de probabilité $f_{ARR}(x)$ en prenant l'inverse des valeurs d'incertitude, c'est-à-dire des marges d'erreurs $\sigma_{p_x}^2$, selon Navidi (2010, p.210). Crawford *et al.* (2013) montre qu'il est plus judicieux de calculer les poids au moyen d'une échelle logarithmique afin que les valeurs de probabilité les plus faibles (proche de 10^{-10}) de vitesse de rotation angulaire puisse être dotées d'une plus grande influence lors de la procédure d'ajustement. L'équation (4.9) formule le calcul des poids avec une échelle logarithmique (à base-10) de chaque classe x en fonction des valeurs d'occurrence N_x et des probabilités observées \hat{p}_x des vitesses de rotation angulaire calculées.

$$\omega_{\log_{10}} = \frac{1}{\sigma_{p_x}^2} = (\ln 10)^2 \times \frac{N_x}{(1 - \hat{p}_x)} \quad (4.9)$$

L'équation (4.10) est le résultat du développement mathématique réalisé par Crawford *et al.* (2013) pour l'expression des poids logarithmiques $\omega_{\log_{10}}$ (Voir ANNEXE XXXIX, p. 225). Wolberg (2005, p.49) montre que les poids de chaque valeur de l'histogramme sont proportionnels aux valeurs d'occurrence N_x des vitesses de rotation angulaire des états de polarisation lorsque celles-ci sont calculées au moyen d'une échelle logarithmique, ce qui est validé par l'expression de la formule des valeurs de poids selon l'équation (4.10). Le résultat issu de l'application du premier script Matlab est une matrice constituée des valeurs de classes x , des valeurs d'occurrence N_x , des valeurs de la fonction de probabilité $F_{ARR}(x)$, des valeurs de la fonction de densité de probabilité $f_{ARR}(x)$ des vitesses de rotation angulaire des états de polarisation et des valeurs de poids logarithmiques $\omega_{\log_{10}}$ (Voir Figure 4.3).

4.3.1.2 Étape 2 : ajustement de l'histogramme des vitesses de rotation angulaire

Les fonctions du module *NonLinearModel* de la librairie *Statistics Toolbox* de Matlab ont été configurées et appliquées sur la matrice précédemment calculée lors de la première étape de calcul (Voir Figure 4.3) en vue du calcul de l'ajustement de l'histogramme des vitesses de rotation angulaire. Le module *NonLinearModel* comporte de nombreuses fonctionnalités

facilitant la mise en place de la procédure d'ajustement. Ce module est composé de plusieurs méthodes qui ajustent l'histogramme des vitesses de rotation angulaire des états de polarisation par plusieurs fonctions de densité de probabilité (p.ex. fonction de densité de Rayleigh). L'histogramme des vitesses de rotation angulaire est tout d'abord ajusté par l'application de la fonction *NonlinearModel.fit* (Voir ANNEXE XL, p. 227). Cette fonction Matlab implémente la méthode pondérée des moindres carrés. Cette méthode désigne la fonction mathématique effectuant le calcul de la somme des carrés des résidus, c'est-à-dire des écarts, entre la fonction de densité des vitesses de rotation angulaire $f_{ARR}(x)$ et la fonction de densité théorique $f_{th}(x, \sigma)$.

Crawford *et al.* (2013) indique la nécessité de calculer et d'appliquer la fonction *NonLinearModel.fit* en prenant le logarithme (à base-10) de la fonction de densité de probabilité des vitesses de rotation angulaire $f_{ARR}(x)$ et de la fonction de densité théorique $f_{th}(x, \sigma)$ afin que les faibles valeurs de l'histogramme, désignant les vitesses rapides de rotation angulaire des états de polarisation, aient une plus grande influence lors la procédure d'ajustement. Pour cette raison, la fonction *NonlinearModel.fit* a été appliquée sur les fonctions $\log f_{ARR}(x)$ et $\log f_{th}(x, \sigma)$ ainsi que sur les poids $\omega_{\log_{10}}$ calculés au moyen d'une échelle logarithmique sur les valeurs de l'histogramme normalisé des vitesses de rotation angulaire. L'équation (4.10) formalise la méthode pondérée des moindres carrés, appelée $S_{\log w}(\sigma)$, par la fonction *NonlinearModel.fit* sur l'histogramme des vitesses de rotation angulaire des états de polarisation.

$$S_{\log w}(\sigma) = \sum_{i=1}^N \omega_{\log_{10}} (\log f_{ARR}(x_i) - \log f_{th}(x_i, \sigma))^2 = \sum_{i=1}^N \omega_{\log_{10}} \times r_i^2(\sigma) \quad (4.10)$$

Adaptée de Walpole *et al.* (2007)

Cette fonction Matlab offre une solution numérique au problème de minimisation de la méthode pondérée des moindres carrés $S_{\log w}(\sigma)$ par l'application de l'algorithme de Levenberg-Marquardt en vue de l'obtention du paramètre optimisé σ_{th} (Voir ANNEXE XXVII, p. 189). Lorsque la méthode pondérée des moindres carrés $S_{\log w}(\sigma)$ est au plus

proche de 0, alors le paramètre optimisé σ_{opt} de la fonction de densité théorique $f_{th}(x, \sigma_{opt})$ est identifié.

L'incertitude Δ du paramètre σ_{opt} est par la suite calculée par l'application de la fonction *nonLinearModel.CoeffCI* offerte par le module *NonLinearModel*. Cette fonction intègre le calcul de l'approximation de l'erreur standard du paramètre optimisé σ_{opt} désigné par la variable $SE_{\sigma_{opt}}$ (Voir ANNEXE XXVIII, p. 182). L'incertitude Δ a été calculée en multipliant l'erreur standard du paramètre $SE_{\sigma_{opt}}$ et le coefficient τ de la table de la loi de *Student* à un intervalle de confiance θ (Voir ANNEXE XXVIII, p. 184). Le résultat de cette fonction se présente sous la forme de deux paramètres σ_{low} et σ_{upp} . Ces deux paramètres désignent la limite inférieure et supérieure à un intervalle d'incertitude de 95%. L'équation (4.11) formalise le calcul de l'incertitude Δ du paramètre σ_{opt} de la fonction de densité théorique $f_{th}(x, \sigma_{opt})$ adaptée de Walpole *et al.* (2007).

$$\Delta = \pm \tau_{1-\theta/2}^{n-1} \times SE_{\sigma_{opt}} \quad (4.11)$$

L'équation (4.12) décrit les limites inférieures et supérieures du paramètre optimisé σ_{opt} à un intervalle de confiance θ adaptée de Walpole *et al.* (2007).

$$\sigma_{opt} - \tau_{1-\theta/2}^{n-1} \times SE_{\sigma_{opt}} \leq \sigma_{opt} \leq \sigma_{opt} + \tau_{1-\theta/2}^{n-1} \times SE_{\sigma_{opt}} \quad (4.12)$$

Les paramètres σ_{low} et σ_{upp} qui désignent les limites à un intervalle de confiance de 95% du paramètre σ_{opt} ont été calculés selon l'équation (4.13) au moyen du coefficient τ de la table de la loi de *Student*.

$$\begin{aligned} \sigma_{opt} - 1,96 \times SE_{\sigma_{opt}} &\leq \sigma_{opt} \leq \sigma_{opt} + 1,96 \times SE_{\sigma_{opt}} \\ \sigma_{low} &\leq \sigma_{opt} \leq \sigma_{upp} \end{aligned} \quad (4.13)$$

Les fonctions de distributions théoriques $f_{th}(x, \sigma_{opt})$, $f_{th}(x, \sigma_{low})$, $f_{th}(x, \sigma_{upp})$ ont été finalement calculées par l'application de la fonction *nonLinearModel.Predict*.

4.3.1.3 Étape 3 : évaluation de l'ajustement de l'histogramme des vitesses de rotation angulaire des états de polarisation

Ma contribution lors de la troisième étape de calcul porte sur l'intégration du calcul de deux indicateurs en vue de l'évaluation des ajustements réalisés sur l'histogramme des vitesses de rotation angulaire des états de polarisation. Le coefficient de corrélation de Pearson R_p et le coefficient de détermination R^2 définissent les deux indicateurs calculés lors de cette étape de calcul.

Walpole *et al.* (2005, p.432) propose le calcul du coefficient de corrélation de Pearson R_p en vue de l'évaluation de l'ajustement des fonctions de densité de probabilité sur l'histogramme des vitesses de rotation angulaire. Le coefficient de corrélation de Pearson estime efficacement la qualité d'un ajustement sur tous les types de fonction de densité de probabilité. Cette méthode mathématique calcule la covariance $COV(f_{ARR}(x), f_{th}(x, \sigma))$ de la fonction de densité $f_{ARR}(x)$ des vitesses de rotation angulaire et des fonctions de densité de probabilité théoriques $f_{th}(x, \sigma_{opt})$, $f_{th}(x, \sigma_{low})$, $f_{th}(x, \sigma_{upp})$ calculées lors de la seconde étape de calcul. Les équations (4.14- 4.15) formulent le calcul du coefficient de corrélation de Pearson R_p appliqué sur la fonction de densité $f_{ARR}(x)$ des vitesses de rotation angulaire calculées et sur les fonctions de densité théoriques $f_{th}(x, \sigma_{opt})$, $f_{th}(x, \sigma_{low})$, $f_{th}(x, \sigma_{upp})$ selon Walpole *et al.* (2011, p.432 - 433).

$$R_p = \frac{COV(f_{ARR}(x), f_{th}(x, \sigma))}{\sigma_{f_{ARR}(x)} \sigma_{f_{th}(x, \sigma)}} \quad (4.14)$$

$$R_p = \frac{\sum_{i=1}^N (f_{th}(x_i, \sigma_{opt}) - \overline{f_{th}(x, \sigma_{opt})}) \times (f_{ARR}(x_i) - \overline{f_{ARR}(x)})}{\sqrt{\sum_{i=1}^N (f_{th}(x_i, \sigma_{opt}) - \overline{f_{th}(x, \sigma_{opt})})^2} \times \sqrt{\sum_{i=1}^N (f_{ARR}(x_i) - \overline{f_{ARR}(x)})^2}} \quad (4.15)$$

Adaptée de Walpole *et al.* (2011, p.432)

Le coefficient de corrélation de Pearson évalue l'ajustement réalisé sur une échelle de 0 à 1. Plus la valeur du coefficient de corrélation de Pearson est proche de 1, plus les fonctions de densité théoriques $f_{th}(x, \sigma)$ sont proches de la fonction de densité $f_{ARR}(x)$ des vitesses de rotation angulaire calculées.

Le second indicateur calculé est le calcul du coefficient de détermination appelé R^2 . Le coefficient de détermination R^2 est plus sensible aux écarts entre la fonction de densité de probabilité $f_{ARR}(x)$ et de la fonction de densité théorique $f_{th}(x, \sigma)$ contribuant à une évaluation plus significative des ajustements réalisés selon Walpole *et al.* (2011, p.407). Le coefficient de détermination R^2 est défini par le rapport de la somme des valeurs résiduelles indiquée par la variable SSR , et de la variance $\sigma_{f_{ARR}(x)}^2$ des valeurs de la fonction de densité $f_{ARR}(x)$ des vitesses de rotation angulaire des états de polarisation. Les équations (4.16- 4.17) formulent le calcul du coefficient de détermination R^2 ajusté au nombre de paramètres variables p de la fonction de densité de probabilité $f_{th}(x, \sigma)$.

$$R^2 = 1 - \left(\frac{n-1}{n-p-1} \right) \times \frac{SSR}{\sigma_{f_{ARR}(x)}^2} \quad (4.16)$$

$$R^2 = 1 - \left(\frac{n-1}{n-p-1} \right) \times \frac{\sum_{i=1}^n (f_{ARR}(x_i) - f_{th}(x_i, \sigma_{opt}))^2}{\sum_{i=1}^n (f_{ARR}(x_i) - \overline{f_{ARR}(x)})^2} \quad (4.17)$$

Adaptée de Walpole *et al.* (2011, p.407)

Le coefficient de détermination est dépendant du nombre total de classes n , du nombre de paramètres variables p , de la somme des valeurs résiduelles SSR et de la variance $\sigma_{f_{ARR}(x)}^2$

de la fonction de densité de probabilité $f_{ARR}(x)$. L'évaluation de l'ajustement par le coefficient de détermination R^2 s'établit sur une échelle de valeurs comprises entre 0 et 1.

4.3.2 Ajustement de l'histogramme des vitesses de rotation angulaire des états de polarisation par la fonction de densité de Rayleigh

Peterson *et al.* (2004) et Matsuda *et al.* (2009) montrent que la fonction de densité de probabilité $f_{ARR}(x)$ des vitesses de rotation angulaire peut être ajustée par une fonction de densité de Rayleigh. Cette étude a pour objectif le test de la fonction de densité de Rayleigh sur l'histogramme des vitesses de rotation angulaire mesurées par le polarimètre sur le lien aérien du réseau Verizon. L'équation (4.18) formule la fonction de densité de Rayleigh désignée par la variable $f_G(x, \sigma)$ constituée du paramètre caractéristique σ selon Walck (2007).

$$f_G(x, \sigma) = \frac{x}{\sigma^2} e^{-\left(\frac{x^2}{2\sigma^2}\right)} \text{ avec } x \geq 0 \text{ et } \sigma \geq 0 \quad (4.18)$$

Comme décrit dans la section (4.3.1.2), Crawford *et al.* (2013) propose de calculer le logarithme de l'histogramme des vitesses de rotation angulaire et de la fonction de distribution de Rayleigh $\log f_G(x, \sigma)$ afin que les faibles valeurs de l'histogramme (proche de 10^{-10}) exercent une plus grande influence lors de la procédure d'ajustement. L'équation (4.19) formule le logarithme de la fonction de la densité de Rayleigh appelée $\log f_G$.

$$\log f_G(x, \sigma) = -2 \log \sigma + \log x - \left(\frac{x}{\sqrt{2} \times \sigma}\right)^2 \text{ avec } x > 0 \text{ et } \sigma > 0 \quad (4.19)$$

L'histogramme normalisé des vitesses de rotation angulaire et les fonctions de densité de Rayleigh sont visualisés par l'application d'un script Matlab dédié à l'optimisation de l'affichage de toutes les valeurs de l'histogramme au moyen d'une échelle logarithmique (Voir ANNEXE XLI, p. 229). La figure 4.4 décrit l'ajustement de la fonction de densité de

Rayleigh sur l'histogramme des vitesses de rotation angulaire pour une période totalisant 14 jours de mesure avec le polarimètre sur le lien aérien du réseau Verizon.

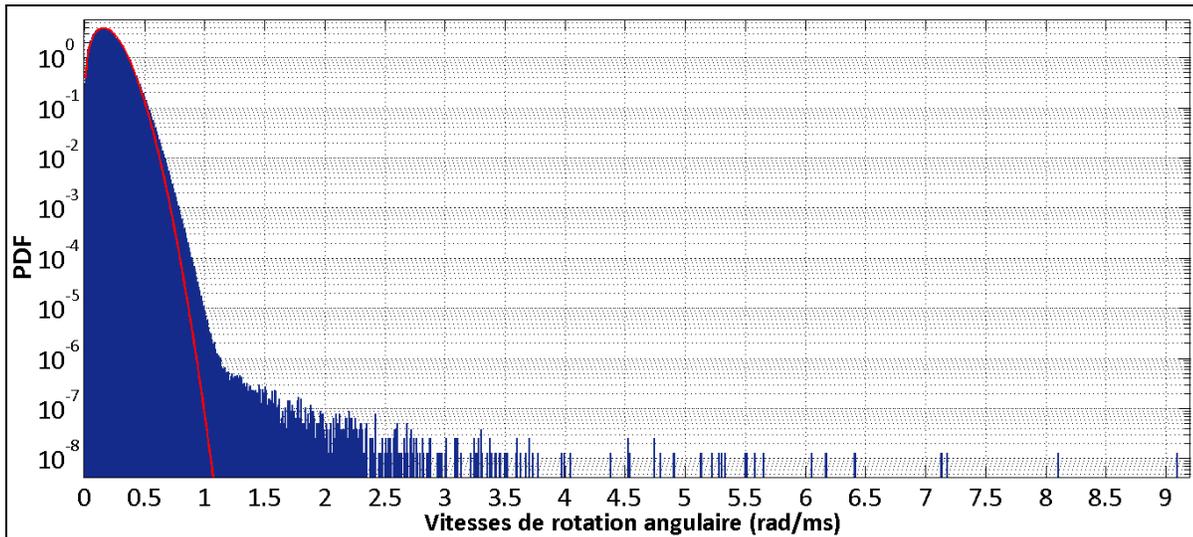


Figure 4.4 Ajustement de l'histogramme $f_{ARR}(x)$ des vitesses de rotation angulaire par la fonction de densité de Rayleigh

Les valeurs de l'histogramme, présentées par la figure 4.4, sont les valeurs de la fonction de densité $f_{ARR}(x)$ des vitesses de rotation angulaire et sont réparties dans 910 classes. La valeur maximale des vitesses de rotation angulaire calculées est égale à 9.09 rad/ms sur le lien optique du réseau Verizon. La moyenne et l'écart type des vitesses de rotation angulaire sont égaux à 0.1982 rad/ms et à 0.1054 rad/ms respectivement sur toutes les périodes de mesures réalisées au polarimètre. Le paramètre optimisé σ_{opt} est égale à 0.1576. La valeur d'incertitude du paramètre optimisé σ_{opt} est de l'ordre de 10^{-32} . Par conséquent, les écarts entre les fonctions de densité de Rayleigh $f(x, \sigma_{opt})$, $f(x, \sigma_{low})$ et $f(x, \sigma_{upp})$ sont négligeables (Voir Figure 4.4). La précision de cet ajustement s'explique donc par le nombre élevé de classes créées et du nombre de valeur des vitesses de rotation angulaire calculées.

Carwford *et al.* (2013) ont déterminé trois types de vitesses de rotation angulaire $\frac{dS}{dt}$ à partir de l'histogramme présenté par la figure 4.4. Les valeurs inférieures à 0.5 rad/ms définissent les vitesses « lentes » de rotation angulaire des états de polarisation. Les valeurs comprises

entre 0.5 et 1.1 rad/ms décrivent des vitesses « modérées » de rotation angulaire. Enfin, les vitesses rapides de rotation angulaire des états de polarisation définissent des valeurs supérieures à 1.1 rad/ms selon Crawford *et al.* (2013). Le tableau (4.1) présente le résultat du coefficient de corrélation de Pearson R_p et du coefficient de détermination R^2 de l'ajustement de la fonction de densité de Rayleigh sur chaque région de l'histogramme déterminée par une classe particulière de vitesse de rotation angulaire.

Tableau 4.1 Évaluation de l'ajustement de la fonction de densité de Rayleigh $f(x, \sigma_{opt})$ sur les valeurs de l'histogramme $f_{ARR}(x)$ des vitesses de rotation angulaire

Type de vitesse de rotation angulaire (en rad/ms)	R_p	R^2
Lente $\frac{dS}{dt} < 0.5$ rad/ms	0.9958	0.988
Modérée $0.5 < \frac{dS}{dt} < 1.1$ rad/ms	0.9914	0.7923
Rapide $\frac{dS}{dt} > 1.1$ rad/ms	0.6812	-0.3678

Le coefficient de corrélation de Pearson est égal à 0.9958 pour les vitesses lentes de rotation angulaire ce qui confirme l'ajustement de qualité des faibles valeurs de l'histogramme par la fonction de densité de Rayleigh. Le coefficient de détermination R^2 et le coefficient de corrélation de Pearson s'élèvent à 0.9958 et à 0.988 pour les vitesses lentes de rotation angulaire des états de polarisation alors que ces coefficients sont égaux à 0.6812 et à -0.3678 pour la région de l'histogramme qui désigne les vitesses rapides de rotation angulaire des états de polarisation. Le coefficient de corrélation de Pearson R_p et le coefficient de détermination R^2 sont égaux à 0.9914 et à 0.7923 sur les vitesses modérées de rotation angulaire de l'histogramme normalisé (Voir Tableau 4.1), ce qui démontre que les vitesses modérées de rotation angulaire de l'histogramme, comprises entre 0.5 et 1.1 rad/ms, s'écartent significativement des valeurs de la fonction de densité de Rayleigh $f_G(x, \sigma_{opt})$.

L'hypothèse selon laquelle les vitesses de rotation angulaire des états de polarisation puissent être ajustées par la fonction de densité de Rayleigh est seulement valide pour les valeurs de l'histogramme dont les vitesses de rotation angulaire sont inférieures à 0.5 rad/ms (*Voir* Tableau 4.1).

4.3.3 Ajustement de l'histogramme des vitesses de rotation angulaire supérieures à 1.1 rad/ms par la fonction de densité exponentielle

L'objectif de cette étude porte sur l'ajustement de la fonction de densité $f_{ARR}(x)$ des vitesses de rotation angulaire supérieures à 1.1 rad/ms. Crawford *et al.* (2013) montre que seules les vitesses de rotation angulaires comprises entre 1.1 et 2.0 rad/ms semblent pouvoir s'ajuster par une fonction de densité exponentielle en vue de comprendre d'avantage le phénomène statistique à l'origine des vitesses rapides de rotation angulaire des états de polarisation mesurés par le polarimètre. L'équation (4.20) formule la fonction de densité exponentielle $f_E(x, \lambda)$ selon Walck (2007).

$$f_E(x, \lambda) = \lambda \times e^{-\lambda x} \quad (4.20)$$

λ est le paramètre caractéristique de la fonction de densité exponentielle $f_E(x, \lambda)$. La figure 4.5 décrit l'ajustement de la fonction de densité de Rayleigh sur l'histogramme entier des vitesses de rotation angulaire et l'ajustement de la fonction de densité exponentielle sur les vitesses de rotation angulaire comprises entre 1.1 et 2.0 rad/ms par l'application de la méthode d'ajustement décrite lors de la seconde étape de calcul (*Voir* section 4.3.1.2).

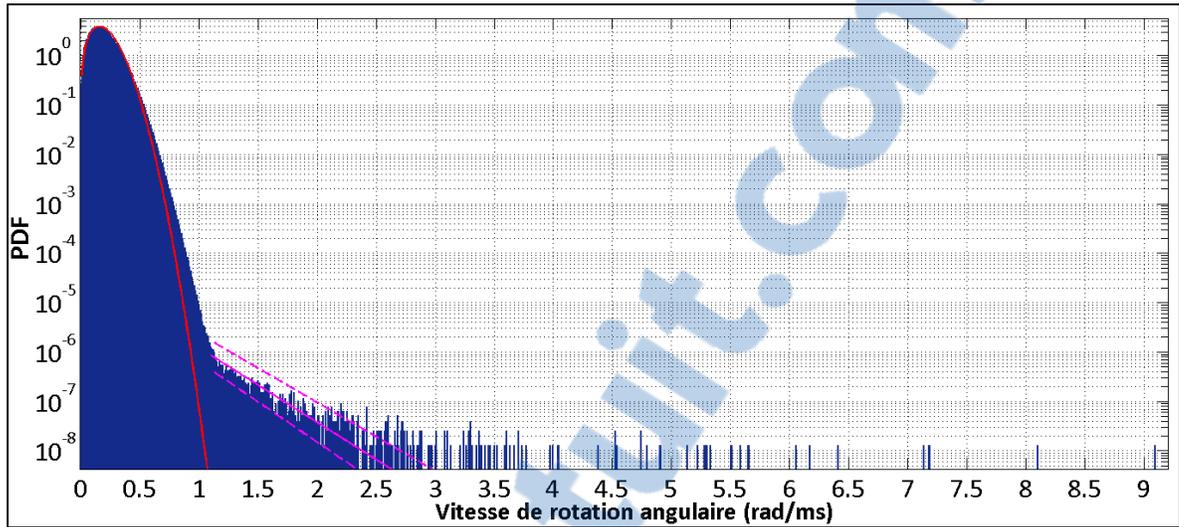


Figure 4.5 Ajustement de l'histogramme normalisé des vitesses de rotation angulaire par la fonction de densité de Rayleigh et par la fonction de densité exponentielle

2011 valeurs de vitesse de rotation angulaire supérieures à 1.1 rad/ms ont été mesurées sur le lien optique du réseau Verizon pour toutes les périodes de mesures réalisées au polarimètre. Le tableau 4.2 décrit les valeurs des paramètres λ , du coefficient de corrélation de Pearson R_p et du coefficient de détermination R^2 obtenues après l'ajustement de l'histogramme des vitesses rapides de rotation angulaire par la fonction de densité exponentielle $f_E(x, \lambda)$.

Tableau 4.2 Évaluation de l'ajustement de la fonction $f_E(x, \lambda)$ sur l'histogramme des vitesses rapides de rotation angulaire comprises entre 1.1 rad/ms et 2.0 rad/ms

Valeurs de vitesse rapide de rotation angulaire	λ		R_p	R^2
$1.1 \leq \frac{dS}{dt} \leq 2.0$ (rad/ms)	λ_{low}	3,22	0.9543	-0,2558
	λ_{opt}	3,50	0.9588	0.9191
	λ_{upp}	3,78	0.9619	0.6617

Les paramètres λ_{opt} , λ_{low} et λ_{upp} des ajustements de la fonction de distribution exponentielle présentés par la figure 4.5 sont égaux à 3.22, 3.50 et à 3.78 respectivement. L'écart relatif entre le paramètre optimisé λ_{opt} et les paramètres d'incertitude λ_{low} et λ_{upp} à un intervalle de

confiance de 95 % est égale à 8 %, ce qui signifie que les écarts entre les fonctions de densité exponentielle $f_E(x, \lambda_{opt})$, $f_E(x, \lambda_{low})$, $f_E(x, \lambda_{upp})$ ne sont pas négligeables. La valeur élevée de cet écart est causée par le faible nombre de vitesses rapides de rotation angulaire calculées. Le coefficient de corrélation de Pearson R_p et le coefficient de détermination R^2 sont égaux à 0.9588 et à 0.9191. Ceci démontre que la fonction de densité exponentielle $f_E(x, \lambda_{opt})$ ajuste de manière satisfaisante les vitesses de rotation angulaire comprises entre 1.1 et 2.0 rad/ms. Crawford *et al.* (2013) valide l'hypothèse selon laquelle les vitesses rapides de rotation angulaire des états de polarisation puissent être ajustées par une fonction de densité exponentielle après l'analyse de ces résultats. Cette analyse a permis d'identifier deux phénomènes physiques à l'origine des vitesses de rotation angulaire des états de polarisation par l'ajustement de la fonction de densité de Rayleigh $f_G(x, \sigma_{opt})$ sur les vitesses de rotation angulaire inférieures à 0.5 rad/ms et par l'ajustement de la fonction de densité exponentielle $f_E(x, \lambda_{opt})$ sur les vitesses de rotation angulaire supérieures à 1.1 rad/ms.

4.3.4 Ajustement de l'histogramme entier des vitesses de rotation angulaire par une fonction de densité double

4.3.4.1 Présentation de la fonction de densité double f_{GEI} sur l'histogramme entier des vitesses de rotation angulaire des états de polarisation

Après analyse des ajustements effectués dans la section 4.3.2 et dans la section 4.3.3, Crawford *et al.* (2013) ont décidé que la prochaine étape serait l'ajustement de l'histogramme entier des vitesses de rotation angulaire des états de polarisation par une fonction de densité double appelée f_{GEI} constituée de la fonction de densité de Rayleigh $f_G(x, \sigma)$ et de la fonction de densité exponentielle $f_E(x, \lambda)$ précédemment ajustées. Crawford *et al.* (2013) ont développé la fonction de densité double f_{GEI} qui combine les deux phénomènes physiques à l'origine des vitesses de rotation angulaire des états de polarisation mesurés par le polarimètre sur le lien aérien du réseau Verizon.

Tout d'abord, la fonction de densité exponentielle $f_E(x, \lambda_{opt})$ qui décrit le phénomène physique à l'origine des vitesses rapides de rotation angulaire des états de polarisation mesurés sur le lien aérien a été développé. Crawford *et al.* (2013) montre que pour chaque vitesse de rotation angulaire appartenant à la fonction de densité de Rayleigh $f_G(x, \sigma)$, la probabilité qu'une vitesse de rotation angulaire « exponentiellement distribuée » se produise est égale à une probabilité ρ . Ceci est équivalent à multiplier la fonction de densité exponentielle $f_E(x, \lambda)$ par une fonction indicatrice I égale à 1 pour une probabilité de ρ ou égale à 0 pour une probabilité de $(1-\rho)$. Crawford *et al.* (2013) ont développé la formule de la fonction f_{EI} définissant la fonction de densité exponentielle à un seuil de probabilité ρ selon l'équation (4.21).

$$f_{EI}(x, \lambda, \rho) = \rho \times \lambda \times e^{-\lambda x} + (1 - \rho) \times \delta(x) \quad (4.21)$$

La fonction f_{EI} définit le phénomène physique à l'origine des vitesses rapides de rotation angulaire des états de polarisation. λ est le paramètre caractéristique de la fonction de densité exponentielle $f_E(x, \lambda)$. ρ correspond à la probabilité où se produit le phénomène de distribution exponentielle. Enfin, δ est la distribution de Dirac. L'intégrale des probabilités de la fonction f_{EI} est égale à 1. Pour cette raison, la fonction f_{EI} est une fonction de densité de probabilité. La convolution de la fonction de densité $f_{EI}(x, \lambda, \rho)$ et de la fonction de densité de Rayleigh $f_G(x, \sigma)$ a été développée en vue de l'obtention de la fonction de densité double f_{GEI} . L'équation (4.22) formule le calcul de la fonction f_{GEI} dépendante de la fonction de convolution $f_{GE}(x, \lambda, \sigma)$, de la fonction de densité de Rayleigh $f_G(x, \sigma)$ et de la fonction de densité exponentielle $f_E(x, \lambda)$.

$$\begin{aligned} f_{GEI}(x, \rho, \lambda, \sigma) &= f_G(x - y, \sigma) * f_{EI}(x, \lambda, \rho) \\ &= \rho \times f_{GE}(x, \lambda, \sigma) + (1 - \rho) \times f_G(x, \sigma) \\ &= \rho \times (f_G(x - y, \sigma) * f_E(x, \lambda)) + (1 - \rho) \times f_G(x, \sigma) \end{aligned} \quad (4.22)$$

Crawford *et al.* (2013) a démontré que la fonction f_{GEI} peut être développée mathématiquement à l'aide de la fonction d'erreur *erf*. L'équation (4.23) formalise le calcul de la fonction de densité double $f_{GEI}(x, \rho, \lambda, \sigma)$ qui résulte de la démonstration mathématique dépendante de la fonction d'erreur *erf* et des paramètres ρ , λ et σ selon Crawford *et al.* (2013).

$$f_{GEI}(x, \rho, \lambda, \sigma) = \rho \lambda e^{\frac{\lambda^2 \sigma^2}{2}} e^{-\lambda x} \left[\left[\sqrt{\frac{\pi}{2}} \operatorname{erf} \left[\frac{x - \lambda \sigma}{\sqrt{2}} \right] - e^{-\left(\frac{x - \lambda \sigma}{2\sigma}\right)^2} \right] \right. \quad (4.23)$$

$$\left. - \left[\sqrt{\frac{\pi}{2}} \lambda \sigma \operatorname{erf} \left[\frac{-\lambda \sigma}{\sqrt{2}} \right] - e^{-\frac{\lambda^2 \sigma^2}{2}} \right] \right] + (1 - \rho) \times \frac{x}{\sigma^2} e^{-\left(\frac{x^2}{2\sigma^2}\right)}$$

La fonction de densité exponentielle $f_{EI}(x, \lambda, \rho)$ et la fonction de densité de Rayleigh $f_G(x, \sigma)$ étant normalisé, la fonction f_{GEI} qui résulte de la convolution des deux précédentes fonctions de densité est donc aussi normalisée, c'est-à-dire que son intégrale est égale à 1.

4.3.4.2 Méthode et analyse de l'ajustement de l'histogramme entier des vitesses de rotation angulaire par la fonction de densité double f_{GEI}

Ma contribution porte sur l'utilisation et la configuration des fonctions offertes par le module *NonlinearModel* de Matlab pour l'ajustement de l'histogramme des vitesses de rotation angulaire par la fonction de densité double f_{GEI} . Les fonctions Matlab du module *NonlinearModel* sont les mêmes que dans la méthode d'ajustement décrite dans la section 4.3.1.2. La figure 4.6 formalise la méthode d'ajustement de la fonction de densité f_{GEI} sur différentes régions de l'histogramme des vitesses de rotation angulaire des états de polarisation.

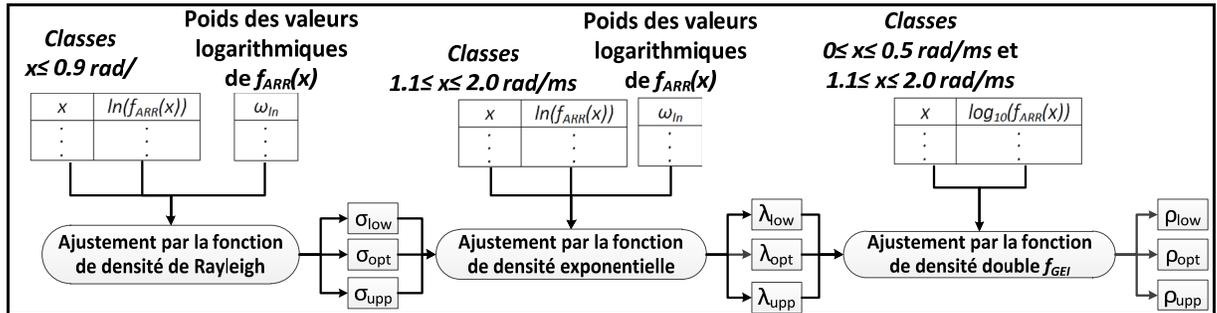


Figure 4.6 Méthode d'ajustement de la fonction de densité double f_{GEI} par l'application du module *NonlinearModel* de la *Statistics Toolbox* de Matlab

La méthode d'ajustement de l'histogramme entier par la fonction de densité double f_{GEI} comporte trois étapes d'ajustement (Voir Figure 4.6). L'objectif de chaque étape d'ajustement est l'obtention d'un des paramètres optimisés ρ_{opt} , λ_{opt} et σ_{opt} et de son incertitude à un intervalle de confiance de 95%. Les paramètres σ_{opt} , σ_{low} , σ_{upp} sont calculés par l'application de la méthode d'ajustement décrite à la section 4.3.1.2 sur les vitesses de rotation angulaire inférieures à 0.9 rad/ms lors de la première étape d'ajustement. La seconde étape d'ajustement porte sur le calcul des paramètres λ_{opt} , λ_{low} et λ_{upp} définissant les paramètres caractéristiques des fonctions de densité exponentielle $f_E(x, \lambda_{opt})$, $f_E(x, \lambda_{low})$, $f_E(x, \lambda_{upp})$ telle que cela est montré sur la figure 4.6. Les paramètres λ_{opt} , λ_{low} et λ_{upp} ont été calculés par l'application de la méthode d'ajustement sur l'histogramme des vitesses de rotation angulaire comprises entre 1.1 rad/ms et 2.0 rad/ms tel que proposée par Crawford *et al.* (2013). Les paramètres λ_{opt} et σ_{opt} qui résultent des deux premières étapes d'ajustement sont identiques aux valeurs obtenues dans les analyses décrites aux sections 4.3.2 et 4.3.3. Enfin, l'ajustement de la fonction de densité double f_{GEI} a été à la fois appliquée sur les vitesses de rotation angulaire inférieures à 0.5 rad/ms et à la fois sur les vitesses rapides de rotation angulaire comprises entre 1.1 rad/ms et 2.0 rad/ms en ne faisant seulement varier que le paramètre ρ lors de la troisième étape d'ajustement (Voir Figure 4.6). La figure 4.7 présente l'ajustement de la fonction de densité double f_{GEL} par l'application de la méthode d'ajustement décrite par la figure 4.6.

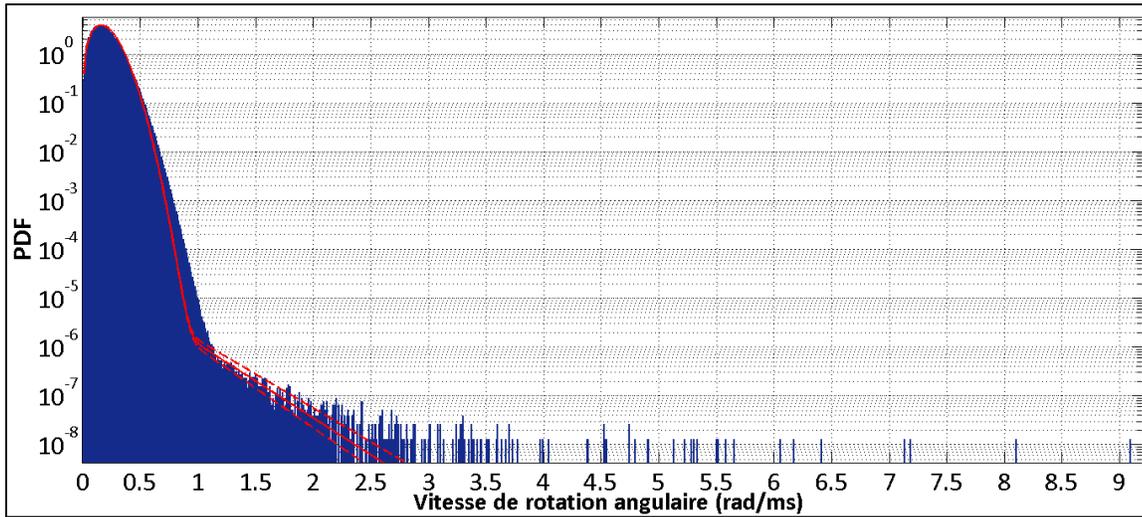


Figure 4.7 Ajustement de l'histogramme entier des vitesses de rotation angulaire des états de polarisation par la fonction f_{GEI}

La valeur du paramètre σ_{opt} est égale à 0.1574 avec une incertitude Δ de 10^{-32} . Les paramètres λ_{low} , λ_{opt} , λ_{upp} sont égaux à 3.7506, 3.4747 et à 3.1989 respectivement et les paramètres ρ_{low} , ρ_{opt} , ρ_{upp} sont égaux à 4.58×10^{-6} , à 4.86×10^{-6} et à 5.20×10^{-6} respectivement. Les écarts relatifs des paramètres λ et ρ sont égaux respectivement à 7.9 %, et à 5.8 %. La valeur de ces écarts signifie que l'incertitude relative à la fonction $f_{GEI}(x, \rho, \lambda_{opt}, \sigma_{opt})$ est quasi négligeable lors de l'évaluation de cet ajustement compte tenu du nombre de classes (plus de 900) et du nombre de valeur de vitesses de rotation angulaire calculées. La valeur du coefficient de corrélation de Pearson R_p et la valeur du coefficient de détermination R^2 sont égales à 0.9975 à 0.9949 respectivement. La fonction de densité f_{GEI} propose donc un modèle d'ajustement satisfaisant sur les vitesses lentes et sur les vitesses rapides de rotation angulaire des états de polarisation. Ceci révèle que deux phénomènes physiques semblent être à l'origine des vitesses de rotation angulaire des états de polarisation. Cette analyse a permis de caractériser la nature et le comportement des vitesses de rotation angulaire des états de polarisation mesurés par le polarimètre sur la liaison du réseau Verizon.

4.4 Caractérisation des paramètres des effets transitoires de polarisation

Cette section porte uniquement sur l'analyse et la caractérisation des vitesses rapides de rotation angulaire des états de polarisation. Le seuil des vitesses rapides de rotation angulaire des états de polarisation a été défini par Reimer *et al.* (2013) à 0.9 rad/ms à partir de l'analyse des résultats précédemment obtenus (*Voir* section 4.3.4). Ce seuil définit une valeur de probabilité proche de 10^{-5} sur l'histogramme des vitesses de rotation angulaire des états de polarisation mesurés sur la liaison aérienne du réseau Verizon (*Voir* Figure 4.7). Crawford *et al.* (2013) montrent que c'est à partir de ce seuil que la fonction de densité $f_{ARR}(x)$ des vitesses de rotation angulaire des états de polarisation peut être ajustée au moyen d'une fonction de densité exponentielle $f_E(x, \lambda)$ (*Voir* Figure 4.7).

Reimer *et al.* (2013) indique que plusieurs vitesses rapides de rotation angulaire peuvent se produire consécutivement sur une période de mesure égale à 10 ms. Ce phénomène est appelé « effet transitoire de polarisation ». Un effet transitoire de polarisation se produit lorsqu'une ou plusieurs vitesses de rotation angulaire supérieures à 0.9 rad/ms sont mesurées sur une durée de 10 ms.

Cette étude a pour objectif la caractérisation des paramètres des effets transitoires de polarisation mesurés sur le lien optique du réseau Verizon après l'application d'une méthode de calcul et d'une méthode d'analyse décrites en première et en seconde partie de cette section.

4.4.1 Méthode de calcul des paramètres des effets transitoires de polarisation

Trois paramètres ont été définis par Reimer *et al.* (2013) en vue de la caractérisation des effets transitoires de polarisation : la durée, l'élasticité et la vitesse maximale de rotation angulaire calculées durant chaque effet transitoire de polarisation mesuré. Crawford *et al.* (2013) introduit un quatrième paramètre pour la caractérisation des effets transitoires de polarisation : l'intervalle de temps séparant deux effets transitoires consécutifs dans un même fichier de mesure.

Ma principale contribution pour la caractérisation des effets transitoires de polarisation porte sur l'intégration de la méthode de calcul qui identifie dans un premier temps le début et la fin d'un effet transitoire et qui calcule, dans un second temps, les paramètres caractéristiques des effets transitoires de polarisation mesurés. La figure 4.8 formalise la méthode de calcul en vue de l'obtention des différents paramètres caractéristiques des effets transitoires de polarisations mesurés sur le lien optique du réseau Verizon.

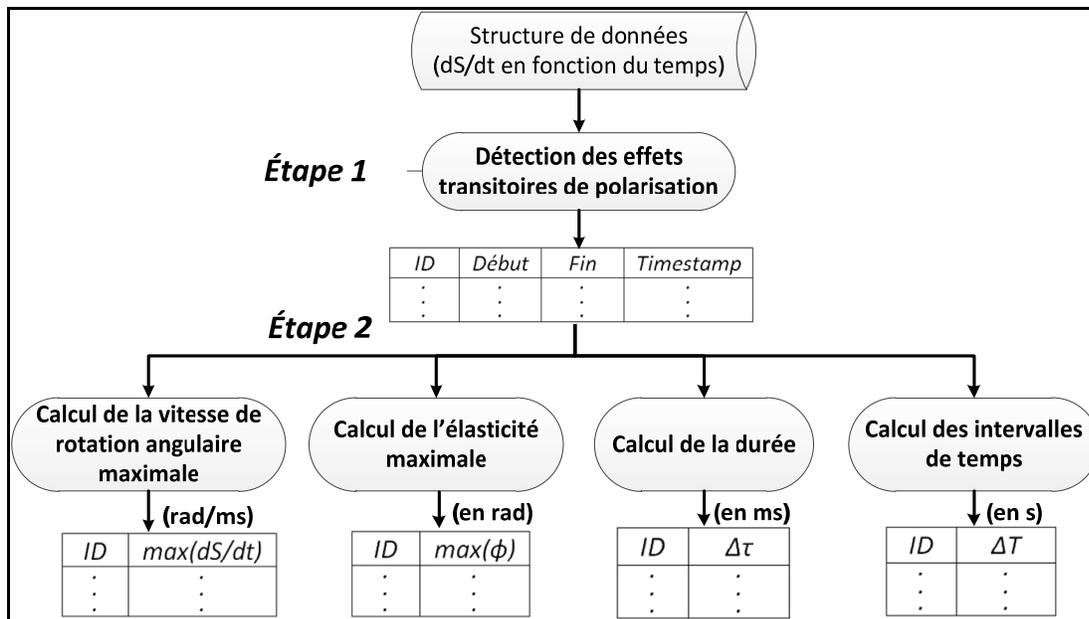


Figure 4.8 Méthode de calcul des paramètres caractéristiques des effets transitoires de polarisation

Deux étapes de calcul constituent ainsi la méthode appliquée sur les vecteurs Stokes (S_1 , S_2 , S_3) durant toutes les périodes de mesure réalisées au polarimètre sur la liaison aérienne du réseau Verizon (Voir Figure 4.8). Pour cette raison, deux scripts Matlab sont développés pour l'intégration de chacune des deux étapes de calcul.

La première étape de calcul a pour objectif l'automatisation de la détection des effets transitoires de polarisation par le développement d'un premier script Matlab (Voir ANNEXE

XLII, p. 233). La position de début et la position de fin des effets transitoires de polarisation ont été ajustées afin que des durées réalistes puissent être attribuées aux effets transitoires de polarisation caractérisés par une seule vitesse rapide de rotation angulaire durant 10 ms. En effet, un effet transitoire de polarisation commence lorsque la vitesse de rotation angulaire des états de polarisation excède 0.45 rad/ms (soit la moitié du seuil désignant une vitesse rapide de rotation angulaire) juste avant la première vitesse rapide de rotation angulaire mesurée et se termine lorsque la vitesse de rotation angulaire devient inférieure à 0.45 rad/ms après la dernière fluctuation rapide mesurée. Les vitesses rapides de rotation angulaire séparées par une durée supérieure à 10 ms sont considérées comme des effets transitoires de polarisation indépendants selon Reimer *et al.* (2013). Le résultat de l'application de ce premier script se présente sous la forme d'une matrice constituée des identifiants des fichiers de mesures où sont mesurés les effets transitoires de polarisation, des valeurs de position de début et des valeurs de position de fin des effets transitoires de polarisation mesurés et du temps de mesure (au format Matlab) auquel débute l'effet transitoire de polarisation (*Voir* Figure 4.8).

Le second script a pour objectif le calcul des paramètres caractéristiques des effets transitoires de polarisation localisés par l'application du précédent script Matlab (*Voir* ANNEXE XLIII, p. 235). La durée désignée par la variable $\Delta\tau$ de chaque effet transitoire de polarisation a été calculée simplement par la soustraction de son temps de fin t_{end} et de son temps de début t_{init} . L'équation (4.24) formule le calcul de la durée d'un effet transitoire de polarisation mesuré.

$$\Delta\tau = t_{fin} - t_{init} \quad (4.24)$$

La vitesse maximale de rotation angulaire, indiquée par la variable ARR_{max} , a par la suite été calculée durant chaque effet transitoire de polarisation. L'équation (4.25) formalise le calcul de la vitesse maximale de rotation angulaire des états de polarisation lorsqu'un effet transitoire a été mesuré. τ est la période d'échantillonnage et indique de cette manière la durée séparant deux vecteurs Stokes mesurés au polarimètre soit 50 μ s.

$$ARR_{max} = \max_{init \leq t < fin} \left(\frac{d\vec{s}}{dt} \right) = \max_{init \leq t < fin} \left(\frac{\text{acos}(\overrightarrow{s(t)} \cdot \overrightarrow{s(t + \tau)})}{\tau} \right) \quad (4.25)$$

Nelson (2012) énonce le terme de l'« élasticité » par le calcul de l'angle de rotation entre le début et la fin d'un effet transitoire de polarisation mesuré avec un polarimètre en mode « rafale ». Dans le cadre de l'expérience de *monitoring* réalisée sur la liaison optique cohérente du réseau Verizon, Reimer *et al.* (2013) définit l'« élasticité » par le calcul de l'angle maximal entre le vecteur Stokes $\overrightarrow{s(t_{init})}$ indiquant le début de l'effet transitoire de polarisation et le vecteur Stokes correspondant au maximum de déplacement sur la sphère de Poincaré durant l'effet transitoire de polarisation. L'équation (4.26) formule le calcul de l'élasticité désignée par la variable ϕ_{max} pour chaque effet transitoire de polarisation selon Reimer *et al.* (2013).

$$\phi_{max} = \max_{init \leq t < fin} (\text{acos}(s_1(t_{init}) \cdot s_1(t) + s_2(t_{init}) \cdot s_2(t) + s_3(t_{init}) \cdot s_3(t))) \quad (4.26)$$

Les intervalles de temps ΔT entre deux effets transitoires adjacents ont été finalement calculés. Crawford *et al.* (2013) définit l'intervalle de temps par la valeur d'une durée, désignée par la variable ΔT , qui sépare le début d'un effet transitoire mesuré et la fin du précédent effet transitoire dans un même fichier de mesures avec le polarimètre. L'équation (4.27) formalise le calcul des intervalles de temps ΔT entre le début de l'effet transitoire de polarisation à l'indice $t_{init}(i)$ et la fin de l'effet transitoire de polarisation $t_{fin}(i - 1)$ précédemment mesuré.

$$\Delta T = t_{init}(i) - t_{fin}(i - 1) \quad (4.27)$$

Le résultat de l'application de ce second script Matlab se présente sous la forme de quatre matrices constituées chacune des valeurs d'un des paramètres caractéristiques des effets transitoires de polarisation mesurés (*Voir* Figure 4.8).

4.4.2 Méthode d'analyse statistique des paramètres caractéristiques des effets transitoires de polarisation

Deux outils d'analyses statistiques ont été développés sous la forme de deux scripts Matlab constituant chacun une étape d'analyse pour la caractérisation des paramètres des effets transitoires de polarisation mesurés sur le lien aérien du réseau Verizon.

Les fonctions de densité des valeurs d'élasticité $f_{\varphi_{max}}(x)$, des valeurs de durée $f_{\Delta\tau}(x)$, et des valeurs d'intervalle de temps $f_{\Delta T}(x)$ sont calculées dans le premier script Matlab lors de la première étape d'analyse. La méthode de calcul appliquée est identique à celle développée dans la section 4.3.1.1.

Crawford *et al.* (2013) propose de calculer deux histogrammes constitués de deux variables conjointes relatives aux paramètres des effets transitoires de polarisation. Le second outil d'analyse porte donc sur le calcul de fonctions de probabilité conjointes de deux variables représentées par la variable $F_{(X,Y)}$. Ces fonctions de probabilités sont la combinaison des fréquences relatives aux valeurs d'occurrence N_x et N_y associées aux variables aléatoires X et Y et comporte les valeurs de probabilité estimées telles que $\hat{p}(X = x, Y = y)$. L'équation (4.28) formule le calcul de la fonction de probabilité conjointe de deux variables $F_{(X,Y)}$ selon Navidi (2011, p.129).

$$\begin{aligned}
 F_{(X,Y)}(x_1 < X \leq x_2 \text{ et } y_1 < Y \leq y_2) & \quad (4.28) \\
 &= p(x_1 < X \leq x_2 | y_1 < Y \leq y_2) \times p(y_1 < Y \leq y_2) \\
 &= p(y_1 < Y \leq y_2 | x_1 < X \leq x_2) \times p(x_1 < X \leq x_2)
 \end{aligned}$$

L'équation (4.29) décrit le calcul des valeurs de la fonction de probabilité $F_{(X,Y)}$ pour toutes les valeurs de la variable Y comprises entre les classes y_1 et y_2 sachant que les valeurs de la variable X sont situées entre les classes x_1 et x_2 . Deux fonctions de probabilité conjointes de deux variables relatives aux paramètres caractéristiques des effets transitoires de polarisation sont calculées par l'application d'un second script Matlab. La première fonction calculée est

appelée $F_{(ARR_m, \phi_m)}$ et combine les valeurs maximales de vitesse de rotation angulaire ARR_{max} et des valeurs d'élasticité ϕ_{max} (Voir ANNEXE XLIV, p. 239). La seconde fonction de probabilité conjointe de deux variables appelée $F_{(ARR_m, \Delta\tau)}$ associe les valeurs maximales de vitesse de rotation angulaire ARR_{max} et les valeurs de durée $\Delta\tau$ des effets transitoires de polarisation mesurés (Voir ANNEXE XLV, p. 241).

Enfin, ces deux outils d'analyse optimisent l'affichage des résultats obtenus en fin de script par la configuration de fonctions de visualisation des données (Voir ANNEXE XLVI, p. 243).

4.4.3 Analyse des valeurs d'élasticité des effets transitoires de polarisation

La figure 4.9 décrit l'histogramme normalisé $f_{\phi_{max}}(x)$ des valeurs d'élasticité (avec une résolution égale à 0.0125 rad) des effets transitoires de polarisation mesurés sur la liaison aérienne du réseau Verizon avec le polarimètre.

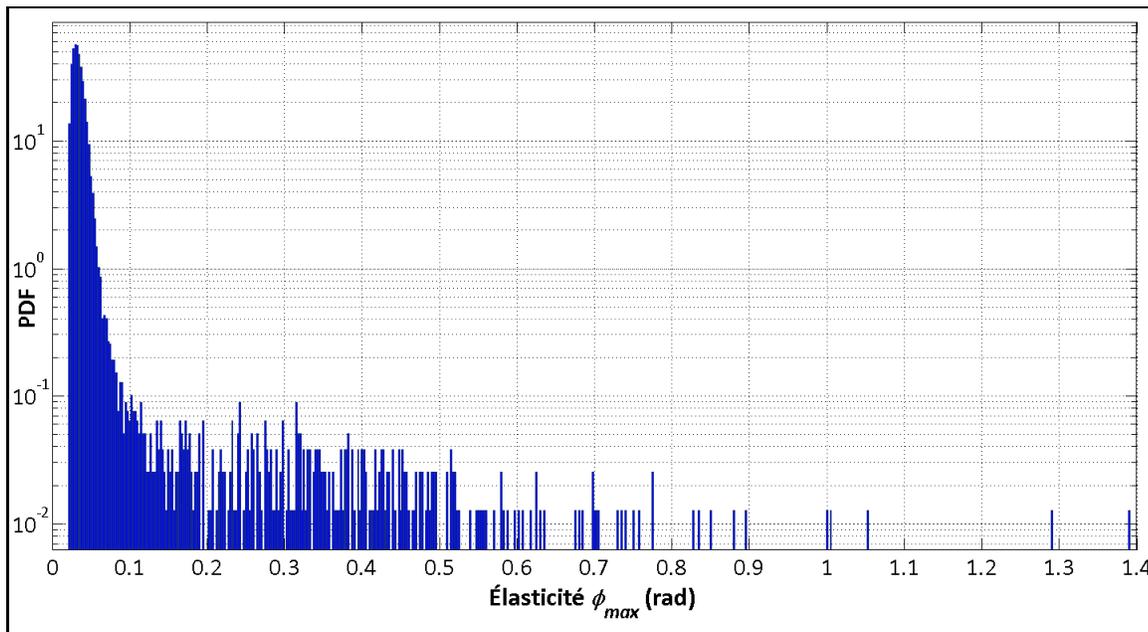


Figure 4.9 Histogramme des valeurs d'élasticité des effets transitoires de polarisation

La figure 4.9 décrit la fonction de densité $f_{\phi_{max}}(x)$ des valeurs d'élasticité pour chaque effet transitoire de polarisation mesuré. 31540 effets transitoires de polarisation ont ainsi été détectés pour toutes les périodes de mesure réalisés au polarimètre sur le lien aérien du réseau Verizon après l'application de la méthode calcul des paramètres des effets transitoires de polarisation (*Voir* section 4.4.1).

La moyenne et l'écart type des valeurs d'élasticité sont égaux à 0.0392 rad et à 0.0402 rad respectivement. Plus de 99% des effets transitoires de polarisation sont caractérisées par des valeurs d'élasticité inférieures à 0.05 rad ce qui démontre que celles-ci sont relativement faibles. Pourtant, il existe des effets transitoires de polarisation caractérisés par des valeurs d'élasticité très supérieures à la moyenne. Reimer *et al.* (2013) montre la nécessité d'étudier les valeurs d'élasticité supérieures à 15° (soit 0.26 rad). 234 valeurs d'élasticité sont supérieures à 0.26 rad pour toutes les périodes de mesures réalisées au polarimètre. La probabilité que se produit une valeur d'élasticité supérieure à 15° (soit 0.26 rad) lorsqu'un effet transitoire de polarisation est mesuré est égale à 7×10^{-3} . La plus grande valeur d'élasticité observée durant 14 jours de mesure est égale à 1.39 rad ($\sim 80^\circ$).

La fonction de probabilité $F(ARR_{max}, \phi_{max})$ constituée des valeurs maximales de vitesse de rotation angulaire ARR_{max} et des valeurs d'élasticité ϕ_{max} a été calculée par l'application de la méthode d'analyse précédemment décrite (*Voir* section 4.4.2). Les tailles des classes correspondant aux vitesses maximales de rotation angulaire ARR_{max} et aux valeurs d'élasticité ϕ_{max} sont égales à 0.1 rad/ms et à 0.0125 rad respectivement. La figure 4.10 présente l'histogramme $F(ARR_{max}, \phi_{max})$ des variables ARR_{max} et ϕ_{max} des effets transitoires de polarisation calculés pour toutes les périodes de mesures réalisées avec le polarimètre sur la liaison majoritairement aérienne du réseau Verizon.

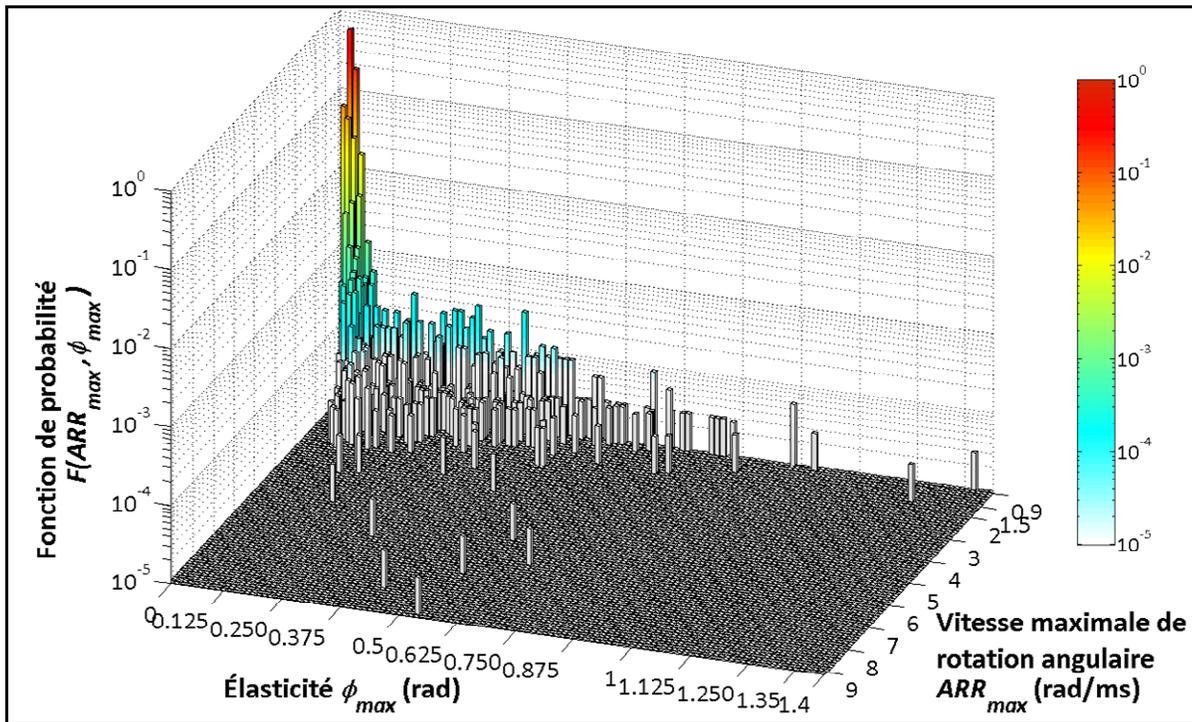


Figure 4.10 Histogramme $F(ARR_{max}, \phi_{max})$ des vitesses maximales de rotation angulaire ARR_{max} et des élasticités ϕ_{max} des effets transitoires de polarisation

88 % des 31540 effets transitoires de polarisation sont caractérisés par des vitesses maximales de rotation angulaire inférieures à 1.0 rad/ms et par des valeurs d'élasticité inférieures ou égales à 0.05 rad. La valeur maximale des vitesses maximales de rotation angulaire calculées est égale à 9.09 rad/ms (Voir Figure 4.10). La valeur d'élasticité correspondant à cette vitesse de rotation angulaire est égale à 0.53 rad. La valeur maximale de vitesse de rotation angulaire correspondant à l'effet transitoire de polarisation le plus élastique est proche du seuil de 0.9 rad/ms.

Ce résultat montre que les valeurs d'élasticité supérieures à 0.26 rad ne présentent aucune corrélation avec les vitesses maximales de rotation angulaire supérieures à 2.0 rad/ms (Voir Figure 4.10). En effet, 123 des 234 effets transitoires de polarisation caractérisés par des valeurs d'élasticité supérieures à 0.26 rad correspondent à des vitesses maximales de rotation angulaire proches de 0.9 rad/ms.

4.4.4 Analyse des valeurs de durée des effets transitoire de polarisation

La figure 4.11 présente l'histogramme normalisé $f_{\Delta\tau}(x)$ des valeurs de durée des effets transitoires de polarisation sur toutes les périodes de mesures réalisées au polarimètre avec une taille pour chaque classe égale à la période d'échantillonnage du polarimètre soit 50 μs .

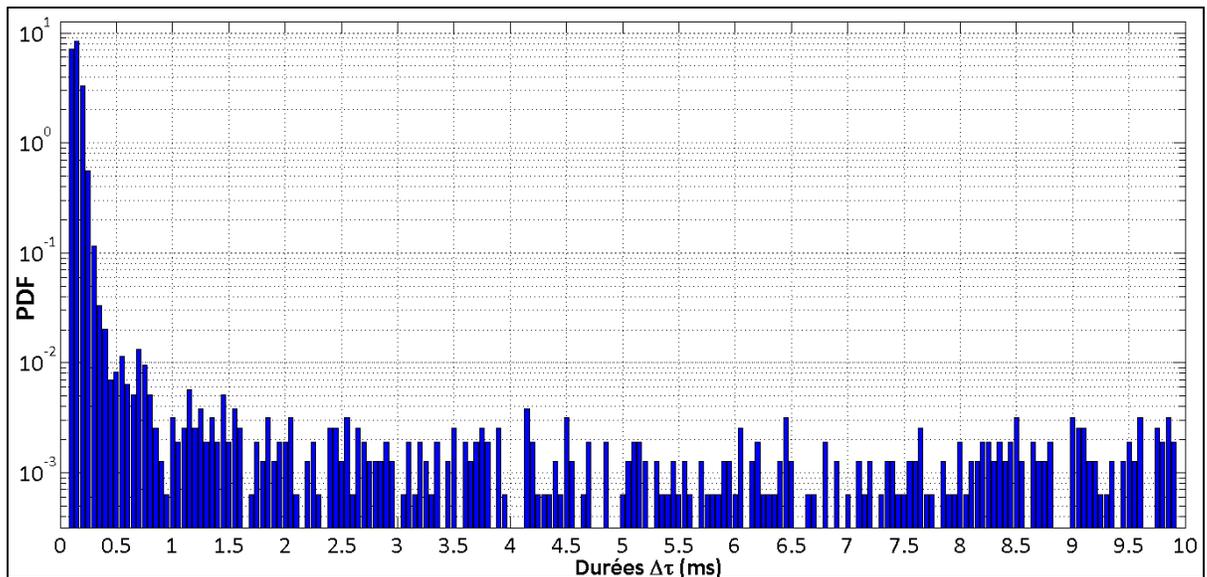


Figure 4.11 Histogramme normalisé $f_{\Delta\tau}(x)$ des valeurs de durée des effets transitoires de polarisation

La moyenne et l'écart type de la durée des effets transitoires de polarisation sont égaux à 0.21 ms et à 0.64 ms respectivement. Plus de 90% des effets transitoires ont une durée inférieure à 0.25 ms ce qui signifie que les effets transitoires de polarisation sont très souvent constitués que d'une seule valeur de vitesse rapide de rotation angulaire. Pourtant, des effets de transitoires peuvent être caractérisés par une durée proche de la limite de la fenêtre temporelle de 10 ms proposée par Reimer *et al.* (2013). 472 effets transitoires de polarisation ont été caractérisés pour une durée supérieure ou égale à 0.5 ms. 1.5 % des effets transitoires comportent 10 valeurs de vitesse rapide de rotation angulaire mesurées par le polarimètre sur une durée de 10 ms.

Reimer *et al.* (2013) montrent que les effets transitoires de polarisation peuvent être classés en deux catégories. La première catégorie définit les effets transitoires caractérisés par une courte durée inférieure ou égale à 0.5 ms. Une seule vitesse rapide de rotation angulaire constitue ce type d'effet transitoire de polarisation. La seconde catégorie définit les effets transitoires de plus longue durée où de nombreuses rafales de vitesses rapides de rotation angulaire peuvent se produire. Ce type d'effet transitoire définit des activités intenses en polarisation.

La fonction de probabilité $F(ARR_{max}, \Delta\tau)$ constituée des vitesses maximales de rotation angulaire ARR_{max} et des durées $\Delta\tau$ de chaque effet transitoire de polarisation a été calculée par l'application de la méthode d'analyse décrite à la section 4.4.2. La figure 4.12 décrit l'histogramme $F(ARR_{max}, \Delta\tau)$ des valeurs de vitesse maximale de rotation angulaire ARR_{max} et des valeurs de durée $\Delta\tau$ mesurées pour chaque effet transitoire de polarisation.

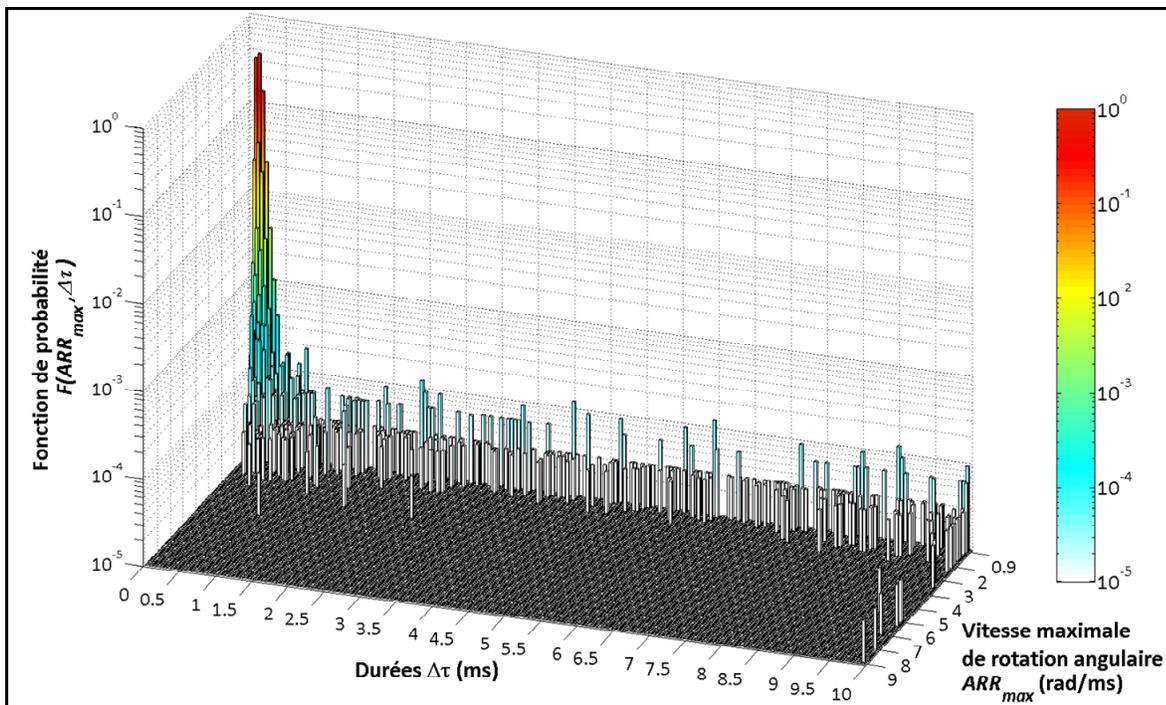


Figure 4.12 Histogramme $F(ARR_{max}, \Delta\tau)$ des vitesses maximales de rotation angulaire ARR_{max} et des durées $\Delta\tau$ des effets transitoires de polarisation

Les effets transitoires de polarisation sont majoritairement caractérisés par de courtes durées et par des valeurs maximales de vitesse de rotation angulaire proches de 0.9 rad/ms (*Voir* Figure 4.12). En effet, 88 % des effets transitoires de polarisation possèdent une durée inférieure à 0.5 ms et une vitesse maximale de rotation angulaire inférieure à 1.0 rad/ms. Les vitesses maximales de rotation angulaire ARR_{max} des états de polarisation fluctuent de 0.9 à 9.09 rad/ms pour les effets transitoires de polarisation caractérisés par une durée supérieure ou égale à 0.5 ms. Pour cette raison, la corrélation entre la durée et la vitesse maximale de rotation angulaire est impossible à déterminer par la figure 4.12. Pourtant, on observe que la durée d'un effet transitoire de polarisation est toujours supérieure à 0.5 ms lorsque la vitesse maximale de rotation angulaire est supérieure à 2.5 rad/ms (*Voir* Figure 4.12), ce qui démontre que les activités intenses en polarisation sont caractérisées par de longues durées. Seulement 1.5 % des effets transitoires se caractérisent par une activité intense en polarisation, elle-même déterminée par une durée supérieure à 0.5 ms. La probabilité que se produit une activité intense en polarisation lorsqu'une vitesse rapide de rotation angulaire est mesurée est de l'ordre de 10^{-4} .

4.4.5 Analyse des valeurs d'intervalle de temps entre deux effets transitoires de polarisation adjacents

L'histogramme normalisé des intervalles de temps ΔT entre deux effets transitoires de polarisation adjacents, correspondant à la fonction de densité de probabilité $f_{\Delta T}(x)$, a été calculé et ajusté par l'application de la méthode de calcul et d'ajustement décrite dans la section 4.3.1 dans un unique script Matlab (*Voir* ANNEXE XLVII, p. 247).

Crawford *et al.* (2013) propose l'ajustement de la fonction de densité des intervalles de temps $f_{\Delta T}(x)$ par la fonction de densité exponentielle $f_E(x, \lambda)$. La figure 4.13 présente l'ajustement de la fonction de densité $f_{\Delta T}(x)$ par la fonction de densité exponentielle au moyen d'une échelle logarithmique.

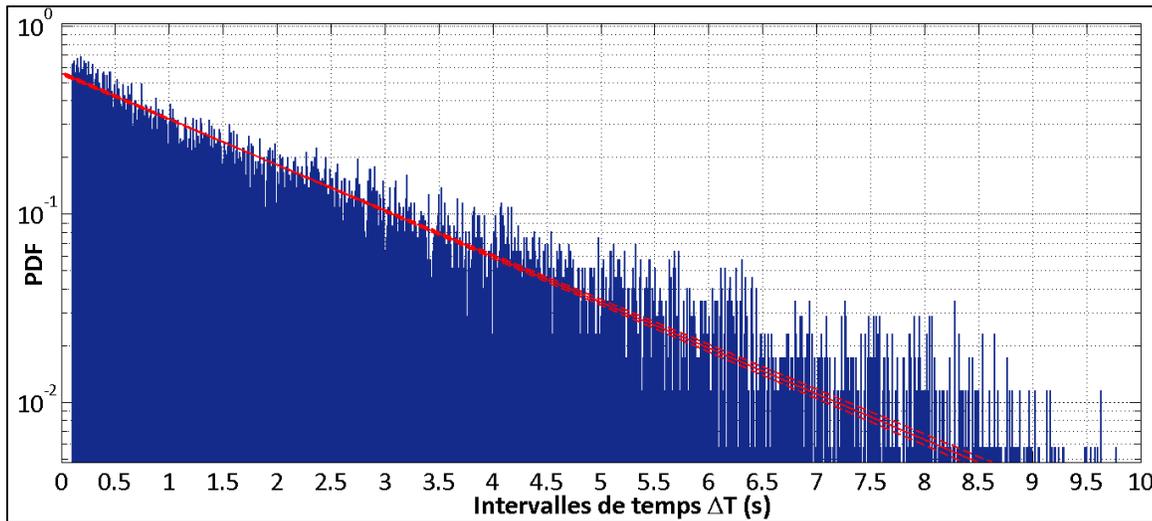


Figure 4.13 Ajustement de la fonction des intervalles de temps $f_{\Delta T}(x)$ par la fonction de densité exponentielle $f_E(x, \lambda)$ à l'aide d'une échelle logarithmique

La limite des intervalles de temps est égale à la durée d'un fichier de mesures avec le polarimètre soit 10 s. Par ailleurs, un seul effet transitoire de polarisation est mesuré dans de nombreux fichiers de mesure. Par conséquent, il y a moins de valeurs d'intervalles de temps ΔT calculées que d'effets transitoires de polarisation mesurés. 20618 valeurs d'intervalle de temps entre deux effets transitoires de polarisation consécutifs ont été calculées pour toutes les périodes de mesure réalisées au polarimètre. La moyenne et l'écart type des intervalles de temps calculés sont égaux à 1.636 s et à 1.787 s respectivement. Les paramètres λ_{opt} , λ_{low} et λ_{upp} sont égaux à 0.561, 0.55 et à 0.572 respectivement. L'écart relatif entre le paramètre optimisé λ_{opt} et les paramètres optimisés λ_{low} et λ_{upp} s'élève à 1.96% pour un intervalle d'incertitude de 95%.

L'évaluation de l'ajustement de la fonction de distribution exponentielle $f_E(x, \lambda_{opt})$ a été effectuée par le calcul du coefficient de détermination R^2 et par le calcul du coefficient de corrélation de Pearson R_p décrits dans la section 4.3.1.3. Les valeurs de R^2 et de R_p s'élèvent à 0.979 et à 0.958. Ces deux valeurs étant proches de 1, l'ajustement de la fonction de distribution $f_{\Delta T}(x)$ par la fonction de densité exponentielle $f_E(x, \lambda)$ est très satisfaisant (Voir Figure 4.13). 10 233 valeurs d'intervalle de temps ΔT sont caractérisées par une durée

inférieure à 1 s. Lorsqu'un effet transitoire de polarisation est mesuré, la probabilité que se produit un effet transitoire de polarisation subséquent dans la même seconde de mesure est de 50.6 %. Par conséquent, les effets transitoires de polarisation mesurés dans un même fichier de mesures ont plus tendance à se concentrer sur de courtes périodes de mesures inférieures à 1 s qu'à se « disperser » sur toute la durée d'un fichier de mesures soit 10 s.

4.5 Conclusion du chapitre

Le traitement des vecteurs Stokes (S_1 , S_2 , S_3) enregistrées au polarimètre a représenté un véritable défi à relever. La structure de données indexée est un outil adapté à l'automatisation et à l'évolution des méthodes de calcul et d'analyse appliquées sur les vecteurs Stokes.

Le calcul des vitesses de rotation angulaire est un moyen efficace d'effectuer le *monitoring* des fluctuations des états de polarisation sur une liaison optique cohérente. Ces travaux ont permis la caractérisation approfondie de la nature et du comportement des vitesses de rotation angulaire des états de polarisation par le développement d'outils d'analyse dans le cadre du projet de *monitoring*. Les ajustements réalisés sur la fonction de densité $f_{ARR}(x)$ des vitesses de rotation angulaire ont permis l'identification de deux phénomènes statistiques à l'origine du mécanisme des fluctuations des états de polarisation. L'ajustement de l'histogramme normalisé $f_{ARR}(x)$ par la fonction de densité de Rayleigh a permis de modéliser seulement les vitesses de rotation angulaire inférieures à 0.5 rad/ms alors que les valeurs de l'histogramme des vitesses de rotation angulaire supérieures à 1.1 rad/ms ont été ajustées de manière satisfaisante avec une fonction de densité exponentielle. Une fonction de densité double, appelée f_{GEI} , combinant la fonction de densité de Rayleigh et la fonction de densité exponentielle a été développée par Crawford *et al.* (2013). La fonction f_{GEI} présente le modèle statistique le plus récent, au meilleur de nos connaissances, des vitesses de rotation angulaire des états de polarisation mesurés avec un polarimètre sur une liaison optique majoritairement aérienne.

Les effets transitoires de polarisation ont ensuite été caractérisés par l'application d'une méthode de calcul et d'une méthode d'analyse développées dans le cadre de ma maîtrise. Ces outils ont permis l'identification, la localisation et le calcul des paramètres caractéristiques des effets transitoires de polarisation. Les effets de transitoire de polarisation ont ainsi été caractérisés par le calcul de la valeur maximale de vitesse de rotation angulaire ARR_{max} , de la valeur d'élasticité ϕ_{max} , de la valeur de durée $\Delta\tau$ et de la valeur d'intervalle de temps ΔT entre deux effets transitoires de polarisation consécutifs dans un même fichier de mesures.

Enfin, les données collectées sur le lien aérien du réseau Verizon présentent une riche source d'études et d'analyses statistiques. La mise à jour des résultats des mesures en cours, au moment de l'écriture, validera les méthodes de calcul et les méthodes d'analyse décrites dans ce chapitre.

CONCLUSION

Les outils d'analyses développés au cours de cette maîtrise ont permis l'apport de nouvelles connaissances sur la nature et le comportement des effets de polarisation mesurés avec le transpondeur cohérent et avec le polarimètre sur la liaison optique cohérente du réseau CANARIE et du réseau Verizon.

Les structures de données indexées par « clé » qui résultent de l'application des outils de traitement de données permettent de stocker et de charger dynamiquement les paramètres de *monitoring* mesurés avec le transpondeur cohérent et avec le polarimètre en fonction du temps de début de chaque période d'acquisition, ce qui répond aux problèmes d'accès rapide des données et de gestion efficace de la mémoire des systèmes d'exploitation. De plus, la fonction de suppression des valeurs transitoires implémentée par les outils de traitement assure que les valeurs enregistrées dans les structures de données indexées « par clé » correspondent bien à des mesures valides.

D'un point de vue scientifique, les résultats issus de l'application des outils d'analyse dédiés aux mesures réalisées avec le transpondeur cohérent ont permis de caractériser la nature et le comportement des valeurs de DGD et de PDL sur deux liaisons optiques cohérentes. Les valeurs de dérivée temporelle de DGD et de PDL sont nettement plus élevées sur la liaison optique aérienne du réseau Verizon que sur la liaison optique enterrée du réseau CANARIE. De plus, les moyennes et les écarts types des valeurs de DGD et de PDL calculées pour chaque minute de mesure sont stables et quasi constants sur la liaison optique majoritairement enterrée du réseau CANARIE alors que les amplitudes de variations de ces mêmes valeurs sont 3 à 4 fois supérieures sur la liaison optique majoritairement aérienne du réseau Verizon, ce qui prouve l'existence d'activités en polarisation intenses sur ce type d'installation optique. De plus, il a été constaté que les moyennes et les écarts types de DGD et de PDL calculés pour chaque minute de mesure sont 2.5 à 4.1 fois supérieurs lors des journées de mesure (de 8 h à 20 h) que lors des nuits de mesure (de 20 h à 8 h le lendemain), ce qui démontrent l'existence de deux types d'activités en polarisation qui se produisent de

manière cyclique lors des mesures réalisées sur le lien aérien du réseau Verizon. Le calcul de la fonction d'autocorrélation normalisée après le filtrage super gaussien des valeurs de DGD et des valeurs de PDL a permis de confirmer que le temps de corrélation correspondant à chacun de ces effets de polarisation est de l'ordre de quelques minutes sur la liaison majoritairement aérienne du réseau Verizon alors que celle-ci s'étend sur plusieurs heures sur la liaison majoritairement enterrée du réseau CANARIE pour une période de 4 jours de mesures réalisées avec le transpondeur cohérent. Ces résultats confirment la présence de plus grandes amplitudes de variations des valeurs de DGD et de PDL mesurées sur la liaison du réseau Verizon et confortent les temps de corrélations calculées par Bao *et al.* (2004), par Waddy *et al.* (2005) et par Kawasaki *et al.* (2007). Enfin, le dernier outil d'analyse a permis de calculer et de tester l'ajustement de la fonction de densité de probabilité des valeurs de DGD par la fonction de densité de Maxwell comme démontrée dans les travaux de Kawasaki *et al.* (2007) et de Woodward *et al.* (2014). Il a été constaté après évaluation de l'ajustement que la fonction de densité de Maxwell n'ajuste pas avec qualité la fonction de densité de probabilité des valeurs de DGD supérieures à 5.5 ps, ce qui démontre qu'une fonction de densité mathématique « plus complexe » permettrait d'approfondir l'analyse statistique des valeurs de DGD mesurées avec le transpondeur cohérent sur la liaison aérienne longue distance du réseau Verizon.

L'application du premier outil d'analyse développée pour la caractérisation des fluctuations des états de polarisation a permis de constater l'existence de trois types de vitesses de rotation angulaire (lentes, modérées et rapides) des états de polarisation mesurés avec le polarimètre sur la liaison optique majoritairement aérienne du réseau Verizon. Le calcul et l'ajustement de la fonction de densité de probabilité des vitesses de rotation angulaire des états de polarisation implémentés dans ce premier outil d'analyse a permis d'identifier deux mécanismes à l'origine des fluctuations des états de polarisation mesurés sur le lien aérien du réseau Verizon; ce qui a permis d'approfondir les travaux de *monitoring* réalisés par Peterson *et al.* (2004) et Matsuda *et al.* (2009). La fonction de densité de probabilité des vitesses de rotation angulaire des états de polarisation a été ajustée par une fonction de densité double appelée f_{GEI} . Cette fonction de densité de probabilité est le résultat du calcul de la

convolution de la fonction de densité de Rayleigh et de la fonction de densité exponentielle développé et démontré par Crawford *et al.* (2013). De plus, les résultats du second outil d'analyse ont permis d'apporter de nouvelles connaissances sur la nature et le comportement des effets transitoires de polarisation et d'approfondir les travaux de caractérisation des effets transitoires de polarisation mesurés par Nelson (2012). En effet, il a été constaté que la majorité des effets transitoires de polarisation mesurés avec le polarimètre sur la liaison aérienne longue distance du réseau Verizon sont peu élastiques (inférieures à 0.26 rad) et de très courtes durées (inférieures à 0.05 ms). Les fonctions de densité de probabilité conjointes calculées par le second outil d'analyse ont permis de démontrer qu'il n'existe pas de corrélation entre les valeurs maximales de vitesse de rotation angulaire des états de polarisation et les valeurs d'élasticité des effets transitoires de polarisation. Par ailleurs, la probabilité que se produisent des effets transitoires de polarisation très élastiques (supérieures à 0.26 rad) sur de longues durées (supérieures à 0.5 ms) s'élève à 10^{-5} , ce qui démontre que des activités en polarisation intenses peuvent subvenir sur une installation aérienne.

L'équipe du *Laboratoire de technologies de réseaux* dispose désormais d'outils de traitement et d'analyse pour la caractérisation des effets de polarisation mesurés avec le transpondeur cohérent et avec le polarimètre. Dès que suffisamment de données auront été acquises sur la liaison optique majoritairement aérienne du réseau Verizon, les constatations et les analyses décrites tout au long de ce mémoire devront être validées ou approfondies par de nouvelles mesures réalisées sur d'autres liaisons optiques cohérentes sensibles aux mesures d'effets transitoires de polarisation.

Directions futures sur la caractérisation des effets de polarisation

Les multiples échanges (p. ex. mails, rencontres téléphoniques, etc.) avec Ciena ont permis de déterminer quelques travaux de caractérisation supplémentaires à réaliser dans le futur pour approfondir la nature et le comportement des effets de polarisation mesurés avec le transpondeur cohérent et avec le polarimètre.

Une validation complète des mesures réalisées avec le transpondeur cohérent reste à faire en comparant les résultats avec ceux obtenus avec le polarimètre. En raison de la dégradation des composantes S_2 et S_3 par le diviseur de polarisation disposé à l'entrée de l'architecture photonique du récepteur cohérent, les moyennes et les écarts types des valeurs angulaires (composante φ) du paramètre S_1 seront calculées à chaque seconde et à chaque minute de mesure comme démontré dans les travaux de *monitoring* de Woodward *et al.* (2012) et de Woodward *et al.* (2014). Un outil d'analyse sera développé pour la caractérisation des valeurs angulaires φ et de leurs dérivées temporelles $d\varphi/dt$ et pour le calcul de la fonction de densité de probabilité et de la fonction d'autocorrélation normalisée des valeurs de dérivée temporelle de l'angle φ . Cet outil d'analyse sera appliqué sur les mesures du paramètre S_1 qui compose les vecteurs Stokes (S_1, S_2, S_3) pour chacun des instruments, ce qui permettra de valider efficacement les données enregistrées avec le transpondeur cohérent. De plus, un travail de caractérisation supplémentaire consiste à corrélérer les variations des valeurs angulaires φ , des valeurs de DGD et des valeurs du taux d'erreur binaire avant application du code de correction d'erreur (pre-FEC BER) enregistrées par le transpondeur cohérent. L'impact des fluctuations des valeurs de S_1 sur les valeurs de DGD et sur les valeurs de pre-FEC BER sera analysé par l'identification d'un seuil de variation des valeurs de DGD correspondant à un seuil de variation des valeurs angulaires φ et possiblement par l'identification d'un seuil des variations des valeurs de pre-FEC BER désigné aussi par un seuil de variation des valeurs angulaires φ .

Une nouvelle fonction de densité de probabilité des valeurs de DGD sera calculée après la modification de l'algorithme de calcul des paramètres de *monitoring* à l'aide des coefficients de filtres numériques sauvegardés par le module DSP du transpondeur cohérent. Ce travail de caractérisation futur permettra d'expliquer les écarts entre l'histogramme des valeurs de DGD mesurées par le transpondeur cohérent sur la liaison aérienne du réseau Verizon et la courbe d'ajustement de la fonction de densité de Maxwell.

Enfin, un dernier travail de caractérisation qui ressort des échanges avec Ciena a pour objectif la réalisation de nouvelles analyses sur la fonction de densité des vitesses de rotation angulaire des états de polarisation mesurés avec le polarimètre sur le lien aérien du réseau Verizon. De cette manière, il sera possible d'expliquer les écarts qui subsistent entre les valeurs de l'histogramme des vitesses de rotation angulaire des états de polarisation et de la courbe d'ajustement de la fonction de densité double f_{GEI} particulièrement sur les vitesses de rotation angulaire modérées comprises entre 0.5 et 1.0 rad/ms.

Directions futures sur les outils de traitement et d'analyse

Un important volume de données (supérieur à 1 Tb) va être généré par les mesures des effets de polarisation avec le transpondeur cohérent et avec le polarimètre durant plusieurs mois à partir du mois d'octobre 2013. L'acquisition d'un serveur d'archivage et l'installation d'une base de données SQL permettrait de subvenir aux problèmes de stockage qui va devenir de plus en plus préoccupant dans les prochains mois du projet de *monitoring*. La plupart des outils d'analyse développés durant ma maîtrise n'utilisent qu'un seul processeur. Avec le nombre croissant des données enregistrées dans le cadre de ce projet de recherche, la parallélisation des calculs dépendamment du nombre de processeur de l'ordinateur utilisé deviendra indispensable pour réduire considérablement le temps de traitement des données vers les structures de données indexées par « clé » et pour rendre les outils d'analyse plus « extensibles » lors des prochains travaux de caractérisation des effets de polarisation mesurés avec le transpondeur cohérent et avec le polarimètre.

ANNEXE I

INTERFACE D'ACQUISITION TELNET DES PARAMÈTRES DE *MONITORING* ET DE DIAGNOSTIQUE AVEC UN TRANSPONDEUR COHÉRENT DP-QPSK

La carte de transmission est connectée à un serveur suivant une connexion de réseau local (LAN). A l'aide des interfaces des outils d'ouverture de session, cinq sessions TELNET peuvent être ouvert avec cette carte. Une session TELNET se met à jour toutes les centaines de millisecondes (à l'ordre du 10Hz).

```
ottuser@pcardqw116:~
  Rx Overview
 2 Chan (Active) 0
 3 PreVoaTapPwr [dBm] -7.21559 19 PostVoaTapPwr [dBm] -14.9912
 4 Last AGC DAC XI 2082 20 Output Power XI 121
 5 Last AGC DAC XQ 1803 21 Output Power XQ 121
 6 Last AGC DAC YI 2194 22 Output Power YI 121
 7 Last AGC DAC YQ 2118 23 Output Power YQ 121
 8 Seq State TRACK STEADY_ 24 Ber 2.90686e-08
 9 Clock Kd 321 25 BER Source PFEC
 11 Dispersion ps/nm 1750 26 PRBS lock Count 832
 12 SNR X (dB) 13.4384 27 Curr Freq Offs -203
 13 PmdEst ps 3.54692 28 SNR Y (dB) 13.2218
 14 CarrSlipRate 0 29 PdlEst dB 0.267687
 15 BER Est:X pol 1 30 Qrew Temperature(C) 60
 16 BER Est 1 31 BER Est:Y pol 1
 32 RxProvModFormat QPSK

-----+-----
' '=refresh, 'c'=change, 'q'=quit
```

Figure-A I-1 Interface d'acquisition TELNET des paramètres de *monitoring* et de diagnostic par le transpondeur cohérent DP-QPSK à 40Gbit/s

L'interface TELNET d'acquisition ci-dessus décrit l'ensemble les principales valeurs mesurées au sein de la ligne optique WDM. Afin d'enregistrer ces valeurs dans des fichiers attitrés, il faut exécuter cette fonction d'écoute et de sauvegarde dans la fenêtre des commandes du serveur linux REDHAT :

```
tcpdump -i eth0 -s0 src xx.xx.xx.xxx -w telnet.bin
```

La première ligne active la fonction TCPDUMP et permettra d'enregistrer l'ensemble des données TCP inhérent à la connexion LAN. Cette fonction permet de créer des fichiers binaires. Ces fichiers n'étant pas exploitables directement. Il faudra appliquer de nouveaux outils à celle-ci afin de garantir un bon recouvrement des données.

La deuxième ligne active la fonction permettant de transformer les fichiers binaires vers des fichiers textes.

```
/usr/sbin/tcpdump -r telnet.bin -tttt -s0 -A > telnet.txt
```

C'est une étape indispensable à l'application d'outils avancés assurant le traitement des données.

ANNEXE II

ANALYSEUR DE TEXTE PHP POUR L'EXTRACTION DES DONNÉES DANS LES FICHIERS TEXTES

Script-A II-1 Analyseur de texte PHP pour la conversion des fichiers textes binaires vers les fichiers CSV

```
1.  #!/usr/bin/php -q
2.  <?php
3.  # don't let this script run for more than 60 seconds
4.  #set_time_limit(60);
5.  # turn off output buffering
6.  ob_implicit_flush(false);
7.
8.  # turn off error reporting, as it will most likely interfere with
9.  # the AGI interface
10. error_reporting(0);
11. $myFile = $argv[1];
12. fwrite(STDOUT, "$myFile \n");
13.
14. $fh = fopen($myFile, 'r');
15.
16. fwrite(STDOUT, "timestamp, PreVoaTapPwr, Last_AGC_DAC_XI,
17. Last_AGC_DAC_XQ, Last_AGC_DAC_YI, Last_AGC_DAC_YQ, Clock_Kd, SNR_X,
18. PmdEst_ps, PostVoaTapPwr, Ber, Curr_Freq_Offs, SNR_Y, PdlEst_dB
19. \n");
20.
21. $timestamp_old = null;
22.
23. while (!feof($fh)) {
24.     $temp = fgets($fh);
25.     if (preg_match("/([0-9]{4})-([0-9]{2})-([0-9]{2}) ([0-9]{2}):([0-9]{2}):([0-9]{2}).([0-9]{5})/i", $temp, $matches)) {
26.         $timestamp_new=$matches[0];
27.     }
28.
29.
30.
31.     if (preg_match("/PreVoaTapPwr
32. \[dBm\]\.\.[3\;27H(.*)\.\.[4\;1H/i", $temp, $matches)) {
33.         $PreVoaTapPwr=$matches[1];
34.     }
35.     if (preg_match("/Last AGC DAC
36. XI\.\.[4\;27H(.*)\.\.[5\;1H/i", $temp, $matches)) {
37.         $Last_AGC_DAC_XI=$matches[1];
38.     }
39.     if (preg_match("/Last AGC DAC
40. XQ\.\.[5\;27H(.*)\.\.[6\;1H/i", $temp, $matches)) {
41.         $Last_AGC_DAC_XQ=$matches[1];
42.     }
43.     if (preg_match("/Last AGC DAC
```

```

44. YI\\.\\[6\\;27H(.*)\\.\\[7\\;1H/i", $temp, $matches) {
45.     $Last_AGC_DAC_YI=$matches[1];
46. }
47. if (preg_match("/Last AGC DAC
48. YQ\\.\\[7\\;27H(.*)\\.\\[8\\;1H/i", $temp, $matches) {
49.     $Last_AGC_DAC_YQ=$matches[1];
50. }
51. if (preg_match("/Clock
52. Kd\\.\\[9\\;27H(.*)\\.\\[10\\;1H/i", $temp, $matches) {
53.     $Clock_Kd=$matches[1];
54. }
55. if (preg_match("/SNR X
56. \\(dB\\)\\.\\[12\\;27H(.*)\\.\\[13\\;1H/i", $temp, $matches) {
57.     $SNR_X=$matches[1];
58. }
59. if
60. (preg_match("/PmdEst_ps\\.\\[13\\;27H(.*)\\.\\[14\\;1H/i", $temp, $matches)
61. ) {
62.     $PmdEst_ps=$matches[1];
63. }
64. if (preg_match("/PostVoaTapPwr
65. \\[dBm\\)\\.\\[3\\;67H(.*)\\.\\[4\\;41H/i", $temp, $matches) {
66.     $PostVoaTapPwr=$matches[1];
67. }
68. if (preg_match("/Curr Freq
69. Offs\\.\\[11\\;67H(.*)\\.\\[12\\;41H/i", $temp, $matches) {
70.     $Curr_Freq_Offs=$matches[1];
71. }
72. if (preg_match("/Ber\\.\\[8\\;67H(.*)\\.\\[9\\;41H/i", $temp, $matches)
73. {
74.     $Ber=$matches[1];
75. }
76. if (preg_match("/SNR Y
77. \\(dB\\)\\.\\[12\\;67H(.*)\\.\\[13\\;41H/i", $temp, $matches) {
78.     $SNR_Y=$matches[1];
79. }
80.
81. if
82. (preg_match("/PdlEst_dB\\.\\[13\\;67H(.*)\\.\\[14\\;41H/i", $temp, $matches
83. )) {
84.     $PdlEst_dB=$matches[1];
85. }
86. if ($timestamp_old != $timestamp_new){
87.     fwrite(STDOUT, "$timestamp_new, $PreVoaTapPwr,
88. $Last_AGC_DAC_XI, $Last_AGC_DAC_XQ, $Last_AGC_DAC_YI,
89. $Last_AGC_DAC_YQ, $Clock_Kd, $SNR_X, $PmdEst_ps, $PostVoaTapPwr,
90. $Ber, $Curr_Freq_Offs, $SNR_Y, $PdlEst_dB \n");
91.     fflush(STDOUT);
92. }
93.     $timestamp_old=$timestamp_new;
94. }
95.
96. fclose ($fd);
97.
98. ?>

```

ANNEXE III

STRUCTURE DES FICHIERS DE MESURE GÉNÉRÉS PAR LE PROGRAMME D'ACQUISITION AVEC LE TRANSPONDEUR COHÉRENT

Le programme développé par Ciena a été appliqué sur les données enregistrées dans la mémoire interne de la carte OCLD afin que celles-ci puissent être converties dans des fichiers CSV. La figure-A III-1 décrit les valeurs qui constituent le titre de chaque fichier de mesure CSV.

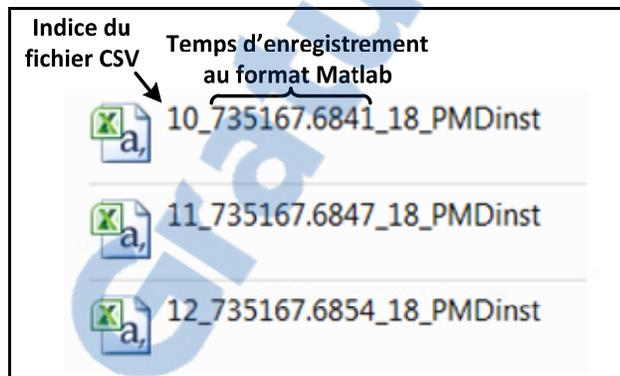


Figure-A III-1 Titre du fichier de mesure CSV des paramètres mesurés par la carte OCLD sur la liaison du réseau Verizon

La deuxième valeur du titre du fichier désigne le temps d'enregistrement du fichier de mesure au format Matlab. La figure-A III-2 présente un exemple de fichier de mesure CSV obtenu après l'acquisition des données par la carte OCLD sur le lien du réseau CANARIE.

timestamp	unix_time	chrono_time	PreVoaTapPwr	Last_AGC_DAC_XI	Last_AGC_DAC_XQ	Last_AGC_DAC_YI	Last_AGC_DAC_YQ
03/11/2011 13:17:27.76087	1320340647.7609	1.143	-7.05724	2141	1848	2163	2088
03/11/2011 13:17:28.82009	1320340648.8201	2.202	-7.05724	2153	1856	2165	2094
03/11/2011 13:17:29.85665	1320340649.8567	3.239	-7.05724	2145	1855	2161	2086
03/11/2011 13:17:30.89825	1320340650.8983	4.280	-7.05724	2145	1851	2153	2081
03/11/2011 13:17:31.92568	1320340651.9257	5.308	-7.06996	2152	1860	2175	2081
Clock_Kd	SNR_X	PmdEst_ps	PostVoaTapPwr	Ber	Curr_Freq_Offs	SNR_Y	PdIEst_dB
318	13.5723	4.14454	-15	3.41976e-08	-187	13.2047	0.362239
320	13.4468	5.11295	-15	3.0033e-08	-187	13.299	0.52346
339	13.1768	2.3433	-15.0003	2.51659e-08	-185	13.2076	0.698124
332	13.1547	4.08348	-15.0003	2.79329e-08	-185	13.0783	0.730372
319	13.3199	2.48668	-14.9857	2.25416e-08	-184	13.1445	0.333119

Figure-A III-2 Exemple de fichier de mesure CSV avec la carte OCLD sur la liaison du réseau CANARIE

La figure-A III-3 présente un exemple de fichier CSV obtenue lors des mesures réalisées par la carte OCLD sur la liaison du réseau Verizon.

index	time_ms	PMD	Dispersion	PDL_dB	SNR_X	SNR_Y	PreFEC BER	inst_PMD	inst_PDL	s0	s1	s2	s3
37	26017	2.2218	15350	1.3629	11.043	11.011	2.7381e-06	3.1188	1.5571	1	-0.43196	-0.032044	0.90132
38	26018	2.2218	15350	1.3629	11.043	11.011	2.7381e-06	0.81131	1.1446	1	-0.42072	0.65952	-0.62292
39	26019	2.2218	15350	1.3629	11.043	11.011	3.4737e-06	2.1791	1.1501	1	-0.37284	-0.35754	0.85624
hxx0	hxx1	hxx2	hxx3	hxx4	hyx0	hyx1	hyx2	hyx3	hyx4	hxy0	hxy1	hxy2	hxy3
1.0542e+09	4.7055e+07	9.9354e+07	5.1708e+07	1.0402e+09	1.045e+09	7.689e+07	1.8521e+08	8.9932e+07	1.0169e+09	1.0643204e+07	2.3933e+07	4.7399e+07	4.7399e+07
1.1026e+07	1.0351e+09	9.4092e+08	1.0378e+09	2.7345e+07	6.5696e+06	1.0254e+09	8.8795e+08	1.0174e+09	3.6978e+07	1.0610713e+09	8.4838e+08	1.0724e+09	1.0724e+09
1.0158e+07	1.0423e+09	9.6838e+08	1.0424e+09	2.2414e+07	1.6859e+07	1.0092e+09	8.7525e+08	1.0063e+09	4.8366e+07	1.05679303e+06	1.0305e+09	8.8479e+06	8.8479e+06
hxy4	hyy0	hyy1	hyy2	hyy3	hyy4	Ch Pwr	RTD	Freq Err					
1.0481e+09	4.866e+06	1.0444e+09	1.0713e+09	1.0505e+09	9.5842e+06	31016	-1	6000					
1.4484e+07	7.3716e+05	9.8309e+06	1.5394e+08	2.6149e+07	1.051e+09	31657	-1	6000					
1.0669e+09	2.7688e+06	2.3099e+06	1.0471e+09	1.3922e+06	1.0728e+09	31221	-1	6000					

Figure-A III-3 Exemple de fichier de mesure CSV avec la carte OCLD sur la liaison du réseau Verizon

Les paramètres de *monitoring* sélectionnés pour la caractérisation de la liaison du réseau Verizon sont les valeurs du chronomètre (colonne *time_ms*), les valeurs de PMD instantanées (colonne *inst_PMD*) et les valeurs de PDL instantanées (colonne *inst_PDL*). On observe sur la figure-A III-3 que le chronomètre s'incrémente à chaque ms ce qui confirme la fréquence d'échantillonnage des valeurs de PMD et de PDL égales à 1 kHz calculées à partir des coefficients de filtre numérique (*filter taps*).

L'évolution temporelle des effets de PMD et de PDL a pu être effectuée à partir du temps d'enregistrement du fichier de mesure au format Matlab et de la valeur du chronomètre appelée *time_ms* présentée sur la figure-A III-3.

ANNEXE IV

STRUCTURE DES FICHIERS DE MESURE GÉNÉRÉS PAR LE PROGRAMME D'ACQUISITION AVEC LE POLARIMÈTRE

Les fichiers de mesures réalisées au polarimètre sont au format Matlab. Chaque titre suit la nomenclature suivante « *Identifiant_Année_Mois_Jour_Heure_Minute_Second* ». Ces valeurs de temps désignent le temps d'enregistrement du fichier de mesure par le polarimètre. La figure-A IV-1 décrit la structure des fichiers Matlab enregistrés par le polarimètre du 25 au 30 octobre 2012 et du 4 au 19 septembre 2013.

DOP	<200000x1 single>
p	<200000x1 single>
sop	<200000x3 single>

Figure-A IV-1 Structure des fichiers Matlab enregistrés par le polarimètre sur la liaison du réseau Verizon du 25 au 30 octobre 2012 et du 4 au 19 septembre 2013

Chaque fichier Matlab enregistré du 25 au 30 octobre est constitué de trois matrices. La matrice « DOP » est composée des valeurs de degré de polarisation. La matrice « p » est composée des valeurs de puissance à la réception et la matrice « sop » définit une matrice de trois colonnes où chaque colonne indique une composante du vecteur Stokes (S_1 , S_2 , S_3) d'après la figure-A IV-1. Toutes ces matrices sont constituées de 200 000 valeurs mesurées durant 10 s avec une fréquence d'échantillonnage égale à 20 kHz. Toutefois, la structure du fichier Matlab enregistré par le polarimètre du 5 au 11 novembre 2012 est plus détaillée que celle de la figure-A IV-1. La figure-A IV-2 décrit la structure du fichier Matlab enregistré par le polarimètre du 5 au 11 novembre 2012.

DOP	<200000x1 single>
p	<200000x1 single>
sampling_rate	20
sop	<200000x3 single>
started_at	[2012 11 5 14 54 41.3120]

Figure-A IV-2 Structure des fichiers Matlab enregistrés par le polarimètre sur la liaison du réseau Verizon du 5 au 11 novembre 2012

Deux valeurs supplémentaires constituent la structure du fichier Matlab selon la figure-A IV-2. La valeur *sampling_rate* définit la fréquence d'échantillonnage de la fonction « polarimètre » du synthétiseur de polarisation et la valeur *started_at* désigne le temps du début de la mesure du synthétiseur de polarisation avec une précision à la microseconde (Voir Figure-A IV-2). Le temps de début de la prise de mesure est plus précis que le temps d'enregistrement du titre de chaque fichier Matlab enregistrée par le polarimètre sur toutes les périodes de mesure.

ANNEXE V

OUTIL DE TRAITEMENT DES DONNÉES ENREGISTRÉES AVEC LE TRANSPONDEUR COHÉRENT DP-QPSK À 40 GBIT/S

Le script-A V-1 définit l'outil de traitement des paramètres de performances et des effets de polarisation mesurés avec le transpondeur cohérent DP-QPSK à 40 Gbit/s développé dans l'environnement Matlab.

Script-A V-1 Développement d'une structure de données indexée par « clé »
des effets de polarisation et de performance mesurés avec le transpondeur cohérent

```
1.  %% Creer des .mat a partir des csv avec la DGD et la PDL INST
2.  close all
3.  clear all
4.  clc
5.
6.  % dossierCSV='C:\Documents and Settings\AK05390\Bureau\test\';
7.  %Dossier des .csv
8.  dossierCSV='D:\Projet - Liaison Sacramento - Portland\Raw Files\10-
9.  25 to 10-30 Recordings_OCLD\';
10. dossierSortie='D:\Projet - Liaison Sacramento - Portland\test\';
11.
12. dirContents = dir([dossierCSV '*.csv']);
13.
14. nFiles = length(dirContents);
15. fprintf('%s\n',dossierCSV)
16.
17.
18. for ind_files=1:nFiles          % recommandé -> for plop=<id premier
19. fichier>:<id premier fichier>+nFiles
20.     tab_mesure = [];
21.
22.     SORTIE=zeros(8738,9);
23.
24.     try
25.         searchResults = dir(sprintf('%s%i_*.csv', dossierCSV,
26. ind_files));
27.         nomfichier = searchResults.name;
28.
29.         chemincomplet=strcat(dossierCSV,nomfichier);
30.         if(exist(chemincomplet,'file'))
31.             %% DONNEES FICHER CSV
32.
33.             % fid = fopen('data.csv', 'rt');
34.             fid = fopen(chemincomplet, 'rt');
35.             car = fread(fid);
36.             fclose(fid);
37.             data = strrep(char(car), ',', ':');%remplace "," par
38.             "."
```



```

94.
95.         if (tab_mesure_DGDinst(i-1,1)==-1 &&
96. tab_mesure_DGDinst(i,1)<12) || (tab_mesure_PDLinst(i-1,1)==-1 &&
97. tab_mesure_PDLinst(i,1)<4.5)
98.             flag=1;
99.             end_init=i;
100.         end
101.     end
102.     % ID du fichier OCLD + fin du glitch
103.     init_DGD(end+1,1:2)=[fileIx,end_init];
104.     % Création de la Structure de tous les paramètres de
105. fichier OCLD valides
106.
107.     if (flag ==1); %% permet de valider le temps de
108. réinitialisation...
109.         SORTIE = removerows(SORTIE,1:end_init);
110.     end
111.
112.     if ((max(SORTIE(end_init:end,2)) > 12) |
113. (min(SORTIE(end_init:end,2)) <= 0))
114.         suspect_files(end+1,1:3) =
115. [fileIx,min(SORTIE(end_init:end,2)),max(SORTIE(end_init:end,2))];
116.     end
117.
118.     if (isempty(fieldnames(Struct_OCLD_data)))
119.         id = 1;
120.     else id = length(Struct_OCLD_data)+1;
121.     end
122.
123.
124.     if(isempty(SORTIE)==0)
125.         Struct_OCLD_data(id).started_at = SORTIE(1,1);
126.         Struct_OCLD_data(id).TS= SORTIE(:,1);
127.         Struct_OCLD_data(id).DGD= SORTIE(:,2);
128.         Struct_OCLD_data(id).PDL= SORTIE(:,3);
129.         Struct_OCLD_data(id).BER= SORTIE(:,4);
130.         Struct_OCLD_data(id).SNRx= SORTIE(:,5);
131.         Struct_OCLD_data(id).SNRy= SORTIE(:,6);
132.         Struct_OCLD_data(id).S1= SORTIE(:,7);
133.         Struct_OCLD_data(id).S2= SORTIE(:,8);
134.         Struct_OCLD_data(id).S3= SORTIE(:,9);
135.     end
136. end
137. catch
138.     fprintf('Pas de %d \n',ind_files);
139.     continue;
140. end
141. end

```


ANNEXE VI

ÉTAPE 1 : CRÉATION D'UN RÉPERTOIRE DE FICHIERS CSV ENREGISTRÉS AVEC LE TRANSPONDEUR COHÉRENT

La figure-A VI-1 décrit le processus de création du répertoire de fichiers appelé *OCLD_Data_Struct* désignant l'adresse de chaque fichier de mesure CSV enregistré par la carte OCLD.

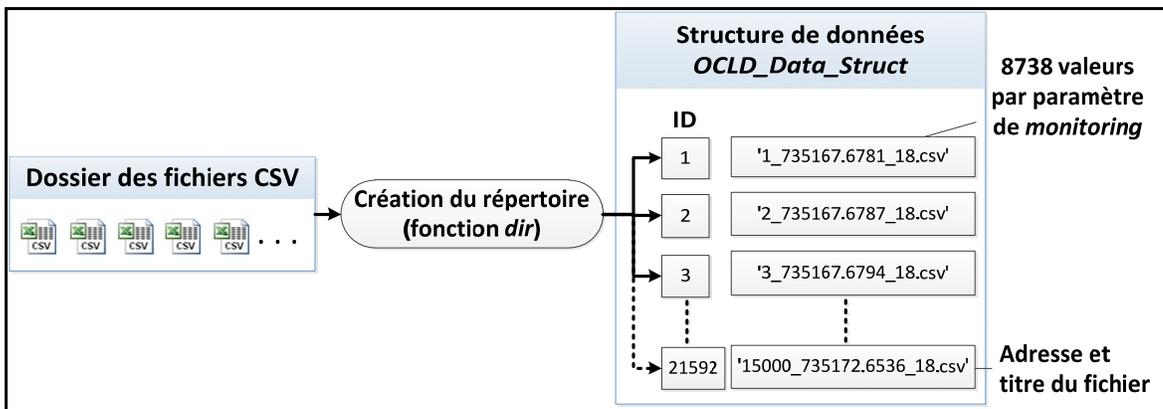


Figure-A VI-1 Création de la structure *OCLD_Data_Struct* des fichiers de mesure CSV

La figure-A VI-1 est une formalisation d'une partie du script du traitement des données élaborant la structure *OCLD_Data_Struct* (Voir ANNEXE V, p. 131). Un identifiant numérique *ID* est associé à chaque fichier CSV désignant son chemin d'accès. La fonction *dir* paramétrée par l'extension des fichiers de données '.csv' a été appliquée sur le dossier des fichiers de mesure CSV pour la création de la structure de données *OCLD_Data_Struct*. La vitesse d'accès et de lecture des fichiers CSV est égale à 211 fichiers CSV/h en mode lecture directe et près de 3100 fichiers CSV/h lorsque la lecture des fichiers CSV est basée sur le répertoire d'adresses. Le passage par une structure de données indexée par « clé » accroît significativement la vitesse d'accès aux données enregistrées avec le transpondeur cohérent. Le résultat de l'application de la fonction sur le dossier des fichiers de mesure OCLD se présente sous la forme d'un répertoire de 21592 adresses. La figure-A VI-2 décrit un exemple d'un index composant le répertoire Matlab des fichiers CSV.

abc	name	'1000_735168.3369_18.csv'		
abc	date	'26-oct.-2012 09:05:18'		
	bytes	2870739	2870739	2870739
	isdir	0		
	datenum	7.3517e+05	7.3517e+05	7.3517e+05

Figure-A VI-2 Exemple d'un index du répertoire Matlab contenant les adresses des fichiers CSV enregistrés avec le transpondeur cohérent

ANNEXE VII

ÉTAPE 2.1 : LECTURE BINAIRE DES FICHIERS CSV ET CONVERSION EN FORMAT TEXTE NUMÉRIQUE

La figure-A VII-1 décrit la méthode de conversion des fichiers CSV en fichier texte numérique *OCLD_text* lors de la première étape de traitement.

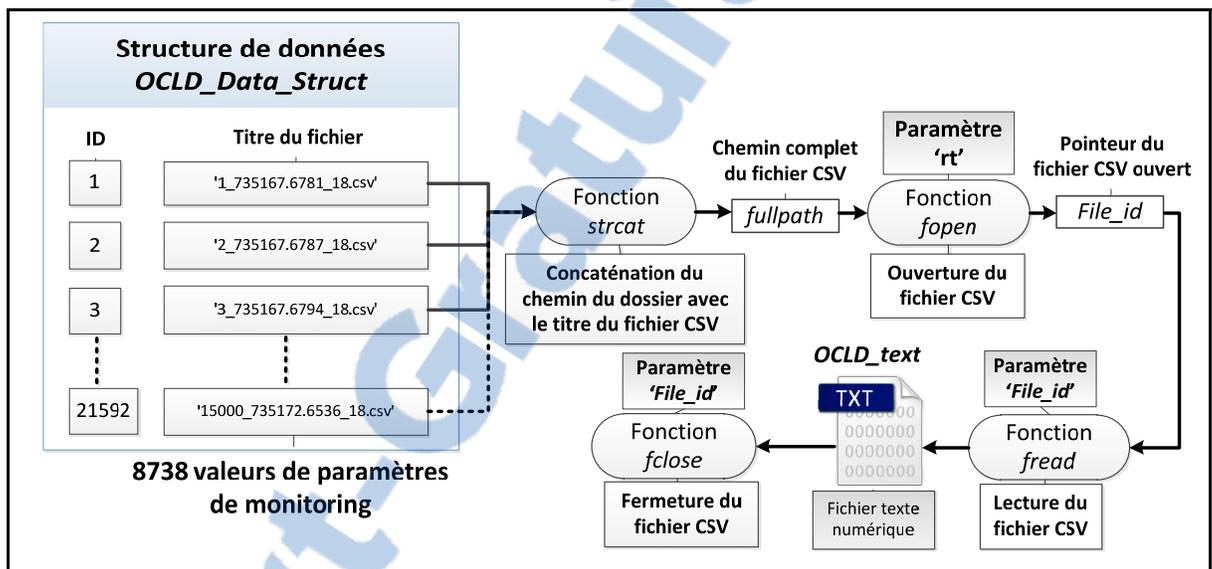


Figure-A VII-1 Conversion des fichiers CSV en fichier texte numérique *OCLD_text*

Le chemin complet *fullpath* a été créé pour chaque fichier CSV par l'application de la fonction *strcat* sur la structure d'adresses *StructData* et sur le dossier des fichiers CSV (Voir Figure-A VII-1).

En effet, cette fonction Matlab effectue la concaténation du chemin du dossier des fichiers CSV et du titre de chaque fichier. Le fichier CSV a ensuite été chargé avec la fonction *fopen*. La fonction *fopen* a été configurée avec l'option 'rt' (signifiant *reading in text mode*) et génère un pointeur sous le nom de *File_id* pour une lecture efficace des données (Voir Figure-A VII-1). Le fichier texte numérique, nommé *OCLD_text*, a été créé avec la fonction *fread*. Cette fonction capture les données du fichier CSV pour les enregistrer dans un fichier texte numérique dont chaque valeur numérique correspond à une lettre ou à un nombre. Enfin, la fonction *fclose* ferme le fichier CSV lorsque la lecture de celui-ci est terminée.

ANNEXE VIII

ÉTAPE 2.2 : CONVERSION DES FICHIERS TEXTES NUMÉRIQUES EN MATRICES MATLAB

La figure-A VIII-1 décrit la seconde étape de traitement pour la conversion des fichiers textes numériques *OCLD_text* en matrices Matlab *OCLD_params* constituées des paramètres de *monitoring*.

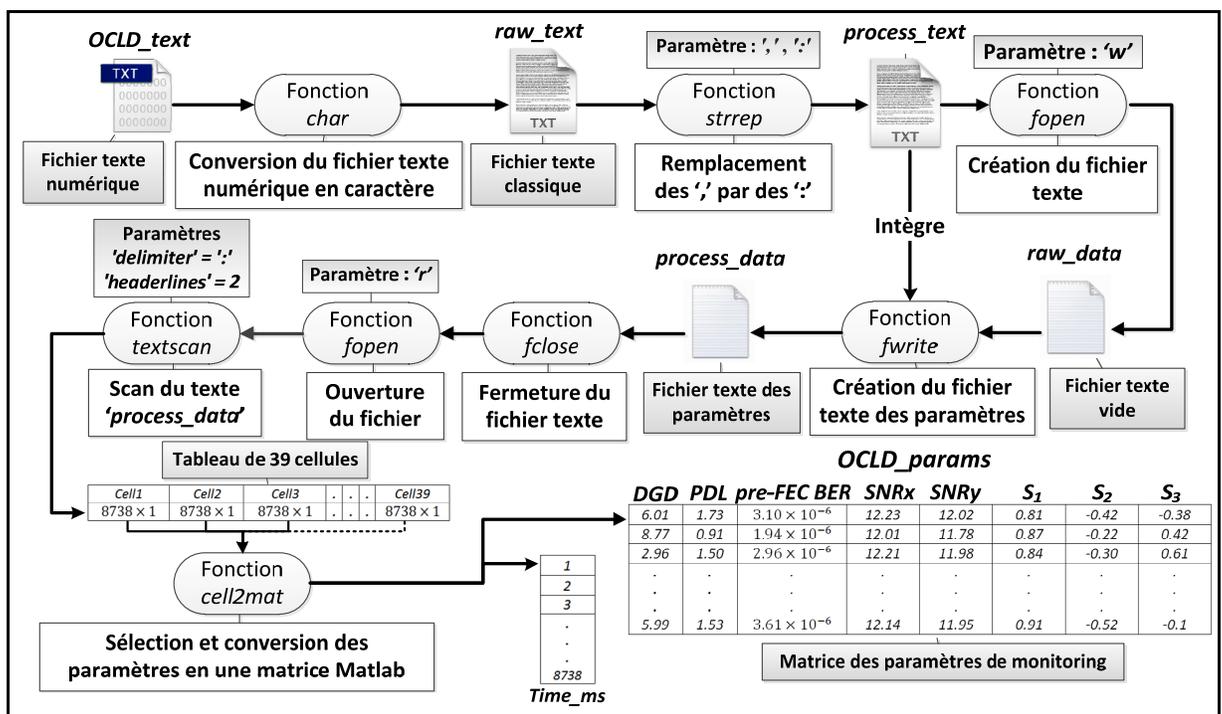


Figure-A VIII-1 Conversion du fichier texte numérique en matrice Matlab

La seconde étape de traitement, décrite par la figure-A VIII-1, est une représentation du code Matlab développé pour la réalisation de cette procédure (Voir ANNEXE V, p. 129).

Tout d'abord, le texte numérique *OCLD_text* précédemment obtenu a été converti en fichier texte classique nommé *raw_text* par l'application de la fonction *char* (Voir Figure-A VIII-1). Afin qu'une détection précise des paramètres puisse être effectuée, il est nécessaire de remplacer toutes les virgules par deux points (soit le symbole ':') par l'application la fonction *strrep*. Ensuite, un fichier texte vide appelé *raw_data* a été créé et configuré en mode écriture 'w' au moyen de la fonction *fopen*. Les données traitées au format texte dans lequel les ',' sont remplacés par ':', sont copiées dans le fichier texte *process_data*. Ce même fichier a été fermé puis a été chargé à nouveau en mode lecture ('r'). La configuration de la précédente fonction requiert deux options : le paramètre 'delimiter' ciblant les cellules

du texte *process_data* et le paramètre '*headerlines*' supprimant les lignes d'entêtes du texte correspondant aux étiquettes (c'est-à-dire aux textes descriptifs) des paramètres de *monitoring*. La fonction *textscan* a ensuite été appliquée sur le fichier texte *process_data* pour que chaque paramètre puisse être capturé et enregistré dans un tableau de cellules. La figure-A VIII-2 présente les paramètres de *monitoring* capturés par la fonction *textscan* dans le tableau de cellule.

1	index	10	inst_PDL	19	hxx4	28	hxy3
2	time_ms	11	s0	20	hyx0	29	hxy4
3	PMD	12	s1	21	hyx1	30	hyy0
4	Dispersion	13	s2	22	hyx2	31	hyy1
5	PDL_dB	14	s3	23	hyx3	32	hyy2
6	SNR_X	15	hxx0	24	hyx4	33	hyy3
7	SNR_Y	16	hxx1	25	hxy0	34	hyy4
8	PreFEC BER	17	hxx2	26	hxy1	35	Ch Pwr
9	inst_PMD	18	hxx3	27	hxy2	36	RTD
						37	Freq Err

Figure-A VIII-2 Paramètres capturés dans le tableau de cellules

37 paramètres de *monitoring* ont été capturés (Voir Figure-A VIII-2). La fonction *cell2mat* a été appliquée sur les cellules correspondant aux paramètres de *monitoring* pour les convertir dans une matrice Matlab appelée *OCLD_params* (Voir Figure-A VIII-1). 21592 matrices Matlab constituées de 8738 lignes par paramètre de *monitoring* ont été créées par l'application de cette seconde étape de traitement pour la conversion des paramètres en matrice Matlab.

ANNEXE IX

ÉTAPE 3 : CALCUL ET INSERTION DES VALEURS D'HORODATAGE

La figure-A IX-1 décrit la procédure de traitement appliquée sur la structure d'adresse *StructData* pour l'automatisation de l'insertion des valeurs d'horodatage.

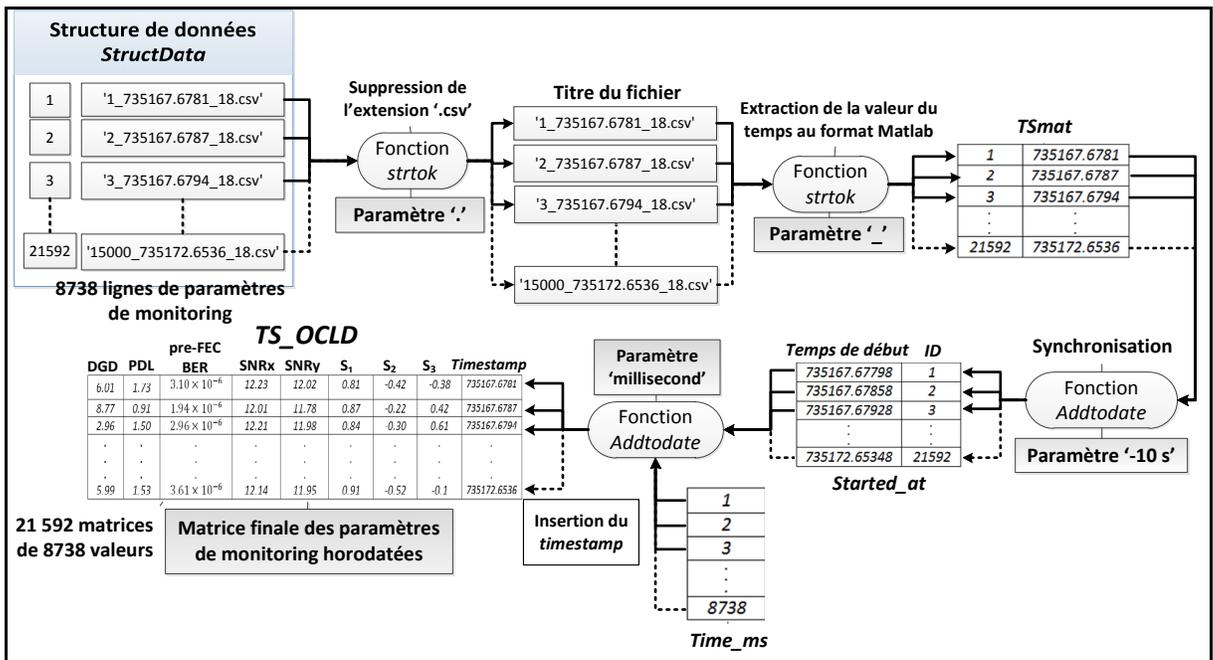


Figure-A IX-1 Procédure de création et d'insertion des valeurs d'horodatage dans les fichiers de mesures enregistrés par la carte OCLD

Le résultat de cette méthode de traitement se présente sous la forme de 21592 matrices de paramètres de *monitoring* horodatés appelés *TS_OCLD* (Voir Figure-A IX-1). La méthode de traitement présentée par la figure-A IX-1 est une représentation du code Matlab développé (Voir ANNEXE V, p. 129) pour l'automatisation de l'insertion des valeurs d'horodatage.

L'extension '.csv' du titre de chaque fichier a tout d'abord été supprimée par l'application de la fonction *strtok* paramétré par l'option '.' pour l'extraction de la deuxième valeur du titre de fichier qui indique le temps d'enregistrement de la mesure. En effet, la date inscrite sur le titre de chaque fichier Matlab correspond au moment où un fichier de mesure est sauvegardé après 8.738 s de capture de données par la carte OCLD. Le temps d'enregistrement au format Matlab a été capturé par la seconde application de la fonction *strtok* paramétrée par l'option '_'. Le temps d'enregistrement *TS* au format Matlab a été enregistré dans la matrice *TSmat* (Voir Figure-A IX-1).

La fonction *addtodate* a par la suite été appliquée sur les temps d'enregistrement *TS* de la matrice *TSmat* pour la synchronisation des valeurs d'horodatage avec l'échelle des temps de mesure. Le fichier étant sauvegardé à la fin de la mesure, il est nécessaire de soustraire la durée de la mesure au temps d'enregistrement de chaque fichier dans le but d'obtenir les valeurs réelles auxquelles les paramètres ont été mesurés. La durée temporelle de 8.738 a été soustraite sur toutes les valeurs d'horodatage de la matrice *TSmat* par l'application de la fonction *addtodate* paramétrée par l'option *second* (Voir ANNEXE V, p. 129). Les valeurs d'horodatage synchronisées ont été enregistrées dans la matrice *Started_at* (Voir Figure-A IX-1).

Enfin, les valeurs d'horodatage associées à chaque paramètre mesuré ont été insérées dans la matrice *TS_OCLD* par l'incrémentation de la valeur du chronomètre *Time_ms*, en ms (Voir Figure-A IX-1). Les valeurs du chronomètre ont été additionnées aux temps de début de chaque fichier par la seconde application de la fonction *addtodate* pour l'insertion des valeurs d'horodatage dans la matrice *TS_OCLD*. Les valeurs du chronomètre correspondant aux mesures réalisées par la carte OCLD ont été enregistrées dans une seconde matrice appelée *Time_ms* (Voir Figure-A IX-1).

La figure-A IX-2 décrit un exemple de matrice Matlab qui indique le résultat de la conversion des fichiers CSV et de l'insertion des valeurs d'horodatage après que la seconde et la troisième étape de traitement aient été appliquées.

TS_OCLD <8738x9 double>									
	1	2	3	4	5	6	7	8	9
1	7.3518e+05	3.8890	1.6929	3.4737e-06	11.3450	11.5320	-0.0453	-0.5193	-0.8534
2	7.3518e+05	338.4500	51.2440	3.4328e-06	11.3450	11.5320	0.9777	-0.2053	0.0441
3	7.3518e+05	30.7130	14.5340	0.1170	11.3450	11.5320	0.4056	-0.8287	-0.3857
4	7.3518e+05	-1	-1	1	11.3450	11.5320	-0.4658	0.2944	-0.8345
5	7.3518e+05	-1	-1	1	11.3450	11.5320	-0.8802	-0.4201	0.2209
6	7.3518e+05	-1	-1	1	11.3450	11.5320	-0.2767	0.0965	-0.9561
7	7.3518e+05	-1	-1	1	11.3450	11.5320	-0.8716	0.3034	0.3852
8	7.3518e+05	-1	-1	1	11.3450	11.5320	0.9560	-0.2310	0.1808
9	7.3518e+05	-1	-1	1	11.3450	11.5320	0.6867	0.5222	0.5057
10	7.3518e+05	-1	-1	1	11.3450	11.5320	0.4443	-0.4439	-0.7782

Figure-A IX-2 Exemple d'un fichier Matlab après l'application de la troisième étape de traitement des données enregistrées avec le transpondeur cohérent

ANNEXE X

ÉTAPE 4 : SUPPRESSION DES VALEURS TRANSITOIRES CORRESPONDANT À LA RÉINITIALISATION DE LA CARTE OCLD

La figure-A X-1 présente la méthode de traitement appliquée sur la matrice TS_OCLD pour la suppression des valeurs transitoires enregistrées par la carte OCLD c'est-à-dire le transpondeur cohérent.

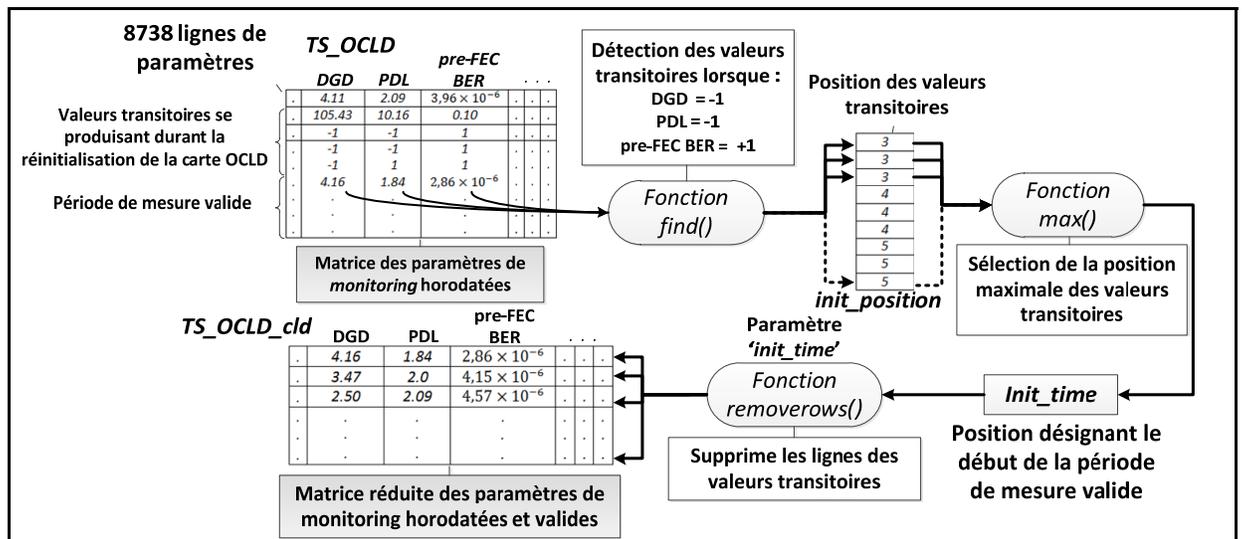


Figure-A X-1 Suppression des valeurs transitoires des matrices TS_OCLD durant la période de réinitialisation de la carte OCLD

La méthode de traitement décrite par la figure-A X-1 formalise le code Matlab intégré dans le script Matlab dédié au traitement des données enregistrées par la carte OCLD (Voir ANNEXE V, p. 125). La fonction *find* a tout d'abord été appliquée sur les matrices TS_OCLD pour la détection des valeurs transitoires sur les mesures de DGD, de PDL et de pre-FEC BER.

Le résultat de cette fonction se présente sous la forme d'une matrice, appelée *init_position*, constituée des positions des valeurs transitoires (Voir Figure-A X-1). La valeur maximale de la position *init_time*, obtenue par l'application de la fonction la fonction *max* sur la matrice *init_position*, définit la fin de la période de réinitialisation de la carte OCLD (Voir Figure-A X-1). La fonction *removerows*, paramétrée par la position désignant la fin de la période de réinitialisation *init_position*, a enfin été appliquée sur la matrice TS_OCLD pour la suppression de toutes les mesures réalisées jusqu'à la dernière valeur transitoire enregistrée par chaque matrice TS_OCLD .

La figure-A X-2 montre un exemple d'une matrice Matlab obtenue après la suppression des valeurs transitoires.

TS_OCLD_cld <8729x9 double>									
	1	2	3	4	5	6	7	8	9
1	7.3518e+05	1.6733	1.8270	5.2718e-06	11.7640	11.7930	-0.4068	-0.7895	-0.4596
2	7.3518e+05	1.4251	1.8407	5.2105e-06	11.7640	11.7930	-0.3934	-0.6977	-0.5988
3	7.3518e+05	1.3018	1.4447	4.6996e-06	11.7640	11.7930	-0.3604	0.1371	0.9227
4	7.3518e+05	1.8582	1.2604	4.7609e-06	11.7640	11.7930	-0.3441	0.9385	-0.0276
5	7.3518e+05	1.4043	1.2528	4.7609e-06	11.7640	11.7930	-0.3629	-0.0970	0.9268
6	7.3518e+05	1.9939	0.7940	5.1492e-06	11.7640	11.7930	-0.3079	-0.0814	0.9479
7	7.3518e+05	2.4970	1.1696	4.0867e-06	11.7640	11.7930	-0.3112	0.6977	0.6453
8	7.3518e+05	0.4833	0.7768	4.0049e-06	11.7640	11.7930	-0.2089	-0.3657	-0.9070
9	7.3518e+05	1.5329	0.4744	3.9028e-06	11.7640	11.7930	-0.1390	-0.9620	0.2353
10	7.3518e+05	1.4843	0.2970	3.9028e-06	11.7640	11.7930	-0.0090	0.9976	-0.0686

Figure-A X-2 Suppression des valeurs transitoires des matrices TS_OCLD durant la période de réinitialisation de la carte OCLD

ANNEXE XI

ÉTAPE 5 : CONCEPTION D'UNE STRUCTURE DE DONNÉES INDEXÉE PAR « CLÉ » DES MESURES DU TRANSPONDEUR COHÉRENT

La figure-A XI-1 présente le processus de conception de la structure de données *Struct_OCLD_data* composée des matrices *TS_OCLD_cld*. Celle-ci est une représentation du code Matlab intégré au script dédié au traitement de données enregistrées par la carte OCLD (Voir ANNEXE V, p. 125).

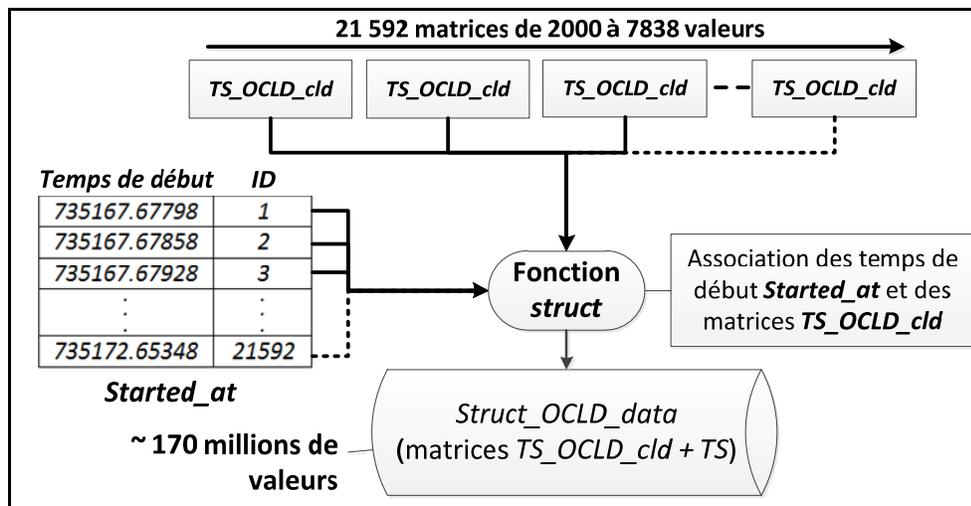


Figure-A XI-1 Diagramme de conception de la structure de données *Struct_OCLD_data*

Les identifiants uniques *ID* situés dans la deuxième colonne des matrices *Started_at* ont été associés aux identifiants des matrices *TS_OCLD_cld*. La fonction *struct* a été appliquée sur les matrices *TS_OCLD_cld* et *Started_at* pour l'association du temps de début avec les paramètres de *monitoring* lors de chaque nouvelle mesure réalisée par la carte OCLD. La figure-A XI-1 décrit le processus de conception de la structure de données *Struct_OCLD_data* constituées des paramètres de *monitoring* mises en structure. 21592 matrices *TS_OCLD_cld* et *Started_at* ont été converties en structure de données.

ANNEXE XII

CONCATÉNATION DES MATRICES DES VALEURS DE DGD ET DE PDL À PARTIR DE LA STRUCTURE DE DONNÉES INDEXÉE

Le script-A XII-1 a pour objectif la sélection des indices de matrices enregistrées dans la structure de données indexée par « clé » et la concaténation des valeurs de DGD et de PDL triés dans une matrice Matlab en fonction du temps de mesures réalisées avec le transpondeur cohérent DP-QPSK.

Script-A XII-1 Concaténation des valeurs de DGD et de PDL en fonction du temps dans la structure de données indexée par « clé »

```
1.  %%Retrieve the zie of all matrices stored in the OCLD data structure
2.  nFiles = length(Struct_OCLD_data);
3.
4.  %%Put the initial and the end of measurement period with the OCLD
5.  card
6.  init_TS = datenum(2012,11,05,11,0,0);
7.  end_TS = datenum(2012,11,11,0,0,0);
8.
9.
10. %%Creating the matrice for storing the ID matrix corresponding to
11. the
12. %%structure data
13.
14. Selected_ID = [];
15. for ind_file=1:nFiles
16.     try
17.         Started_at = Struct_OCLD_data(1, ind_file).started_at;
18.         %% Checking if the matrix corresponds to the measurement period
19.         if( Started_at >= init_TS && Started_at <=end_TS);
20.             Selected_ID(end+1) = ind_file;
21.         end
22.     catch
23.         fprintf('Pas de %d \n',ind_file);
24.         continue;
25.     end
26. end
27. %% Creating the a matrix to store the data corresponding to the
28. matrix
29. DGD_PDL_TS = [];
30.     %% DGD, PDL, and timetsamp concatenation
31.
32. for ind=1:length(Selected_ID)
33.     try
34.         matDGD = Struct_OCLD_data(1, ind).DGD;
35.         matPDL = Struct_OCLD_data(1, ind).PDL;
36.         matTS = Struct_OCLD_data(1, ind).TS;
```

```

37.
38.         size_data = length(matTS);
39.         DGD_PDL_TS(end+1:(end+size_data), 1:3) = [matTS, matDGD,
40. matPDL];
41.         catch
42.             fprintf('Pas de %d \n',ind); % Display the error ID File
43.             continue;
44.         end
45.     end
46.     %% Sort the data with the timestamp
47.     sortrows(DGD_PDL_TS,1);

```

Tout d'abord, le temps de début et le temps de fin ont été convertis au format Matlab par l'application de la fonction *datenum* (Voir lignes 6 et 7 du script-A XII-1). Une boucle *for* a ensuite été développée en vue de parcourir chaque matrice de la structure de données *Struct_OCLD_data*. Le temps de début de chaque fichier de mesure, appelée *Started_at*, a ensuite été extraite (Voir ligne 17 du script-A XII-1). L'identifiant de la matrice *ind_file* a été sélectionnée seulement si le temps de début du fichier de mesure *Started_at* extrait précédemment se situe entre le temps de début *init_TS* et le temps de fin *end_TS* de la période de mesure sélectionnée. Les lignes 19 à 21 permettent de sélectionner les identifiants des matrices correspondant à la période de mesure sélectionnées dans la matrice *Selected_ID*. (Voir script-A XII-1).

Une boucle *for* a été développée pour parcourir toutes les matrices précédemment sélectionnées (Voir ligne 32 du script-A XII-1). Les valeurs de *DGD*, les valeurs de *PDL* et les valeurs d'horodatage *TS* correspondantes ont été extraites à partir des matrices de la structure de données *Struct_OCLD_data* (Voir les lignes 34 à 36 du script-A XII-1). La concaténation des valeurs de *DGD*, de *PDL* et des temps est effectuée par la ligne 39 sur la matrice *DGD_PDL_TS*. Lorsque toutes les matrices ont été chargées et concaténées, toutes les valeurs de la matrice *DGD_PDL_TS* a été triée chronologiquement par l'application de la fonction *sortrows* sur les valeurs d'horodatage enregistrées dans la troisième colonne de la matrice *DGD_PDL_TS* (Voir ligne 47 du script-A XII-1).

ANNEXE XIII

CALCUL DE LA DÉRIVÉE TEMPORELLES DES VALEURS DE DGD ET DE PDL ENREGISTRÉES DANS LA STRUCTURE DE DONNÉES INDEXÉE

Le script-A XIII-1 présente le traitement et les calculs de dérivée temporelle des valeurs de DGD et de PDL concaténées dans la précédente matrice (Voir ANNEXE XII, p. 146).

Script-A XIII-1 Calcul de la dérivée temporelle des valeurs de DGD et des valeurs de PDL

```
1.  %% Comouiting the tive derivative function of the DGD and the PDL
2.  value
3.  %% recorded by the OCLD card by ensuring 1-ms step
4.  dt=addtodate(0,+1,'millisecond'); %% 1-ms gap
5.  size_Matrix= length(DGD_PDL_TS(:,1));
6.
7.  dDGD_dPDL_TS = [];
8.  start_period = 1;
9.  ind = 1;
10.
11.  while ind < (size_Matrix-1)
12.      ind = start_period;
13.
14.      % Moving the windows of the adjacent measurement...
15.      while (DGD_PDL_TS(ind+1,1)-DGD_PDL_TS(ind,1) == dt)
16.          && (ind < (size_Matrix-1))
17.              ind = ind +1;
18.          end
19.
20.      % Start and end period are now well defined...
21.      end_period = ind;
22.
23.      % Computing of time derivative of DGD and PDL
24.      dDGD = DGD_PDL_TS((start_period +1) : end_period, 2)...
25.              - DGD_PDL_TS((start_period : (end_period-1)),2);
26.      dPDL = DGD_PDL_TS ( (start_period +1) : end_period, 3)...
27.              - DGD_PDL_TS ( (start_period : (end_period-1)), 3);
28.      dTS = DGD_PDL_TS ( (start_period : (end_period-1)) , 1);
29.
30.      size_dparams = length(dTS);
31.
32.      % Concatenation of the values of time derivative of DGD and PDL
33.      dDGD_dPDL_TS (end+1: (end + size_dparams),1:3) =
34.      [dDGD,dPDL,dTS];
35.      start_period = end_period+1;
36.      clear dDGD dPDL dTS;
37.
38.  end
```

Tout d'abord, la période d'échantillonnage a été enregistrée dans la variable *dt*. Celle-ci est égale à 1 ms pour les mesures réalisées sur le lien optique du réseau Verizon et est égale à 1 s pour les mesures réalisées sur le lien optique du réseau Canarie. Cette valeur est calculée par l'application de la fonction *addtodate* (Voir ligne 5 du script-A XIII-1).

Ensuite, une première boucle *while* a été développée pour identifier une période de mesure continue sur les millions de valeurs enregistrées dans la matrice *DGD_PDL_TS*. L'indice correspondant au début de la période de mesure est enregistré dans la valeur *start_period*. L'indice *ind* est par la suite incrémenté tant que la durée séparant deux points est égale à *dt*, c'est-à-dire 1 ms ou 1 s dépendamment de la liaison sélectionnée. Lorsque la durée entre deux points est supérieure à *dt*, alors la fin de la période de la mesure continue est identifiée et l'indice correspondant est enregistré dans la valeur *end_period* (Voir lignes 13 à 23 du script-A XIII-1).

Les valeurs de dérivée temporelle de DGD et de PDL ont par la suite été calculées par la soustraction des mesures consécutives débutant à l'indice *start_period* jusqu'à l'indice *end_period* et ont été enregistrées dans les matrices *dDGD* et *dPDL* pour chaque période de mesure continue (Voir instructions 26 à 30 du script-A XIII-1).

Enfin, les matrices *dDGD*, *dPDL* et *TS* ont par la suite été concaténées dans la matrice *dDGD_dPDL_TS* regroupant toutes les fluctuations des valeurs de DGD et des valeurs de PDL sur toutes les périodes de mesure continues (Voir lignes 36 à 37 du script-A XIII-1).

ANNEXE XIV

AFFICHAGE DES COURBES D'ÉVOLUTION TEMPORELLE DES VALEURS DE DGD ET DES VALEURS DE PDL

Le script-A XIV-1 a pour l'affichage de la courbe d'évolution temporelle des valeurs de DGD et de PDL mesurées avec le transpondeur cohérent DP-QPSK à 40 Gbit/s.

Script-A XIV-1 Méthode d'affichage des valeurs de DGD et de PDL en fonction du temps

```
1. start_hour_temp = datenum(2012,11,5,0,0,0);
2. end_hour_temp = datenum(2012,11,11,12,0,0);
3.
4. start_datetick = datenum(2012,11,5,10,0,0);
5. end_datetick = datenum(2012,11,11,06,0,0);
6.
7. Number = 10;
8. % PAS = (end_hour_temp - start_hour_temp)/Number;
9. PAS = datenum(0,0,0,12,0,0);
10.
11. % Calcul du ber logarithmique
12. StringXlabel_sec = sprintf('Time %sSec/div',datestr(PAS,'SS'));
13. StringXlabel_min = sprintf('Time %sMin/div',datestr(PAS,'MM'));
14. StringXlabel_hour = sprintf('Time (%sh/div)',datestr(PAS,'HH'));
15. StringXlabel_hour_min = sprintf('Time
16. %sh%sMin/div',datestr(PAS,'HH'),datestr(PAS,'MM'));
17. StringXlabel_min_sec = sprintf('Time
18. %sMin%sSec/div',datestr(PAS,'MM'),datestr(PAS,'SS'));
19.
20. figure(1);
21.
22. subplot(2,1,1);
23.
24.
25. k0 = plot(TS_DGD_PDL_10ms(:,1),TS_DGD_PDL_10ms(:,2));
26. set(gca,'XLim',[start_datetick end_datetick]);
27. set(gca,'XTick',start_hour_temp:PAS:end_hour_temp);
28. XTick = get(gca,'XTick');
29. XTick = XTick(:);
30.
31. hx = get(gca,'XLabel'); % Handle to xlabel
32. %set(hx,'Units','data');
33.
34. x = get(hx,'Position');
35. y = x(2);
36. fs = get(gca,'fontsize');
37. for k=1:length(XTick)
38.     text(XTick(k)-0.1,-1.55,[datestr(start_hour_temp + (k-
39. 1)*PAS,'dd/mm'),sprintf('\n'),datestr(start_hour_temp+(k-
40. 1)*PAS,'dd/mm')]);
```

```

41. 1)*PAS, 'HH'), 'h'], 'fontsize', 18, 'fontname', 'calibri');
42. end
43. set(gca, 'XTickLabel', [])
44. set(gca, 'FontSize', 20);
45.
46. xlabel({}, 'FontName', 'Calibri', 'FontSize', 18);
47.
48. ylabel('DGD (ps)', 'FontName', 'Calibri', 'FontSize',
49. 24, 'FontWeight', 'bold', 'color', 'k');
50. set(gca, 'Position', [0.0431918008784773 0.605911330049261
51. 0.929721815519766 0.371082534981414], ...
52. 'FontSize', 20);
53. set(gca, 'YLim', [0 10.5]);
54. set(gca, 'YTick', (0:2:10));
55. figure(gcf)
56. grid on;
57. hold off;
58.
59. subplot(2,1,2);
60.
61.
62. k1 = plot(TS_DGD_PDL_10ms(:,1), TS_DGD_PDL_10ms(:,3), 'color', 'r');
63. set(gca, 'XLim', [start_datetick end_datetick]);
64. set(gca, 'XTick', start_hour_temp:PAS:end_hour_temp);
65.
66. XTick = get(gca, 'XTick');
67. XTick = XTick(:);
68. hx = get(gca, 'XLabel'); % Handle to xlabel
69. x = get(hx, 'Position');
70. y = x(2);
71. fs = get(gca, 'fontsize');
72.
73. for k=1:length(XTick)
74.     text(XTick(k)-0.1, -0.58, [datestr(start_hour_temp+(k-
75. 1)*PAS, 'dd/mm'), sprintf('\n'), datestr(start_hour_temp+(k-
76. 1)*PAS, 'HH')], 'h'], 'fontsize', 18, 'fontname', 'calibri');
77. end
78. set(gca, 'XTickLabel', [])
79. set(gca, 'Position', [0.0431918008784773 0.139573070607553
80. 0.930453879941435 0.35632183908046], 'FontSize', 20);
81. xlabel(StringXlabel_hour, 'FontName', 'Calibri', 'FontSize', 20);
82. ylabel('PDL (dB)', 'FontName', 'Calibri', 'FontSize',
83. 24, 'FontWeight', 'bold', 'color', 'k');
84. set(gca, 'YLim', [0 4.2]);
85. set(gca, 'YTick', (0:1:4));
86. figure(gcf)
87. grid on;
88. hold off;

```

ANNEXE XV

CALCUL DE LA MOYENNE ET DE L'ÉCART TYPE DES VALEURS DE DGD ET DE PDL À CHAQUE MINUTE DE MESURE

Script-A XV-1 Calcul de la moyenne et de l'écart type des valeurs de DGD et de PDL à chaque minute de mesure

```
1. min_datenum = min(DGD_PDL_TS(:,1));
2. max_datenum = max(DGD_PDL_TS(:,1));
3.
4. % min_datenum = datenum(2011,10,14,8,0,0);
5. % max_datenum = datenum(2011,11,18,8,0,0);
6.
7. %% Récupération des bornes limites temporelles
8.
9.
10. year_min_PMD = str2num(datestr(min_datenum, 'yyyy'));
11. month_min_PMD = str2num(datestr(min_datenum, 'mm'));
12. day_min_PMD = str2num(datestr(min_datenum, 'dd'));
13. hour_min_PMD = str2num(datestr(min_datenum, 'HH'));
14. minute_min_PMD = str2num(datestr(min_datenum, 'MM'));
15. second_min_PMD = str2num(datestr(min_datenum, 'MM'));
16.
17. year_max_PMD = str2num(datestr(max_datenum, 'yyyy'));
18. month_max_PMD = str2num(datestr(max_datenum, 'mm'));
19. day_max_PMD = str2num(datestr(max_datenum, 'dd'));
20. hour_max_PMD = str2num(datestr(max_datenum, 'HH'));
21. minute_max_PMD = str2num(datestr(max_datenum, 'MM'));
22. second_max_PMD = str2num(datestr(max_datenum, 'SS'));
23.
24. %% Mean and standard derivation of instantaneous DGD and PDL at
25. each minute
26. init_minute_date =
27. datenum(year_min,month_min,day_min,hour_min,minute_min,0);
28. end_minute_date =
29. datenum(year_max,month_max,day_max,hour_max,minute_max,0);
30.
31. minute_stats_DGD_PDL = [];
32. STEP_time = datenum(0,0,0,0,1,0);
33.
34. for minuteTS=init_minute_date:STEP_time:end_minute_date
35.     id = find(DGD_PDL_TS(:,1) >= minuteTS &
36.             DGD_PDL_TS(:,1) < (minuteTS + STEP_time));
37.
38.     if isempty(id)
39.         sprintf('Pas de valeurs minute PMD')
40.     else
41.         minute_stats_DGD_PDL(end+1,1:3)=[minuteTS...
```

```

42.         ,mean(DGD_PDL_TS(id_DGD,2)),mean(DGD_PDL_TS(id_DGD,3))...
43.         ,std(DGD_PDL_TS(id_DGD,2)),std(DGD_PDL_TS(id_DGD,3))];
44.     end
45.     id = [];
46. end

```

Tout d'abord, le temps de début *init_minute_time* et le temps de fin *end_minute_time* de la période de mesure sélectionnée sont définies jusqu'à la minute. La valeur *Step_time* définit une minute de mesure. Chacune de ces valeurs ont été créées par l'application de la fonction *datenum* qui converti les temps de mesure au format Matlab (*Voir* les lignes 26 à 29 du script-A XV-1).

Une boucle *for* a par la suite été développée pour la sélection des mesures de DGD et de PDL pour chaque minute de mesure de la période sélectionnée. L'incrémenter la valeur *minuteTS* est itérée à partir du temps de mesure initial, sous le nom de *init_minute_time* jusqu'à la valeur du temps de mesure final, nommée *end_minute_time* (*Voir* ligne 34 du script-A XV-1).

La variable *minuteTS* est le temps de mesure, incrémenté d'une durée de 1 min à chaque itération. Si des valeurs de DGD et de PDL de la matrice *DGD_PDL_TS* (*Voir* Figure 3.12) sont mesurées temporellement entre le temps *minuteTS* et le temps *minuteTS + Step_time*, alors les identifiants correspondants aux valeurs de DGD et de PDL sélectionnées sont enregistrées dans la matrice *Selected_id_data*. La recherche des valeurs de DGD et de PDL située sur une durée de 1 min à partir du temps de mesure *minuteTS* a été réalisée par l'application de la fonction *find* (*Voir* lignes 35 et 36 du script-A XV-1).

ANNEXE XVI

CALCUL DE LA DÉRIVÉE TEMPORELLE DE LA MOYENNE ET DE L'ÉCART TYPE DES VALEURS DE DGD ET DE PDL À CHAQUE MINUTE DE MESURE

Le script-A XVI-1 a pour objectif le calcul de la dérivée temporelle des moyennes et des écarts types des valeurs de DGD et de PDL précédemment calculées (Voir ANNEXE XV, p.152) à chaque minute de mesure.

Script-A XVI-1 Calcul des valeurs de dérivée temporelle de la moyenne et de l'écart type des valeurs de DGD et de PDL à chaque minute de mesure

```
1. dt=datenum(0,0,0,0,1,0); %% 1-min gap
2. size_Matrix= length(minute_stats_DGD_PDL(:,1));
3.
4. minute_derivate_stats_DGD_PDL = [];
5. start_period = 1;
6. ind = 1;
7.
8. while ind < (size_Matrix-1)
9.     ind = start_period;
10.    while ( minute_stats_DGD_PDL(ind+1,1) -
11. minute_stats_DGD_PDL(ind,1) == dt) && (ind < (size_Matrix-1))
12.        ind = ind +1;
13.    end
14.
15.    end_period = ind;
16.    size_mesure =
17. length(minute_stats_DGD_PDL(start_period:end_period,1))-1;
18.
19.
20.
21. minute_derivate_stats_DGD_PDL((end+1):(end+(size_mesure)),1:5) ...
22. = [minute_stats_DGD_PDL(start_period:(end_period-1),1), ...
23.    abs(minute_stats_DGD_PDL((start_period+1):end_period,2) -
24. minute_stats_DGD_PDL(start_period:(end_period-1),2)), ...
25.    abs(minute_stats_DGD_PDL((start_period+1):end_period,3) -
26. minute_stats_DGD_PDL(start_period:(end_period-1),3)), ...
27.    abs(minute_stats_DGD_PDL((start_period+1):end_period,4) -
28. minute_stats_DGD_PDL(start_period:(end_period-1),4)), ...
29.    abs(minute_stats_DGD_PDL((start_period+1):end_period,5) -
30. minute_stats_DGD_PDL(start_period:(end_period-1),5))];
31.
32.    start_period = end_period+1;
33. end
```


ANNEXE XVII

AFFICHAGE DE LA MOYENNE ET DE L'ÉCART TYPE DES VALEURS DE DGD ET DE PDL AINSI QUE LEURS VALEURS DE DÉRIVÉES TEMPORELLES RESPECTIVES POUR CHAQUE MINUTE DE MESURE

Le script-A XVII-1 effectue l'affichage des courbes d'évolution temporelle de la moyenne et de l'écart type des valeurs de DGD et de PDL en superposant leurs valeurs de dérivées temporelles respectives.

Script-A XVII-1 Affichage des courbes d'évolution temporelle de la moyenne et de l'écart type de DGD et de PDL ainsi que leurs valeurs de dérivées temporelles respectives

```
1. minute_stats_param_DGD_PDL =
2. sortrows(minute_stats_param_DGD_PDL(:,1:5),1);
3. derivate_minute_stats_param_DGD_PDL =
4. sortrows(derivate_minute_stats_param_DGD(:,1:5),1);
5.
6. start_hour_temp = datenum(2012,11,5,0,0,0);
7. end_hour_temp = datenum(2012,11,11,12,0,0);
8.
9. start_datetick = datenum(2012,11,5,10,0,0);
10. end_datetick = datenum(2012,11,11,06,0,0);
11.
12.
13. Number = 10;
14. PAS = datenum(0,0,0,12,0,0);
15.
16. StringXlabel_sec = sprintf('Time %sSec/div',datestr(PAS,'SS'));
17. StringXlabel_min = sprintf('Time %sMin/div',datestr(PAS,'MM'));
18. StringXlabel_hour = sprintf('Time (%sh/div)',datestr(PAS,'HH'));
19. StringXlabel_hour_min = sprintf('Time
20. %sh%sMin/div',datestr(PAS,'HH'),datestr(PAS,'MM'));
21. StringXlabel_min_sec = sprintf('Time
22. %sMin%sSec/div',datestr(PAS,'MM'),datestr(PAS,'SS'));
23.
24. figure(1);
25. subplot(2,1,1);
26. [AX,H1,H2]=
27. plotyy(minute_stats_param_DGD_PDL(:,1),minute_stats_param_DGD_PDL(:,
28. 2)...
29. ,derivate_minute_stats_param_DGD_PDL(1:(end-
30. 1),1),derivate_minute_stats_param_DGD_PDL(:,2));
31. set(AX(1),'XLim',[start_datetick end_datetick]);
32.
33. set(AX(1),'XTick',start_hour_temp:PAS:end_hour_temp);
34. set(AX(2),'XLim',[start_datetick end_datetick]);
35.
36. set(AX(2),'XTick',start_hour_temp:PAS:end_hour_temp);
37.
```

```

38.         XTick1 = get(AX(1), 'XTick');
39.         XTick2 = get(AX(2), 'XTick');
40.         XTick1 = XTick1(:);
41.         XTick2 = XTick2(:);
42.
43.         hx = get(gca, 'XLabel'); % Handle to xlabel
44.         x = get(hx, 'Position');
45.         y = x(2);
46.
47.         fs = get(gca, 'fontsize');
48.
49.         for k=1:length(XTick1)
50.             text(XTick1(k)-0.1, -
51. 3.5, [datestr(start_hour_temp+(k-1)*PAS, 'dd/mm'), sprintf('\n')...
52.             , datestr(start_hour_temp+(k-
53. 1)*PAS, 'HH'), 'h'], 'fontsize', 18, 'fontname', 'calibri');
54.         end
55.
56.
57.         set(AX(1), 'XTickLabel', []);
58.         set(AX(2), 'XTickLabel', []);
59.
60.         set(AX(1), 'box', 'off');
61.         set(AX(2), 'box', 'off');
62.
63.         set(gca, 'FontSize', 20);
64.         xlabel({}, 'FontName', 'Calibri', 'FontSize',
65. 18);
66.         axes(AX(1)); ylabel('<DGD> (ps)', 'FontName',
67. 'Calibri', 'FontSize', 24, 'FontWeight', 'bold', 'color', 'k');
68.         axes(AX(2)); ylabel('|d(DGD)/dt|
69. (ps/s)', 'FontName', 'Calibri', 'FontSize', 24...
70. , 'FontWeight', 'bold', 'color', 'k', 'rot', -90);
71.         set(gca, 'Position', [0.0431918008784773
72. 0.605911330049261 0.929721815519766 0.371082534981414], ...
73.         'FontSize', 20);
74.
75.         set(AX(1), 'YLim', [-2 10]);
76.         set(AX(1), 'YTick', (0:2:10));
77.         set(AX(2), 'YLim', [0 8]);
78.         set(AX(2), 'YTick', (0:1:4));
79.
80.         set(AX(2), 'FontSize', 20);
81.         set(AX(2), 'YColor', [0.0392156876623631
82. 0.141176477074623 0.415686279535294]);
83.         set(H2, 'color', [0.0392156876623631
84. 0.141176477074623 0.415686279535294]);
85.         figure(gcf)
86.         set(AX(1), 'Xgrid', 'on')
87.         %         set(AX(2), 'Ygrid', 'on')
88.         set(AX(1), 'Ygrid', 'on')
89.         hold off;
90.
91.
92. subplot(2,1,2);

```

```

93.             [AX,H1,H2]=
94. plotyy(minute_stats_param_DGD_PDL(:,1),minute_stats_param_DGD_PDL(:,
95. 3) ...
96.             ,derivate_minute_stats_param_DGD_PDL(1:(end-
97. 1),1),derivate_minute_stats_param_DGD_PDL(:,3));
98.             set(AX(1),'XLim',[start_datetick end_datetick]);
99.
100. set(AX(1),'XTick',start_hour_temp:PAS:end_hour_temp);
101.             set(AX(2),'XLim',[start_datetick end_datetick]);
102.
103. set(AX(2),'XTick',start_hour_temp:PAS:end_hour_temp);
104.
105.             set(gca,'XLim',[start_datetick end_datetick]);
106.
107. set(gca,'XTick',start_hour_temp:PAS:end_hour_temp);
108.
109.             XTick1 = get(AX(1),'XTick');
110.             XTick2 = get(AX(2),'XTick');
111.             XTick1 = XTick1(:);
112.             XTick2 = XTick2(:);
113.
114.             hx = get(gca,'XLabel'); % Handle to xlabel
115. %             set(hx,'Units','data');
116.             x = get(hx,'Position');
117.             y = x(2);
118.
119.             fs = get(gca,'fontsize');
120.
121.             for k=1:length(XTick1)
122.                 text(XTick1(k)-0.1,-
123. 1.0,[datestr(start_hour_temp+(k-1)*PAS,'dd/mm'),sprintf('\n')...
124.                 ,datestr(start_hour_temp+(k-
125. 1)*PAS,'HH'),'h'],'fontsize',18,'fontname','calibri');
126.             end
127.
128.
129.
130.             set(AX(1),'XTickLabel',[]);
131.             set(AX(2),'XTickLabel',[]);
132.
133.             set(AX(1),'box','off');
134.             set(AX(2),'box','off');
135.
136.             set(gca,'Position',[0.0431918008784773
137. 0.139573070607553 0.930453879941435 0.35632183908046],...
138.             'FontSize',20);
139.
140.
141.             xlabel(StringXlabel_hour,'FontName','Calibri',
142. 'FontSize',20);
143.             axes(AX(1)); ylabel('<PDL> (dB)','FontName',
144. 'Calibri','FontSize',24,'FontWeight','bold','color','k');
145.             axes(AX(2)); ylabel('|d(PDL)/dt|
146. (dB/s)','FontName','Calibri','FontSize',24,...
147.             'FontWeight','bold','color','k','rot',-90);

```

```

148.
149.         set (AX(1), 'YLim', [-0.5 4]);
150.         set (AX(1), 'YTick', (0:1:4));
151.         set (AX(2), 'YLim', [0 8]);
152.         set (AX(2), 'YTick', (0:1:2));
153.
154.         set (AX(1), 'FontSize', 20);
155.         set (AX(2), 'FontSize', 20);
156.
157.
158.         set (H1, 'color', 'r');
159.         set (AX(1), 'YColor', 'r');
160.
161.         set (H2, 'color', [0.694117665290833
162. 0.0156862754374743 0.0156862754374743]);
163.         set (AX(2), 'YColor', [0.694117665290833
164. 0.0156862754374743 0.0156862754374743]);
165.         figure(gcf)
166.         set (AX(1), 'Xgrid', 'on')
167. %         set (AX(2), 'Ygrid', 'on')
168.         set (AX(1), 'Ygrid', 'on')
169.         hold off;
170.
171.
172. figure(2);
173.
174. subplot(2,1,1);
175.         [AX,H1,H2]=
176. plotyy(minute_stats_param_DGD_PDL(:,1),minute_stats_param_DGD_PDL(:,
177. 4) ...
178.         ,derivate_minute_stats_param_DGD_PDL(1:(end-
179. 1),1),derivate_minute_stats_param_DGD_PDL(:,4));
180.         set (AX(1), 'XLim', [start_datetick end_datetick]);
181.
182. set (AX(1), 'XTick', start_hour_temp:PAS:end_hour_temp);
183.         set (AX(2), 'XLim', [start_datetick end_datetick]);
184.
185. set (AX(2), 'XTick', start_hour_temp:PAS:end_hour_temp);
186.
187.         XTick1 = get (AX(1), 'XTick');
188.         XTick2 = get (AX(2), 'XTick');
189.         XTick1 = XTick1(:);
190.         XTick2 = XTick2(:);
191.
192.
193.         hx = get (gca, 'XLabel'); % Handle to xlabel
194.         x = get (hx, 'Position');
195.         y = x(2);
196.
197.         fs = get (gca, 'fontsize');
198.
199.         for k=1:length(XTick1)
200.             text (XTick1(k)-0.1, -
201. 0.25, [datestr (start_hour_temp+(k-1)*PAS, 'dd/mm'), sprintf ('\n') ...
202.             ,datestr (start_hour_temp+(k-

```

```

203. 1)*PAS, 'HH'), 'h'], 'fontsize', 18, 'fontname', 'calibri');
204.         end
205.         set(AX(1), 'XTickLabel', []);
206.         set(AX(2), 'XTickLabel', []);
207.
208.         set(AX(1), 'box', 'off');
209.         set(AX(2), 'box', 'off');
210.
211.         set(gca, 'FontSize', 20);
212.
213.         xlabel({}, 'FontName', 'Calibri', 'FontSize',
214. 18);
215.         axes(AX(1)); ylabel('\sigma_D_G_D
216. (dB)', 'FontName', 'Calibri', 'FontSize', 24, ...
217.         'FontWeight', 'bold', 'color', 'k');
218.         axes(AX(2)); ylabel('|d(\sigma_D_G_D)/dt|
219. (ps/s)', 'FontName', 'Calibri', 'FontSize', 24, ...
220.         'FontWeight', 'bold', 'color', 'k', 'rot', -90);
221.         set(gca, 'Position', [0.0431918008784773
222. 0.605911330049261 0.929721815519766 0.371082534981414], ...
223.         'FontSize', 20);
224.
225.
226.         set(AX(1), 'YLim', [0 2.25]);
227.         set(AX(1), 'YTick', (0:0.5:2.25));
228.         set(AX(2), 'YLim', [0 4]);
229.         set(AX(2), 'YTick', (0:0.5:1));
230.
231.         set(AX(2), 'FontSize', 20);
232.         set(AX(2), 'YColor', [0.0392156876623631
233. 0.141176477074623 0.415686279535294]);
234.         set(H2, 'color', [0.0392156876623631
235. 0.141176477074623 0.415686279535294]);
236.         figure(gcf)
237.         set(AX(1), 'Xgrid', 'on')
238. %         set(AX(2), 'Ygrid', 'on')
239.         set(AX(1), 'Ygrid', 'on')
240.         hold off;
241.         subplot(2,1,2);
242.         [AX, H1, H2] =
243. plotyy(minute_stats_param_DGD_PDL(:,1), minute_stats_param_DGD_PDL(:,
244. 5) ...
245.         , derivate_minute_stats_param_DGD_PDL(1:(end-
246. 1),1), derivate_minute_stats_param_DGD_PDL(:,5));
247.
248.         set(AX(1), 'XLim', [start_datetick end_datetick]);
249.         set(AX(1), 'XTick', start_hour_temp:PAS:end_hour_temp);
250.         set(AX(2), 'XLim', [start_datetick end_datetick]);
251.
252.         set(AX(2), 'XTick', start_hour_temp:PAS:end_hour_temp);
253.
254.         set(gca, 'XLim', [start_datetick end_datetick]);
255.
256.         set(gca, 'XTick', start_hour_temp:PAS:end_hour_temp);
257.         XTick1 = get(AX(1), 'XTick');

```

```

258.         XTick2 = get (AX(2), 'XTick');
259.         XTick1 = XTick1(:);
260.         XTick2 = XTick2(:);
261.         hx = get(gca, 'XLabel'); % Handle to xlabel
262.         x = get(hx, 'Position');
263.         y = x(2);
264.         fs = get(gca, 'fontsize');
265.
266.         for k=1:length(XTick1)
267.             text(XTick1(k)-0.1, -
268. 0.35, [datestr(start_hour_temp+(k-1)*PAS, 'dd/mm'), sprintf('\n')...
269.             ,datestr(start_hour_temp+(k-
270. 1)*PAS, 'HH'), 'h'], 'fontsize', 18, 'fontname', 'calibri');
271.         end
272.         set(AX(1), 'XTickLabel', []);
273.         set(AX(2), 'XTickLabel', []);
274.
275.         set(AX(1), 'box', 'off');
276.         set(AX(2), 'box', 'off');
277.
278.         set(gca, 'Position', [0.0431918008784773
279. 0.139573070607553 0.930453879941435 0.35632183908046], ...
280.             'FontSize', 20);
281.
282.         xlabel(StringXlabel_hour, 'FontName', 'Calibri' ,
283. 'FontSize', 20);
284.         axes(AX(1)); ylabel('\{\sigma\}_P_D_L
285. (dB)', 'FontName', 'Calibri' , 'FontSize', 24, ...
286.             'FontWeight', 'bold', 'color', 'k');
287.         axes(AX(2)); ylabel('|d(\{\sigma\}_P_D_L)/dt|
288. (dB/s)', 'FontName', 'Calibri' , 'FontSize', 24, ...
289.             'FontWeight', 'bold', 'color', 'k', 'rot', -90);
290.
291.         set(AX(1), 'FontSize', 20);
292.         set(AX(2), 'FontSize', 20);
293.
294.         set(H1, 'color', 'r');
295.         set(AX(1), 'YColor', 'r');
296.
297.         set(H2, 'color', [0.694117665290833
298. 0.0156862754374743 0.0156862754374743]);
299.         set(AX(2), 'YColor', [0.694117665290833
300. 0.0156862754374743 0.0156862754374743]);
301.         set(AX(1), 'YLim', [-0.2 1]);
302.         set(AX(1), 'YTick', (0:0.2:1));
303.         set(AX(2), 'YLim', [0 2]);
304.         set(AX(2), 'YTick', (0:0.25:0.5));
305.         figure(gcf)
306.         set(AX(1), 'Xgrid', 'on')
307.         set(AX(1), 'Ygrid', 'on')
308.         hold off;
309.

```

ANNEXE XVIII

ÉTAPE 1 : CALCUL DE LA MOYENNE DES VALEURS DE DGD ET DE PDL POUR CHAQUE SECONDE DE MESURE

Le script-A XVIII-1 permet le calcul de la moyenne des valeurs de DGD et de PDL à chaque seconde de mesure et est exécuté sur la structure de données indexée par « clé ».

Script-A XVIII-1 Calcul de la moyenne des valeurs de DGD et de PDL pour chaque seconde de mesure

```
1.  %%Retrieve the size of all matrices stored in the OCLD data
2.  structure
3.  nFiles = length(Struct_OCLD_data);
4.
5.  %%Put the initial and the end of measurement period with the OCLD
6.  card
7.  init_TS = datenum(2012,11,05,11,0,0);
8.  end_TS = datenum(2012,11,11,0,0,0);
9.
10. %%Creating the matrice for storing the ID matrix corresponding to
11. the
12. %%structure data
13.
14. Selected_ID = [];
15. DGD_per_sec = [];
16. PDL_per_sec = [];
17. mean_sec_DGD_PDL = [];
18.
19. for ind_file=1:nFiles
20.     try
21.         Started_at = Struct_OCLD_data(1, ind_file).started_at;
22.         %% Checking if the matrix corresponds to the measurement period
23.         if( Started_at >= init_TS && Started_at <=end_TS);
24.             Selected_ID(end+1) = ind_file;
25.         end
26.     catch
27.         fprintf('Pas de %d \n',ind_file);
28.         continue;
29.     end
30. end
31. end
32.
33. for ind_file=1:nFiles
34.
35.     % Computing mean of instantaneous DGD and PLD at each second
36.     min_datenum = min(Struct_OCLD_data(1, ind_file).TS(:,1));
37.     max_datenum = max(Struct_OCLD_data(1, ind_file).TS(:,1));
38.
39.     %% Récupération des bornes limites temporelles
40.
```

```

41.     year_min = str2num(datestr(min_datenum, 'yyyy'));
42.     month_min = str2num(datestr(min_datenum, 'mm'));
43.     day_min = str2num(datestr(min_datenum, 'dd'));
44.     hour_min = str2num(datestr(min_datenum, 'HH'));
45.     minute_min = str2num(datestr(min_datenum, 'MM'));
46.     second_min = str2num(datestr(min_datenum, 'SS'));
47.
48.     year_max = str2num(datestr(max_datenum, 'yyyy'));
49.     month_max = str2num(datestr(max_datenum, 'mm'));
50.     day_max = str2num(datestr(max_datenum, 'dd'));
51.     hour_max = str2num(datestr(max_datenum, 'HH'));
52.     minute_max = str2num(datestr(max_datenum, 'MM'));
53.     second_max = str2num(datestr(max_datenum, 'SS'));
54.
55.     % 2 algorithmes similaires qui permettront de récupérer les
56.     moyennes et les écart-types à la seconde t à la minute sur les
57.     clean_tab_mesure_DGD
58.
59.     % Statistical instantaneous second DGD and PDL
60.     init_second_date =
61.     datenum(year_min, month_min, day_min, hour_min, minute_min, second_min);
62.     end_second_date =
63.     datenum(year_max, month_max, day_max, hour_max, minute_max, second_max);
64.
65.     second_stats_DGD_PDL = [];
66.     STEPsec = datenum(0,0,0,0,1,0);
67.
68.
69.     for secondTS=init_second_date:STEPsec:end_second_date
70.
71.         id = find(Struct_OCLD_data(1, ind_file).TS(:,1) >= secondTS
72. & Struct_OCLD_data(1, ind_file).TS(:,1) < (secondTS + STEPsec));
73.
74.         if(isempty(id))
75.             sprintf('Pas de valeurs minute PMD')
76.         else
77.             mean_sec_DGD_PDL(end+1,1:3)=[secondTS...
78.                 ,mean(Struct_OCLD_data(1,
79. ind_file).DGD(id,1))...
80.                 ,mean(Struct_OCLD_data(1,
81. ind_file).PDL(id,1))];
82.             end
83.
84.             id = [];
85.         end
86.
87.
88.     end

```

ANNEXE XIX

ÉTAPE 2 : FILTRAGE SUPER GAUSSIEN DES VALEURS DE <DGD> ET DE <PDL> POUR CHAQUE SECONDE DE MESURE

Le script-A XIX-1 effectue le filtrage super gaussien d'ordre 4 appliqué sur les valeurs de <DGD> et <PDL>.

Script-A XIX-1 Filtrage super gaussien des valeurs de <DGD> et de <PDL> mesurées par le transpondeur cohérent

```
1.  % Remplacement des valeurs / triage
2.  mean_sec_DGD_PDL = sortrows(mean_sec_DGD_PDL(:,1:3),1);
3.
4.  % FFT absolu
5.  FFT_DGD_PDL = ...
6.  [fft(mean_sec_DGD_PDL(:,2),length(mean_sec_DGD_PDL(:,2))),...
7.   fft(mean_sec_DGD_PDL(:,3),length(mean_sec_DGD_PDL(:,3)))];
8.
9.  % Filtre super gaussien / définition
10. x_DGD = 0:(length(mean_sec_DGD_PDL(:,2),1)-1);
11. x_PDL = 0:(length(mean_sec_DGD_PDL(:,3),1)-1);
12.
13. sigma=500;
14. Filtre_SuperGauss_DGD = exp((-4)*(x_DGD/sigma).^4);
15. Filtre_SuperGauss2_DGD = transpose(Filtre_SuperGauss_DGD);
16.
17. Filtre_SuperGauss_PDL = exp((-4)*(x_PDL/sigma).^4);
18. Filtre_SuperGauss2_PDL = transpose(Filtre_SuperGauss_PDL);
19.
20.
21. % Convolution du filtre Super Gaussien et
22. Z_filtered_DGD_PDL = ...
23. [(FFT_DGD_PDL(:,1)*Filtre_SuperGauss2_DGD(:,1)),...
24.  (FFT_DGD_PDL(:,2)*Filtre_SuperGauss2_PDL(:,1))];
25.
26.           % Calcul de l'IFFT
27. IFFT_filtered_DGD_PDL = ...
28. ifft(Z_filtered_DGD_PDL(:,1),length(Z_filtered_DGD_PDL(:,1))),...
29. ifft(Z_filtered_DGD_PDL(:,2),length(Z_filtered_DGD_PDL(:,2)))]);
30.
31. filtered_DGD_PDL = [real(IFFT_filtered_DGD_PDL(:,1)),...
32.                    real(IFFT_filtered_DGD_PDL(:,2))]
33.
34. clear Filtre_SuperGauss_DGD Filtre_SuperGauss2_DGD
35. Filtre_SuperGauss_PDL Filtre_SuperGauss2_PDL Z_filtered_DGD_PDL
36. filtered_DGD_PDL;
37.
```


ANNEXE XX

ÉTAPE 3 : CALCUL DE LA FONCTION D'AUTOCORRÉLATION DES VALEURS DE <DGD> FILTRÉES ET DES VALEURS DE <PDL> FILTRÉES

Le script-A XX-1 effectue le calcul de la fonction d'autocorrélation normalisée sur les valeurs de <DGD> et de <PDL> après filtrage super gaussien (Voir ANNEXE XIX, p.163).

Script-A XX-1 Calcul de la fonction d'autocorrélation normalisée des valeurs de <DGD> et de <PDL> filtrées

```
1. % Calcul de la fonction l'autocorrelation biaisée
2. size_ACF = length(mean_sec_DGD_PDL(:,1),1);
3. ACF_DGD_PDL = zeros(size_ACF,3);
4.
5. ACF_DGD_PDL = ...
6. [xcov(filtered_DGD_PDL(:,1),...
7.     floor(length(filtered_DGD_PDL(:,1))/2), 'biased'), ...
8.     xcov(filtered_DGD_PDL(:,2), ...
9.         floor(length(filtered_DGD_PDL(:,2))/2), 'biased')];
10.
11.     % Normalization...
12. ACF_DGD_PDL = [ACF_DGD_PDL(:,1) ./ var(filtered_DGD_PDL(:,1)), ...
13.               ACF_DGD_PDL(:,2) ./ var(filtered_DGD_PDL(:,2))];
14.
```


ANNEXE XXI

ÉTAPE 4 : CALCUL DU TEMPS DE CORRÉLATION DES VALEURS DE <DGD> ET DE <PDL> FILTRÉES

Le script-A XXI-1 effectue le calcul du temps de corrélation sur les valeurs de fonction d'autocorrélation normalisée des valeurs de <DGD> et de <PDL> filtrées.

Script-A XXI-1 Calcul de la fonction d'autocorrélation normalisée
des valeurs de <DGD> et de <PDL> filtrées

```
1.  % Detection coherente des temps de correlations
2.
3.  tab_correlation_time_DGD=[];
4.  tab_correlation_time_PDL=[];
5.  indice=1;
6.
7.  correlation_time=0.5;
8.  correlation_time2=0.63;
9.  correlation_time3=0.37;
10.
11. %DETECTIONTEMPSCORRELATIONPMD
12.
13.
14. day=str2num(datestr(mean_sec_DGD_PDL(1,1), 'dd'));
15. month=str2num(datestr(mean_sec_DGD_PDL(1,1), 'mm'));
16. year=str2num(datestr(mean_sec_DGD_PDL(1,1), 'yyyy'));
17. tab_correlation_time_DGD(1,1:3)=[day,month,year];
18.
19. for ind=1:(length(ACF_DGD_PDL(:,1))-1)
20.
21.     if(ACF_DGD_PDL(ind,1)<correlation_time2...
22.         &&ACF_DGD_PDL(ind+1,1)>correlation_time2)
23.         first_datenum_cste=mean_sec_DGD_PDL(ind);
24.     end
25.
26.     if(ACF_DGD_PDL(ind,1)<correlation_time...
27.         &&ACF_DGD_PDL(ind+1,1)>correlation_time)
28.         first_datenum_FWHM=mean_sec_DGD_PDL(ind);
29.     end
30.
31.     if(ACF_DGD_PDL(ind,1)>correlation_time...
32.         &&ACF_DGD_PDL(ind+1,1)<correlation_time)
33.         second_datenum_FWHM=mean_sec_DGD_PDL(ind);
34.     end
35.
36.
37.
38.     if(ACF_DGD_PDL(ind,1)>correlation_time3...
```

```

39.         &&ACF_DGD_PDL(ind+1,1)<correlation_time3)
40.         second_datenum_cste=mean_sec_DGD_PDL(ind);
41.     end
42.
43. end
44.
45. fwhm_time=second_datenum_FWHM-first_datenum_FWHM;
46. cste_time=second_datenum_cste-first_datenum_cste;
47. tab_correlation_time_DGD(1,4:5)=[fwhm_time,cste_time];
48.
49. %DETECTION TEMPS CORRELATION PDL
50.
51. day=str2num(datestr(mean_sec_DGD_PDL(1,1),'dd'));
52. month=str2num(datestr(mean_sec_DGD_PDL(1,1),'mm'));
53. year=str2num(datestr(mean_sec_DGD_PDL(1,1),'yyyy'));
54. tab_correlation_time_PDL(1,1:3)=[day,month,year];
55.
56.
57. for ind=1:(length(ACF_DGD_PDL(:,2))-1)
58.
59.     if(ACF_DGD_PDL(ind,2)<correlation_time2...
60.         &&ACF_DGD_PDL(ind+1,2)>correlation_time2)
61.         first_datenum_cste=mean_sec_DGD_PDL(ind);
62.     end
63.
64.     if(ACF_DGD_PDL(ind,2)<correlation_time...
65.         &&ACF_DGD_PDL(ind+1,2)>correlation_time)
66.         first_datenum_FWHM=mean_sec_DGD_PDL(ind);
67.     end
68.
69.     if(ACF_DGD_PDL(ind,2)>correlation_time...
70.         &&ACF_DGD_PDL(ind+1,2)<correlation_time)
71.         second_datenum_FWHM=mean_sec_DGD_PDL(ind);
72.     end
73.
74.     if(ACF_DGD_PDL(ind,2)>correlation_time3...
75.         &&ACF_DGD_PDL(ind+1,2)<correlation_time3)
76.         second_datenum_cste=mean_sec_DGD_PDL(ind);
77.     end
78.
79.
80. end
81. fwhm_time=second_datenum_FWHM-first_datenum_FWHM;
82. cste_time=second_datenum_cste-first_datenum_cste;
83. tab_correlation_time_PDL(1,4:5)=[fwhm_time,cste_time];
84.

```

ANNEXE XXII

AFFICHAGE ET SUPERPOSITION DES FONCTIONS D'AUTOCORRELATIONS DE DGD ET DE PDL SUR LA LIAISON DU RÉSEAU CANARIE ET VERIZON

Le script-A XXII-1 optimise l'affichage et la superposition des valeurs de fonctions d'autocorrélation de <DGD> et de <PDL> en fonction du temps.

Script-A XXII-1 Affichage des fonctions d'autocorrélation des valeurs de DGD et de PDL ainsi que leur temps de corrélation respectif

```
1.  % Gestion des timestamp relatifs de la liaison Canarie
2.
3.  ACF_DGD_can = [ACF_DGD_PDL_can(:,1), ACF_DGD_PDL_can(:,2)];
4.  ACF_PDL_can = [ACF_DGD_PDL_can(:,1), ACF_DGD_PDL_can(:,3)];
5.  ACF_DGD_vz = [ACF_DGD_PDL_vz(:,1), ACF_DGD_PDL_vz(:,2)];
6.  ACF_PDL_vz = [ACF_DGD_PDL_vz(:,1), ACF_DGD_PDL_vz(:,3)];
7.
8.  [ACFmax_canarie,loc_canarie] = max(ACF_DGD_can(:,2));
9.  TS_acf_max1 = ACF_DGD_can(loc_canarie,1);
10. TS_canarie_DGD = ACF_DGD_can(1:end,1)-TS_acf_max1; %% Translation
11. relative du temps
12.
13. [ACFmax_canarie,loc_canarie] = max(ACF_PDL_can(:,2));
14. TS_acf_max1 = ACF_PDL_can(loc_canarie,1);
15. TS_canarie_PDL = ACF_PDL_can(:,1)-TS_acf_max1;
16.
17. [ACFmax_wc,loc_vz] = max(ACF_DGD_vz(:,2));
18. TS_acf_max1 = ACF_DGD_vz(loc_vz,1);
19. TS_vz_DGD = ACF_DGD_vz(:,1)-TS_acf_max1;
20.
21. [ACFmax_wc,loc_vz] = max(ACF_PDL_vz(:,2));
22. TS_acf_max1 = ACF_PDL_vz(loc_vz,1);
23. TS_vz_PDL = ACF_PDL_vz(:,1)-TS_acf_max1;
24.
25.
26. bound_hour_temp = datenum(0,0,2,12,0,0);
27. Number = 10;
28. PAS = datenum(0,0,0,12,0,0);
29.
30. StringXlabel_sec = sprintf('Time %sSec/div',datestr(PAS,'SS'));
31. StringXlabel_min = sprintf('Time %sMin/div',datestr(PAS,'MM'));
32. StringXlabel_hour = sprintf('Time %sh/div',datestr(PAS,'HH'));
33. StringXlabel_hour_min = sprintf('Time
34. %sH%sMin/div',datestr(PAS,'HH'),datestr(PAS,'MM'));
35. StringXlabel_min_sec = sprintf('Time
36. %sMin%sSec/div',datestr(PAS,'MM'),datestr(PAS,'SS'));
37.
38. %% Affichage Temps de corrélation
39.
40.
```

```

41. figure(1);
42.
43. k1 = area(TS_canarie_DGD(:,1),ACF_DGD_can(1:(end-1),3),...
44. 'FaceColor',[0.937254905700684 0.866666674613953
45. 0.866666674613953],...
46. 'EdgeColor',[0.847058832645416 0.160784319043159 0]);
47.
48. hold on;
49.
50. k2 = area(TS_vz_DGD(:,1),ACF_DGD_vz(1:(end-1),3),...
51. 'FaceColor',[0.533333361148834 0.752941191196442 1],...
52. 'EdgeColor',[0 0 1]);
53.
54. hold off;
55.
56. XTick = get(gca,'XTick');
57. XTick = XTick(:);
58.
59.
60. hx=get(gca,'XLabel');%Handletoxlabel
61. x=get(hx,'Position');
62. y=x(2);
63.
64. fs=get(gca,'fontsize');
65.
66. set(gca,'XLim',[-bound_hour_temp+bound_hour_temp]);
67. set(gca,'XTick',-bound_hour_temp:PAS:bound_hour_temp);
68. set(gca,'XTickLabel',{'2.5','2','1.5','1','0.5','0','0.5','1','1.5',
69. '2','2.5'},'FontName','Calibri','fontSize',24);
70.
71. M=[k1 k2];
72.
73.
74. legend(M...
75. ,['{\bfCorrelationtimeonburiedlink:}{\tau_F_W_H_M}=',...
76. datestr(tab_correlation_time_DGD_can(1,4),'HH:MM')]...
77. ,['{\bfCorrelationtimeonaerialLink:}{\tau_F_W_H_M}=',...
78. datestr(tab_correlation_time_DGD_vz(1,4),'HH:MM')]');
79.
80. set(gca,'FontSize',20,'FontName','Calibri');
81. xlabel('Days','FontName','Calibri','FontSize',24);
82. ylabel('ACFDGD','FontName','Calibri','FontSize',24,'FontWeight','bol
83. d','color','k');
84. set(gca,'YLim',[-0.21]);
85. set(gca,'YTick',(-0.3:0.1:1));
86. figure(gcf)
87. grid on;
88. hold off;
89.
90. figure(2);
91.
92. h1=area(TS_canarie_PDL(:,1),ACF_PDL_can(:,3),...
93. 'FaceColor',[0.9372549057006840.8666666746139530.866666674613953],...
94. .
95. 'EdgeColor',[0.8470588326454160.1607843190431590]);

```

```

96.
97. hold on;
98.
99. h2=area(TS_vz_PDL(:,1),ACF_PDL_vz(1:(end-1),3),...
100. 'FaceColor',[0.5333333611488340.7529411911964421],...
101. 'EdgeColor',[001]);
102. hold off;
103.
104. XTick=get(gca,'XTick');
105. XTick=XTick(:);
106.
107.
108. hx=get(gca,'XLabel');%Handletoxlabel
109. x=get(hx,'Position');
110. y=x(2);
111.
112. fs=get(gca,'fontsize');
113.
114. set(gca,'XLim',[-bound_hour_temp+bound_hour_temp]);
115. set(gca,'XTick',-bound_hour_temp:PAS:bound_hour_temp);
116. set(gca,'XTickLabel',{'2.5','2','1.5','1','0.5','0','0.5','1','1.5',
117. '2','2.5'},'FontName','Calibri','fontSize',24);
118.
119. M=[h1 h2];
120.
121. legend(M...
122. ,['{\bfCorrelationtimeonburiedlink:}{\tau_F_W_H_M}=',...
123. datestr(tab_correlation_time_PDL_can(1,4),'HH:MM')]...
124. ,['{\bfCorrelationtimeonaerialLink:}{\tau_F_W_H_M}=',...
125. datestr(tab_correlation_time_PDL_vz(1,4),'HH:MM')]);
126.
127.
128. set(gca,'FontSize',24,'FontName','Calibri');
129. xlabel('Days','FontName','Calibri','FontSize',24);
130. ylabel('ACFPDL','FontName','Calibri','FontSize',24,'FontWeight','bold',
131. 'color','k');
132. set(gca,'YLim',[-0.21]);
133. set(gca,'YTick',(-0.3:0.1:1));
134. figure(gcf)
135. grid on;
136. hold off;

```


ANNEXE XXIII

RÉSULTAT DE LA FONCTION D'AUTOCORRÉLATION APPLIQUÉE SUR LES VALEURS DE $\langle PDL \rangle$ FILTRÉES

La figure-A XXIII-1 présente les courbes de fonction d'autocorrélation normalisée des valeurs de $\langle PDL \rangle$ mesurées avec le transpondeur cohérent sur la liaison majoritairement enterrée du réseau Canarie et sur la liaison majoritairement aérien du réseau Verizon.

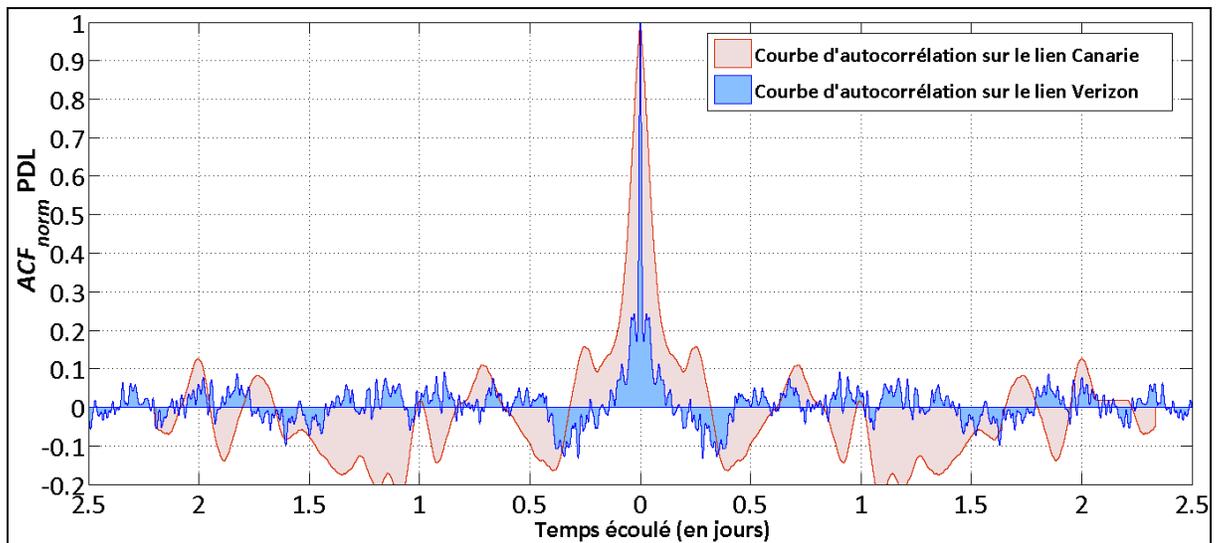


Figure-A XXIII-1 Fonctions d'autocorrélation ACF_{norm} des valeurs de $\langle PDL \rangle$ filtrées sur la liaison du réseau Canarie et sur la liaison du réseau Verizon durant 5 jours

ANNEXE XXIV

CALCUL ET AFFICHAGE DE LA FONCTION DE DENSITÉ DE PROBABILITÉ DES VALEURS DE DGD MESURÉES AVEC LE TRANSPONDEUR COHÉRENT

Le script-A XXIV-1 effectue le calcul et l'affichage de la fonction de densité de probabilité des valeurs de DGD mesurées avec le transpondeur cohérent et enregistrées dans la structure de données indexée par « clé ».

Script-A XXIV-1 Calcul et affichage de la fonction de densité des valeurs de DGD

```
1.  %%Retrieve the size of all matrices stored in the OCLD data
2.  structure
3.  nMatrix = length(Struct_OCLD_data);
4.
5.  %%Preparation des matrices de distributions
6.  countMatrix_DGD= [0,0,0,0];
7.  DGD_stats = [];
8.
9.  %%On remplit ensuite la première colonne des valeurs de PMd à
10. scanné....
11. size_histDGD = length(countMatrix_DGD(:,1));
12. cellule_DGD = 1;
13. ValDGDScan = 0;
14. maxDGD = 0;
15.
16. bin_size = 0.01;
17.
18. %%Chargement des valeurs dans la matrice de distributions
19.
20.
21. for k=1:nMatrix
22.
23.
24. DGD = Struct_OCLD_data(1, k).DGD;
25.
26. %% Mean and standard variation for each matrix
27.
28.
29. DGD_stats(end+1,1:2) = [mean(DGD(:,1)),std(DGD(:,1))];
30.
31. if(max(DGD(:,1))>maxDGD)
32.     max_bins=floor(max(DGD(:,1))*(1/bin_size))+1;
33.     maxDGD = max(DGD(:,1));
34.     temp_countDGD =...
35.
36. transpose(roundn((length(countMatrix_DGD(:,1))+1):max_bins)...
37.     *bin_size,-2));
38.
39. countMatrix_DGD((end+1):(end+length(temp_countDGD)),1:4) =...
40. [temp_countDGD,
```

```

41. zeros(length(temp_countDGD(:,1)),3)];
42.
43. end
44.
45. %% Counting the occurrences of DGD
46.
47. for index = 1 : length(DGD)
48.     valDGD = DGD(index,1);
49.     idDGD=floor(valDGD*(1/bin_size))+1;
50.     countMatrix_DGD(idDGD,2)=countMatrix_DGD(idDGD,2)+1;
51. end
52.
53.
54. % IDdgd = round(roundn(DGD(:,1),-2)*(1/bin_size))+1;
55. %
56. %     for cell = 1:length(IDdgd)
57. %         countMatrix_DGD(IDdgd(cell),2) =
58. countMatrix_DGD(IDdgd(cell),2)+1;
59. %     end
60.
61.     DGD = [];
62. end
63.
64. %% Computing the probability values of DGD
65. Sum_Prob_DGD = sum(countMatrix_DGD(:,2));
66. countMatrix_DGD(:,3) = countMatrix_DGD(:,2)/Sum_Prob_DGD;
67.
68. %% Deleting the 0 probability values
69. id_pos_DGD = find(countMatrix_DGD(:,3)~=0);
70. countMatrix_DGD_pos = countMatrix_DGD(id_pos_DGD,:);
71.
72.
73. % DGD histogram normalization
74. countMatrix_DGD(:,4) = countMatrix_DGD(:,3)*...
75.     (1/trapz(countMatrix_DGD_pos(:,1),countMatrix_DGD_pos(:,3)));
76. countMatrix_DGD_pos(:,4) = countMatrix_DGD_pos(:,3)*...
77.     (1/trapz(countMatrix_DGD_pos(:,1),countMatrix_DGD_pos(:,3)));
78.
79. % HISTOGRAMME DGD
80. figure(1)
81.
82. h1 = bar(countMatrix_DGD(:,1),countMatrix_DGD(:,4),...
83.     'b','BaseValue',min(countMatrix_DGD_pos(:,4))/2);
84. %set(gca,'YDir','reverse')
85. hold on;
86.
87. MeanPMD=mean(DGD_stats(:,1));
88. StdPMD=mean(DGD_stats(:,2));
89.
90. legend(['\bfStatistical Parameters :} Mean =',
91. num2str(MeanPMD),...
92. ' ps Standard deviation ({\sigma}) = ', num2str(StdPMD), '
93. ps']);
94.
95.

```

```

96. ylim([min(countMatrix_DGD_pos(:,4))/2
97. max(countMatrix_DGD_pos(:,4))*2]);
98. xlim([-0.01 15]);
99.
100. xlabel('PMD [ps]', 'fontsize', 20, 'fontweight', 'b')
101. ylabel('PDF ', 'fontsize', 20, 'fontweight', 'b')
102. set(gca, 'FontSize', 14, 'fontName', 'Calibri');
103. set(gca, 'YScale', 'log')
104. set(gca, 'YMinorTick', 'on');
105. set(gca, 'XTick', 0:1:15);
106. set(gca, 'Position', [0.0609375 0.0893939393939394 0.92890625
107. 0.886363636363636]);
108. set(gca, 'FontSize', 20, 'FontName', 'Calibri');
109. figure(gcf)
110. hold off;

```

La matrice qui contient les valeurs de classe, d'occurrence, de probabilité et de densité de DGD est construite progressivement durant cette méthode de traitement. La première étape consiste à augmenter la taille de celle-ci de manière dynamique en fonction des valeurs maximales de DGD mesurées (Voir lignes 31-43 du script-A XXIV-1). Le script compte les valeurs d'occurrences pour chaque classe à chaque chargement d'une nouvelle matrice *TS_OCLD_cld* (Voir lignes 49 à 53 du script-A XXIV-1). Le comptage des valeurs de DGD pour chaque classe s'effectue au moyen du calcul d'un indice correspondant à chaque valeur de DGD mesurée, appelée ID_{DGD} . La valeur d'occurrence correspondant à l'indice ID_{DGD} est incrémentée dans la matrice *countMatrix_DGD*. L'équation (3.9) présente le calcul des valeurs d'indice développées en vue d'optimiser la recherche des valeurs d'occurrences pour chaque classe de DGD.

$$ID_{DGD} = DGD \times \frac{1}{(bins\ size)} + 1 \quad (\text{A XXIV-1})$$

La valeur de DGD est multipliée par l'inverse de la résolution de l'histogramme, nommée *bins size*, puis additionné de +1 en vue d'ajuster la valeur d'indice ID_{DGD} aux valeurs d'occurrences de la matrice *countMatrix_DGD*. Le script-A XXIV-1 présente le traitement appliqué sur la structure de données *Struct_OCLD_data* en vue de l'obtention de la matrice *countMatrix_DGD*. Les valeurs de moyenne et d'écart type des valeurs de DGD de chaque matrice *TS_OCLD_cld* ont été calculées et sauvegardés dans la matrice *stats_DGD_stats* (Voir ligne 29 du script-A XXIV-1). Le résultat de ce script se présente sous la forme d'une matrice constituée de quatre colonnes : la répartition en classes des valeurs de DGD est enregistrée dans la première colonne, les valeurs d'occurrences sont stockées dans la deuxième colonne, les valeurs de probabilités et les valeurs de densités sont enregistrées dans la troisième colonne et dans la quatrième colonne respectivement.

ANNEXE XXV

AJUSTEMENT DE LA FONCTION DE DENSITÉ DE PROBABILITÉ DES VALEURS DE DGD PAR LA FONCTION DE DENSITÉ DE MAXWELL

Le script-A XXIV-1 effectue l'ajustement de la fonction de densité de probabilité des valeurs de DGD par la fonction de densité de Maxwell.

Script-A XXV-1 Calcul et évaluation de l'ajustement de la fonction de densité des valeurs de DGD par la fonction de densité de Maxwell

```
1. % Weight computing...
2. N = sum(countMatrix_DGD_pos(:,2));
3. p = countMatrix_DGD_pos(:,2);
4. H = countMatrix_DGD_pos(:,2);
5. sigma2_lin = p./N;
6. sigma_lin = sqrt(sigma2_lin);
7. weights_lin = 1./sigma2_lin;
8. mean_weight_lin = mean(weights_lin);
9. W_linear_relative = weights_lin / mean_weight_lin;
10.
11. %%%%%%%%%%%%% LINEAR FIT MAXWELLIAN
12.
13. alpha_th2 = var(countMatrix_DGD(:,4))/(3-(8/pi)); % Estimate alpha
14. of new density normalization
15. alpha_th = sqrt(alpha_th2);
16.
17.
18. x_fm = countMatrix_DGD_pos(:,1);
19. y_fm = countMatrix_DGD_pos(:,4); %density
20.
21. modelFun_fm = @(alpha,x) (sqrt(2/pi)) * (x.^2/alpha^3) .* exp(-
22. (x.^2)/(2*(alpha^2)));
23.
24. optionsLSQ = optimset('Display','Iter','TolFun',1e-250,'TolX',1e-
25. 250,'MaxFunEvals',500000,'MaxIter',500000);
26.
27. [alpha_nlinfit,Residual_fm,Jacob_fm,Cov_fm,mse_fm] = ...
28.     nlinfit(x_fm, y_fm, modelFun_fm, alpha_th,...
29.     'Weights',W_linear_relative,'options', optionsLSQ);
30.
31. % Confirm with Algorithm fminbnd
32. [alpha_fminbnd,FVAL] =
33.     fminbnd(modelFun_fm,0,(alpha_nlinfit+1),optionsLSQ);
34.
35. %Extract the Jacobian Matrix
36. Jacobian_fm = full(Jacob_fm);
37.
38.
```

```

39. % Computing the confidence interval of the Alpha parameter
40. % based on Jacobian
41. CI_fm = nlparci(alpha_nlinfit,Residual_fm,'jacobian',Jacobian_fm);
42. alpha_low = CI_fm(1);
43. alpha_high = CI_fm(2);
44.
45. %Autre fonction à utiliser....
46. xdata = 0:0.01:15;
47. xdata = transpose(xdata);
48.
49. fit_maxwell_lin = (sqrt(2/pi)) * (xdata.^2/alpha_nlinfit.^3) .*
50. exp(-(xdata.^2)/(2*(alpha_nlinfit.^2))); % Density
51.
52. fit_maxwell_lin_low = (sqrt(2/pi)) * (xdata.^2/alpha_low.^3) .*
53. exp(-(xdata.^2)/(2*(alpha_low.^2))); % Density low
54.
55. fit_maxwell_lin_high = (sqrt(2/pi)) * (xdata.^2/alpha_high.^3) .*
56. exp(-(xdata.^2)/(2*(alpha_high.^2))); % Density high
57.
58.
59.
60. %% Fitting Evaluation
61. numerator = sum((countMatrix_DGD(:,4) -
62. mean(countMatrix_DGD(:,4))).*(fit_maxwell_lin(:,1) -
63. mean(fit_maxwell_lin(:,1))));
64. denominator = sqrt(sum((countMatrix_DGD(:,4) -
65. mean(countMatrix_DGD(:,4)).^2)).*sqrt(sum((fit_maxwell_lin(:,1) -
66. mean(fit_maxwell_lin(:,1)).^2)));
67. Pearson_maxwell = (numerator/denominator);
68.
69. Pearson_maxwell_2 = Pearson_maxwell.^2;
70. Pearson_maxwell = sqrt(Pearson_maxwell_2);
71.
72. evalMaxFit = corrcoef(countMatrix_DGD(:,4), fit_maxwell_lin(:,1));
73. PearsonMaxFit = evalMaxFit(1,1);
74.
75.
76.
77. %% Displaying Fitting curves + PDF DGD
78.
79. figure(1)
80.
81. k0 =
82. bar(countMatrix_DGD(1:1201,1),countMatrix_DGD(1:1201,4),'b','BaseVal
83. ue',min(countMatrix_DGD_pos(:,4))/1.2);
84. %set(gca,'YDir','reverse')
85. hold on;
86.
87. k1 = plot(xdata(:,1), fit_maxwell_lin(:,1));
88. set(k1,'color','r');
89. set(k1,'linewidth',2);
90. set(k1,'marker','none');
91. hold on
92.
93.

```

```
94. k2 = plot(xdata(:,1), fit_maxwell_lin_low(:,1), '--');
95. set(k2, 'color', 'r');
96. set(k2, 'linewidth', 2);
97. set(k2, 'marker', 'none');
98. hold on
99.
100. k3 = plot(xdata(:,1), fit_maxwell_lin_high(:,1), '--');
101. set(k3, 'color', 'r');
102. set(k3, 'linewidth', 2);
103. set(k3, 'marker', 'none');
104. hold on
105. ylim([min(countMatrix_DGD_pos(:,4))/1.2
106. max(countMatrix_DGD_pos(:,4))*1.5]);
107. ylim([1e-5 1e0]);
108. xlim([-0.01 12]);
109. xlabel('DGD [ps]', 'fontsize', 20, 'fontweight', 'b');
110. ylabel('PDF ', 'fontsize', 20, 'fontweight', 'b');
111. set(gca, 'YTick', [1e-5 1e-4 1e-3 1e-2 1e-1 1e0]);
112. set(gca, 'XTick', 0:1:12);
113. set(gca, 'Position', [0.0609375 0.0893939393939394 0.92890625
114. 0.886363636363636]);
115. set(gca, 'FontSize', 20, 'FontName', 'Calibri');
116. set(gca, 'YMinorTick', 'on');
117. set(gca, 'YScale', 'log');
118. grid on;
119. figure(gcf)
120. hold off;
```


ANNEXE XXVI

MÉTHODE DE TRAPÈZE POUR LE CALCUL DE L'AIRE D'UN HISTOGRAMME

Kreyszig *et al.* (2011) propose la méthode des trapèzes afin que l'intégrale des valeurs de probabilité soit égale à 1. Cette méthode porte sur le calcul d'intégration par partie de la courbe des probabilités p_x d'une variable aléatoire x . L'équation (A XXVI-1) formule le calcul de la méthode des trapèzes appliqués sur les valeurs de probabilités de la variable x .

$$\begin{aligned} \text{Aire} &= \int_a^b p(x)dx && \text{(A XXVI-1)} \\ &\approx h \left[\frac{1}{2}p(a) + p(x_1) + p(x_2) + \dots + p(x_{n-1}) + \frac{1}{2}p(b) \right] \\ &\text{Adaptée de Kreyszig } et al. \text{ (2011)} \end{aligned}$$

h définit le facteur de l'aire de la courbe des probabilités et est calculé par l'équation (A XXVI-2).

$$h = \left(\frac{b - a}{N_c} \right) \quad \text{(A XXVI-2)}$$

Tirée de Kreyszig *et al.* (2011)

N_c définit le nombre de classes calculé dans la matrice d'occurrence. La figure-A XXVI-1 décrit la méthode des trapèzes effectuant le calcul de l'intégration par partie des probabilités $p(x_i)$.

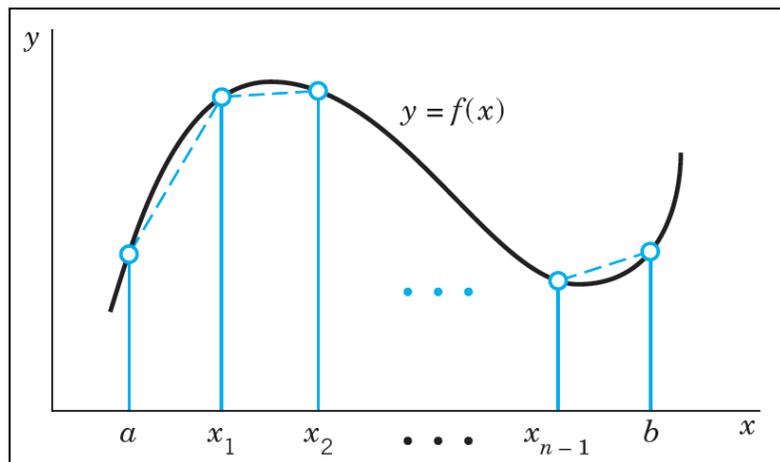


Figure-A XXVI-1 Représentation de la méthode des trapèzes
Tirée de Kreyszig *et al.* (2011)

La figure 4.2 indique que l'intégration par partie s'effectue sur une région délimitée entre la valeur a et la valeur b . Les valeurs a et b sont égales à la première classe et à la dernière classe respectivement afin que l'aire soit calculée sur toute la courbe des probabilités de la variable x . L'équation (A XXVI-3) formule le calcul réduit de la méthode des trapèzes formulée par l'équation (A XXVI-2).

$$\text{Aire} = \frac{h}{2}(p(a) + p(b)) + h \cdot \sum_{i=1}^{n-1} p(x_i) \quad (\text{A XXVI-3})$$

Tirée de Kreyszig *et al.* (2011)

ANNEXE XXVII

PRÉSENTATION DE L'ALGORITHME DE LEVENBERG-MARQUARDT

Le paramètre optimisé α_{opt} est obtenu par la minimisation la fonction de la méthode pondérée des moindres carrés. L'algorithme Levenberg-Marquardt est appliqué sur la fonction $S_w(\alpha)$. Cet algorithme apporte une solution numérique au problème de minimisation de la méthode des moindres carrés (Levenberg, 1963). Gavin (2013) énonce que l'algorithme de Levenberg-Marquardt est une combinaison des algorithmes de Gauss-Newton et de l'algorithme du Gradient. Cet algorithme tire parti de la performance des deux précédents algorithmes. Afin que le résultat de cet algorithme puisse être satisfaisant, le paramètre α est approché théoriquement.

L'algorithme de Levenberg-Marquardt effectue plusieurs itérations de la distribution maxwellienne en calculant la dérivée partielle du paramètre θ à chaque variation du paramètre α . Marquardt (1963) propose la fonction de linéarisation $f(x_i, \alpha + \theta)$ formulée par l'équation (4.17) en vue d'approximer le paramètre d'optimisation θ .

$$f(x_i, \alpha + \theta) = f(x_i, \alpha) + J_i \times \theta \quad (\text{A XXVII-1})$$

J_i représente le gradient de la fonction de distribution par rapport au paramètre d'optimisation θ sur une région de valeurs. L'équation (A XXVII-2) présente le calcul de la dérivée partielle de la fonction de densité, appelée $f(x_i, \alpha)$, par rapport au paramètre d'optimisation θ .

$$J_i = \frac{\partial f(x_i, \alpha)}{\partial \alpha} \quad (\text{A XXVII-2})$$

Adaptée de Levenberg (1963)

Gavin (2013) formule l'application de l'algorithme de Levenberg-Marquardt sur la méthode des moindres carrés en y intégrant les valeurs de poids w_i . Gavin (2013) formule la fonction $S_w(\alpha + \theta)$ présentant la méthode des moindres carrés pondérée et adaptée par l'algorithme *Levenberg-Marquardt*. Cette fonction est composée des poids w_i précédemment calculés, des valeurs de la distribution statistique des valeurs mesurées y_i , de la fonction de densité théorique f , de son gradient rapport au paramètre α ainsi que du paramètre d'optimisation θ .

$$S_w(\alpha + \theta) = \sum_{i=1}^N w_i (y_i - f(x_i, \alpha) - J_i \times \theta)^2 \quad (\text{A XXVII-3})$$

Adaptée de Gavin (2013)

Gavin (2013) propose la représentation matricielle de la fonction $S_w(\alpha + \theta)$ selon l'équation (4.20).

$$S_w(\alpha + \theta) = y_{obs}^T w y_{obs} + f(\alpha)^T w f(\alpha) - 2y_{obs}^T w f(\alpha) - 2(y_{obs} - f(\alpha))^T w J \theta + \theta^T J^T w J \theta \quad (\text{A XXVII-4})$$

Adaptée de Gavin (2013)

y_{obs} définit la distribution statistique des valeurs mesurées. $f(\alpha)$ représente la fonction de densité de probabilité théorique paramétrée par la valeur α . w spécifie la matrice des poids calculés par l'équation (A XXVII-4). J définit le gradient de la fonction de densité de probabilité théorique par rapport à α , et enfin, θ est le paramètre d'optimisation de l'algorithme *Levenberg-Marquardt*.

La valeur de la variable $S_w(\alpha + \theta)$ est minimale lorsque la dérivé partielle de $S_w(\alpha + \theta)$ en θ est nulle d'après (Levenberg, 1944). L'équation (A XXVII-5) formule le calcul de la dérivée partielle de la dernière itération de l'algorithme selon Levenberg (1963) :

$$\frac{\partial S_w(\alpha + \theta)}{\partial \theta} = 0 \quad (\text{A XXVII-5})$$

Gavin (2013) effectue la dérivée partielle $S_w(\alpha + \theta)$ en fonction du paramètre d'optimisation θ . La dérivée partielle de la fonction $\partial S_w(\alpha + \theta)$ est formulée par l'équation (4.22).

$$\frac{\partial S_w(\alpha + \theta)}{\partial \theta} = -2(y_{obs} - f(\alpha))^T w J + 2\theta^T J^T w J \quad (\text{A XXVII-6})$$

Adaptée de Gavin (2013)

Gavin (2013) déduit ainsi la formule lorsque la fonction $S_w(\alpha + \theta)$ est minimale, c'est-à-dire que la dérivée partielle de $\frac{\partial S_w(\alpha + \theta)}{\partial \theta}$ est nulle.

$$[J^T w J] \theta = J^T w (y_{obs} - f(\alpha)) \quad (\text{A XXVII-7})$$

Adaptée de Gavin (2013)

L'algorithme optimise cette expression en intégrant à chaque itération un facteur d'amortissement positif symbolisé par λ . Gavin (2013) présente l'optimisation de l'équation (4.23) afin que la solution puisse être la meilleure possible tout en réduisant le nombre maximal d'erreur. L'optimisation de l'algorithme Levenberg-Marquardt, appliquée dans cette étude, est formulée par l'équation (A XXVII-8).

$$[J^T w J + \lambda \text{diag}(J^T w J)]\theta = J^T w (y_{obs} - f(\alpha)) \quad (\text{A XXVII-8})$$

Adaptée de Gavin (2013)

Le facteur d'amortissement λ est modifié au cours des itérations dans le but de trouver la meilleure solution. Le paramètre optimisé α_{opt} est obtenue en additionnant la valeur du paramètre α initiale à la valeur θ finale.

La fonction Matlab *nlinfit* effectue plusieurs itérations de la méthode pondérée des moindres carrés $S_w(\alpha)$ en appliquant l'algorithme Levenberg-Marquardt. Cette fonction est configurée par des options spécifiques de la fonction *statset* en vue d'optimiser la solution du paramètre α_{opt} de la fonction de densité de probabilité théorique $f(x_i, \alpha)$.

ANNEXE XXVIII

APPROXIMATION DE L'ERREUR STANDARD DU PARAMÈTRE D'UNE FONCTION DE DENSITÉ DE PROBABILITÉ

L'approximation de l'erreur standard du paramètre optimisé α_{opt} a été calculée en vue de l'obtention du paramètre inférieure α_{low} et du paramètre supérieure α_{upp} à un intervalle de confiance de 95 %. Gavin (2013) énonce que le calcul de l'erreur standard nécessite la matrice de covariance du paramètre α_{opt} et l'erreur quadratique moyenne MSE . La matrice de covariance est calculée en utilisant la matrice jacobienne J_{opt} du paramètre optimisé α_{opt} obtenu par l'application de l'algorithme Levenberg-Marquardt (*Voir* Annexe XXVII, p.180). La matrice jacobienne J_{opt} est le résultat dernière itération de l'algorithme Levenberg-Marquardt sur la méthode pondérée des moindres carrés. L'équation (XXVIII.1) formule le calcul de matrice jacobienne J_{opt} définissant la dérivée partielle de la fonction de densité de probabilité théorique doté du paramètre optimisé α_{opt} .

$$J_{opt} = \frac{\partial f}{\partial \alpha}(\alpha_{opt}, x) \quad (\text{A XXVIII-1})$$

Adapté de Gavin (2013)

Avant de procéder aux calculs de la matrice de covariance, la variance des valeurs résiduelles entre la fonction $f(\alpha_{opt}, x)$ et des valeurs de densités observées y_{obs} a été calculée par l'équation (XXVIII.2).

$$MSE = \frac{SSR}{n - p} = \frac{\sum_{i=1}^n (f(x_i, \alpha_{opt}) - y_{obs})^2}{n - p} \quad (\text{A XXVIII-2})$$

Adaptée de Walpole *et al.* (2007)

SSR représente la somme des valeurs résiduelles entre la fonction de distribution de Maxwell $f_M(x_i, \alpha_{opt})$ et les valeurs y_{obs} de la fonction de densité de probabilité des valeurs mesurées. n définit le nombre total de classes calculées et p définit le nombre de paramètres variables dans cette étude. L'erreur quadratique moyenne MSE définit donc la valeur ajustée aux degrés de libertés, c'est-à-dire du nombre total de paramètres optimisés p (Walpole *et al.*, 2007). Wolberg (2005) et Björck (1996) proposent le calcul de la matrice de covariance du paramètre d'ajustement α_{opt} par la multiplication de la matrice jacobienne J_{opt} et de l'erreur quadratique moyenne MSE . L'équation (XXVIII.3) formule le calcul de la matrice de covariance du paramètre optimisé de la fonction de densité de probabilité théorique.

$$Cov(\alpha) = MSE \times (J_{opt}^T \times J_{opt})^{-1} \quad (\text{A XXVIII-3})$$

Adapté de Wolberg (2005)

La matrice de covariance des paramètres optimisés est définie par la représentation matricielle suivante :

$$Cov(\alpha_i, \alpha_j) = \begin{bmatrix} \sigma_{\alpha_i}^2 & \cdots & \sigma_{\alpha_{ij}} \\ \vdots & \ddots & \vdots \\ \sigma_{\alpha_{ji}} & \cdots & \sigma_{\alpha_j}^2 \end{bmatrix} \quad (\text{A XXVIII-4})$$

Seules les variances de chaque paramètre $\sigma_{\alpha_i}^2 \dots \sigma_{\alpha_j}^2$ sont nécessaires afin que l'erreur standard du paramètre optimisé de la fonction de densité de probabilité théorique puisse être approximée. Ces variances sont situées sur la diagonale de la matrice de covariance $Cov(\alpha_i, \alpha_j)$. La matrice de covariance $Cov(\alpha_i, \alpha_j)$ est une matrice symétrique composée de valeurs réelles. Moler (2004) propose la factorisation QR de la matrice jacobienne J_{opt} en vue de l'extraction de la partie triangulaire supérieure de la matrice de covariance $Cov(\alpha_i, \alpha_j)$. La variance $\sigma_{\alpha_{opt}}^2$ est extraite de la partie triangulaire supérieure de la matrice de covariance $Cov(\alpha_i, \alpha_j)$ du paramètre optimisé α_{opt} . L'équation (XXVIII.5) formule le calcul de la factorisation QR de la matrice jacobienne J_{opt} en vue de l'obtention de la matrice orthogonale Q_{opt} et de la matrice triangulaire R_{opt} .

$$J_{opt} = Q_{opt} \times R_{opt} \quad (\text{A XXVIII-5})$$

Adaptée de Moler (2004)

Wolberg (2005) démontrent que la variance du paramètre optimisé α_{opt} peut être calculé à partir seulement de la partie triangulaire supérieure R_{opt} . La variance du paramètre optimisé $\sigma_{\alpha_{opt}}^2$ est calculée par l'équation (XVIII.6) avec la matrice triangulaire R_{opt} .

$$\sigma_{\alpha_{opt}}^2 = (R_{opt}^T \times R_{opt})^{-1} \quad (\text{A XXVIII-6})$$

Adaptée de Wolberg (2005)

L'approximation de l'erreur standard du paramètre α_{opt} de la fonction de densité de probabilité théorique est associée à l'erreur quadratique moyenne. Smyth (2002) et Wolberg (2005) formulent le calcul de l'approximation de l'erreur standard $SE_{\alpha_{opt}}$ du paramètre α_{opt} par la multiplication avec l'erreur quadratique moyenne MSE selon l'équation (XXVIII.7).

$$SE_{\alpha_{opt}} = \sqrt{\sigma_{\alpha_{opt}}^2 \times MSE} \quad (\text{A XXVIII-7})$$

Adaptée de Smyth (2002) : «Nonlinear Regression»,
Volume 3, (p. 1405–1411)

Les paramètres supérieurs et inférieurs du paramètre α_{opt} à un intervalle de confiance défini sont calculés par la table de la loi de *Student* (Voir ANNEXE XXIX, p. 186).

ANNEXE XXIX

TABLE DE LA LOI DE STUDENT

La figure XXXIII-1 présente la table de *Student* pour estimer les intervalles de confiance des paramètres des fonctions de densité de probabilité.

$F(z)$	Number of Degrees of Freedom									
	1	2	3	4	5	6	7	8	9	10
0.5	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.6	0.32	0.29	0.28	0.27	0.27	0.26	0.26	0.26	0.26	0.26
0.7	0.73	0.62	0.58	0.57	0.56	0.55	0.55	0.55	0.54	0.54
0.8	1.38	1.06	0.98	0.94	0.92	0.91	0.90	0.89	0.88	0.88
0.9	3.08	1.89	1.64	1.53	1.48	1.44	1.41	1.40	1.38	1.37
0.95	6.31	2.92	2.35	2.13	2.02	1.94	1.89	1.86	1.83	1.81
0.975	12.7	4.30	3.18	2.78	2.57	2.45	2.36	2.31	2.26	2.23
0.99	31.8	6.96	4.54	3.75	3.36	3.14	3.00	2.90	2.82	2.76
0.995	63.7	9.92	5.84	4.60	4.03	3.71	3.50	3.36	3.25	3.17
0.999	318.3	22.3	10.2	7.17	5.89	5.21	4.79	4.50	4.30	4.14

$F(z)$	Number of Degrees of Freedom									
	11	12	13	14	15	16	17	18	19	20
0.5	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.6	0.26	0.26	0.26	0.26	0.26	0.26	0.26	0.26	0.26	0.26
0.7	0.54	0.54	0.54	0.54	0.54	0.54	0.53	0.53	0.53	0.53
0.8	0.88	0.87	0.87	0.87	0.87	0.86	0.86	0.86	0.86	0.86
0.9	1.36	1.36	1.35	1.35	1.34	1.34	1.33	1.33	1.33	1.33
0.95	1.80	1.78	1.77	1.76	1.75	1.75	1.74	1.73	1.73	1.72
0.975	2.20	2.18	2.16	2.14	2.13	2.12	2.11	2.10	2.09	2.09
0.99	2.72	2.68	2.65	2.62	2.60	2.58	2.57	2.55	2.54	2.53
0.995	3.11	3.05	3.01	2.98	2.95	2.92	2.90	2.88	2.86	2.85
0.999	4.02	3.93	3.85	3.79	3.73	3.69	3.65	3.61	3.58	3.55

$F(z)$	Number of Degrees of Freedom									
	22	24	26	28	30	40	50	100	200	∞
0.5	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
0.6	0.26	0.26	0.26	0.26	0.26	0.26	0.25	0.25	0.25	0.25
0.7	0.53	0.53	0.53	0.53	0.53	0.53	0.53	0.53	0.53	0.52
0.8	0.86	0.86	0.86	0.85	0.85	0.85	0.85	0.85	0.84	0.84
0.9	1.32	1.32	1.31	1.31	1.31	1.30	1.30	1.29	1.29	1.28
0.95	1.72	1.71	1.71	1.70	1.70	1.68	1.68	1.66	1.65	1.65
0.975	2.07	2.06	2.06	2.05	2.04	2.02	2.01	1.98	1.97	1.96
0.99	2.51	2.49	2.48	2.47	2.46	2.42	2.40	2.36	2.35	2.33
0.995	2.82	2.80	2.78	2.76	2.75	2.70	2.68	2.63	2.60	2.58
0.999	3.50	3.47	3.43	3.41	3.39	3.31	3.26	3.17	3.13	3.09

Figure-A XXIX-1 Table de la loi de Student
Tirée de Kreyszig *et al.* (2011, Table A9, p.A103)

ANNEXE XXX

OUTIL DE TRAITEMENT DES DONNÉES ENREGISTRÉES AVEC LE POLARIMÈTRE

Le script-A XXX-1 est l'outil de traitement de données développé dans l'environnement Matlab pour la création d'une structure de données qui sauvegarde les vitesses de rotation angulaire des états de polarisation en fonction du temps.

Script-A XXX-1 Développement d'une structure de données indexée des vitesses de rotation angulaire des états de polarisation mesurés avec le polarimètre

```
1. close all
2. clear all
3. clc
4.
5. dossierdesmat = 'D:\Projet - Liaison Sacramento - Portland\Raw
6. Files\11-05 to 11-11 Recordings_Polarimeter\';
7.
8. struct=what(dossierdesmat);           % Files repertory in a structure
9. taille=length(struct.mat);
10. taille=taille(1,1);
11. nFiles = taille;
12. info_glitch = [];
13. glitch_tab = [];
14. mean_transient = [];
15. std_transient = [];
16. TS= [];
17. Struct_ARR = struct();
18.
19. for fileIx = 1:taille
20.
21.     % File loading method
22.     nomfichier=struct.mat(fileIx);
23.     nomfichier=nomfichier{1:1}; %récuperation du nom de fichier dans
24.     la structure s, ya des cell et autres conneries
25.     chemincomplet=strcat(dossierdesmat,nomfichier);
26.     %concatenation du chemin du dossier (C:\fljkl...\) avec le nom du
27.     fichier courant k
28.
29.
30.     try
31.         FileData = load(chemincomplet);
32.
33.     catch
34.         fprintf('Could not load: %s\n', nomfichier);
35.         continue;
36.     end
37.     S = FileData.sop.';
38.     P = FileData.p;
```

```

39.  %--
40.  % Note that S is a 3-by-nSample matrix of measured Stokes vectors
41.  %--
42.  dt = 0.05; % Sampling time. Units of [msec]
43.  S = bsxfun(@rdivide, S, sqrt(sum(S.^2))); % Normalize the measured
44.  Stokes vectors to unit length
45.
46.  dot_product = sum(S(:,1:end-1) .* S(:,2:end)); % Size 1-by-
47.  (nSamples-1)
48.  dot_product(dot_product < -1) = -1; % Avoids numerical artifacts
49.  dot_product(dot_product > +1) = +1;
50.
51.  angular_velocity = acos(dot_product)./dt; % Units of [rad/msec]
52.  angular_velocity=transpose(angular_velocity);
53.
54.  %% Timestamp creation
55.      if isfield(FileData,'started_at')
56.          for i = 1:length(angular_velocity)
57.              TS(end+1)=addtodate(datenum(FileData.started_at), ...
58.              round(0.05*i), 'millisecond');
59.          end
60.
61.
62.      else
63.          remain = nomfichier(1:length(nomfichier-4));
64.          TS_init=zeros(1,7);
65.          [TS_remain, remain] = strtok(remain, '.');%
66.  String TS sans ".mat"
67.
68.          for k = 1:7
69.              % parse le nom de fichier suivant "_"
70.              [token, TS_remain] = strtok(TS_remain, '_');
71.              TS_init(1,k)=str2double(token);
72.          end
73.
74.          TS_start = datenum(TS_init(1,2),TS_init(1,3),...
75.
76.          TS_init(1,4),TS_init(1,5),TS_init(1,6),TS_init(1,7));
77.
78.          for i = 1:length(angular_velocity)
79.              TS(end+1)=addtodate(datenum(TS_start),round(0.05*i), 'millisecond');
80.
81.          end
82.      end
83.
84.  TS=transpose(TS);
85.  TS_ARR = [angular_velocity(:,1),TS(:,1)];
86.
87.
88.  %% Polarimeter glitch detection
89.  [MaxVal loc_glitch] = max(angular_velocity(:,1));
90.  fin_glitch = loc_glitch+50; % Initialisation are the init point of
91.  each polarimeter file
92.
93.  glitch_tab = [transpose(loc_glitch:fin_glitch),

```

```
94. angular_velocity(loc_glitch:fin_glitch)];
95. init_position = find(glitch_tab(:,2)<0.5);
96. init = loc_glitch + min(init_position);
97.
98. %% Glitch tab
99. info_glitch(end+1,1:3) = [fileIx,loc_glitch,MaxVal];
100.
101. %% Suppression des valeurs de l'effet transitoire (glitch)
102. TS_ARR_cld = removerows(TS_ARR,1:(init-1));
103.
104. %% Record the mean of rotation rate for each file
105. mean_transient(fileIx) =
106. mean(angular_velocity(init:length(angular_velocity(:,1)),1));
107. %% Record the standard deviation of rotation rate for each file
108. std_transient(fileIx) =
109. std(angular_velocity(init:length(angular_velocity(:,1)),1));
110.
111.
112. %% Indexed data structure of cleaned polarization rotation rates
113.
114. fldn = ['id',num2str(fileIx)];
115. %         peak_intervals = peak_intervals.*dt_norm;
116. started_at = TS_ARR(1,2);
117. Struct_ARR(1).(fldn) = started_at;
118. Struct_ARR(2).(fldn) = TS_ARR_cld;
119. TS=[];
120. angular_velocity = [];
121.
122. end
123.
```


ANNEXE XXXI

ÉTAPE 1 : CRÉATION D'UNE STRUCTURE DE DONNÉES D'ADRESSE DE FICHIERS MATLAB

La figure XXXI-1 présente la structure de données *StructData* créée à partir du dossier des fichiers de mesure obtenus par le polarimètre.

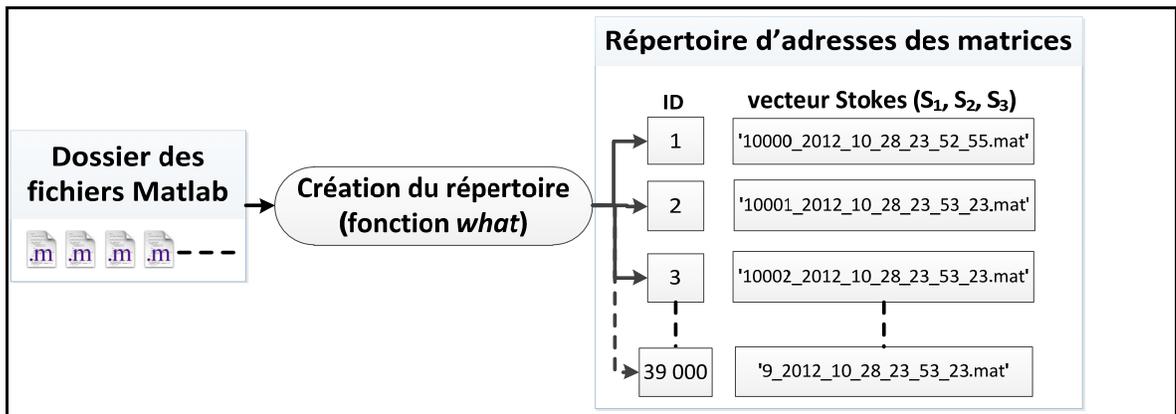


Figure-A XXXI-1 Création de la structure de fichiers Matlab

La fonction *what* de Matlab a été appliquée sur le dossier des fichiers de mesure pour la création de la structure de données *StructData*. La structure de donnée *StructData* améliore la vitesse d'accès aux données constituées de 200 000 vecteurs Stokes pour chaque fichier de mesure enregistré au polarimètre. La figure 4.2 décrit la structure de données obtenue après l'application de la fonction *what* sur le dossier des fichiers Matlab.

struct_Polarimeter_data.mat <15000x1 cell>	
	1
1	10000_2012_11_9_8_38_46.mat
2	10001_2012_11_9_8_39_19.mat
3	10002_2012_11_9_8_39_51.mat
4	10003_2012_11_9_8_40_24.mat
5	10004_2012_11_9_8_40_56.mat
6	10005_2012_11_9_8_41_28.mat
7	10006_2012_11_9_8_42_1.mat
8	10007_2012_11_9_8_42_33.mat
9	10008_2012_11_9_8_43_6.mat
10	10009_2012_11_9_8_43_39.mat

Figure-A XXXI-2 Structure de données obtenue après l'application de la première étape de traitement

ANNEXE XXXII

ÉTAPE 2 : EXTRACTION ET NORMALISATION DES VECTEURS STOKES

La fonction *bsxfun* de Matlab a été appliquée sur la matrice des vecteurs Stokes (S_1, S_2, S_3) afin que le calcul de normalisation puisse être effectué. Cette fonction a été sélectionnée car elle accélère la vitesse de calcul entre différentes matrices. La fonction *rdivide* (*right divide*) définit le paramètre de la fonction *bsxfun*. La figure-A XXXII-1 présente la procédure d'extraction et le calcul de normalisation des paramètres Stokes (S_1, S_2, S_3).

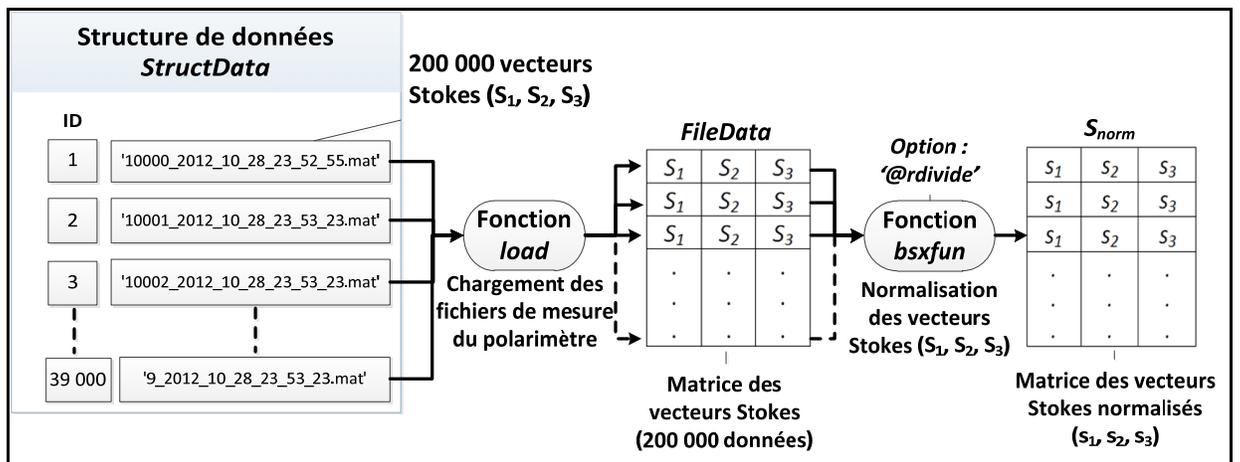


Figure-A XXXII-1 Extraction et normalisation des matrices de vecteurs Stokes

La figure-A XXXII-1 indique les différentes matrices obtenues après que les fonctions *load* et *bsxfun* ont été exécutées (Voir ANNEXE XXX, p. 186). La matrice *FileData*, constituée de 200 000 vecteurs Stokes (S_1, S_2, S_3) est chargée par l'application de la fonction *load* (Voir Figure-A XXXII-1). Les vecteurs Stokes normalisés (s_1, s_2, s_3) ont par la suite été enregistrés dans la matrice S_{norm} après l'utilisation de la fonction *bsxfun* (Voir Figure-A XXXII-1).

ANNEXE XXXIII

ÉTAPE 3 : CALCUL DES VITESSES DE ROTATION ANGULAIRE DES ÉTATS DE POLARISATION

Le produit scalaire de chaque vecteur Stokes a été calculé par l'application de la fonction *times* sur les vecteurs Stokes adjacents. Les résultats obtenus par l'application de cette fonction se présentent sous la forme de trois matrices distinctes. Chaque matrice correspond au produit scalaire appliqué sur une composante spécifique du vecteur Stokes (p. ex. $s_1(t_1) \cdot s_1(t_2)$) soit S_1 pour *Dot_product_S1*, S_2 pour *Dot_product_S2* et S_3 pour *Dot_product_S3*. La somme des produits scalaires a été ensuite calculée par l'application de la fonction *sum* et a été enregistrée dans une matrice de données intitulée *Sum_dot_product* (Voir Figure-A XXXIII-1).

Les sommes des produits scalaires ($s_1(t_1) \cdot s_1(t_2) + s_2(t_1) \cdot s_2(t_2) + s_3(t_1) \cdot s_3(t_2)$) supérieures à '+1' ou inférieures à '-1' sont égales respectivement à '+1' et à '-1' par l'application de la fonction *find* afin que des artefacts numériques puissent être évités. Les valeurs d'angle, en rad, ont été obtenues par l'application de la fonction *acos* qui calcule le cosinus inverse de la somme des produits scalaires précédemment obtenue. Les valeurs d'angle obtenues ont par la suite été divisées par la période d'échantillonnage *dt* pour l'obtention des vitesses de rotation angulaire exprimées en rad/ms. *dt* définit la soustraction des temps de mesure adjacents t_1 et t_2 et est égale à 0.05 ms. La fonction *rdivide* divise les valeurs d'angle par *dt*. Les vitesses de rotation angulaire ont été enregistrées dans la matrice *ARR* (Voir Figure-A XXXIII-1).

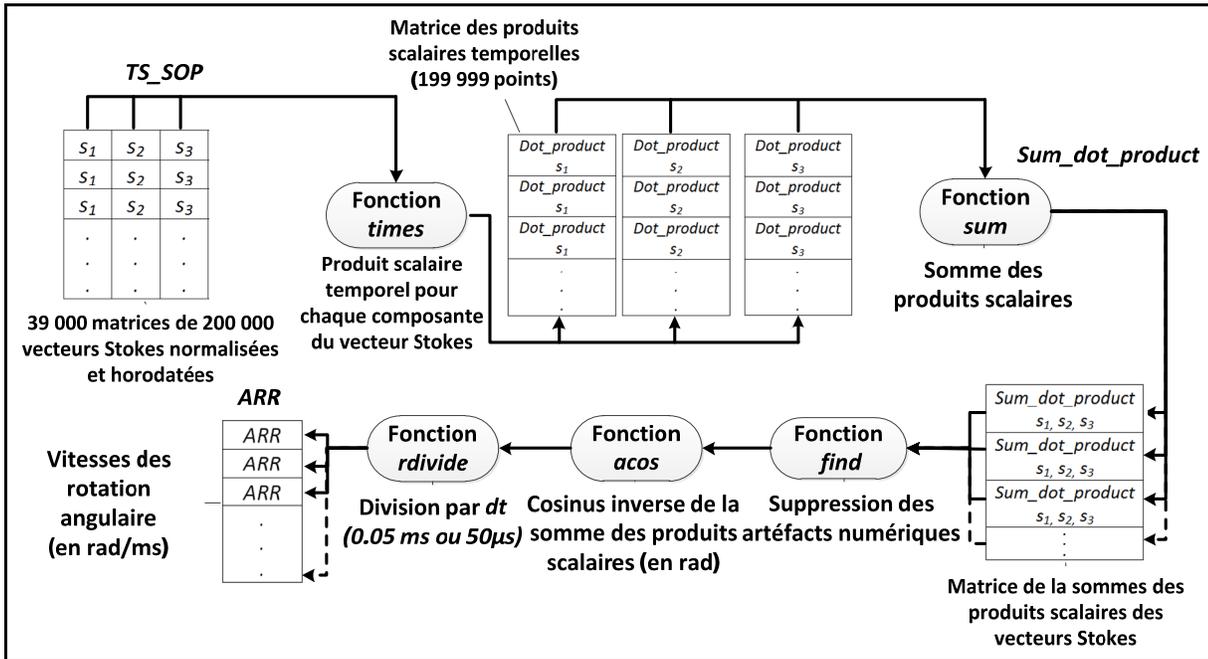


Figure-A XXXIII-1 Méthode de calcul des vitesses de rotation angulaire dans Matlab

La méthode de calcul présentée par la figure-A XXXIII-1 formalise le code Matlab développé en vue de l'automatisation du calcul de 7.8 milliards valeurs de vitesse de rotation angulaire sur une période de 14 jours de mesure effectuée avec le polarimètre (Voir ANNEXE XXX, p. 186).

ANNEXE XXXIV

ÉTAPE 4: SUPPRESSION DES VALEURS TRANSITOIRES DES VITESSES DE ROTATION ANGULAIRE DES ÉTATS DE POLARISATION

La figure-A XXXIV-1 présente la procédure de traitement appliquée sur les matrices *ARR* en vue de la suppression des valeurs transitoires de vitesse de rotation angulaire des états de polarisation causées par la période de réinitialisation du polarimètre.

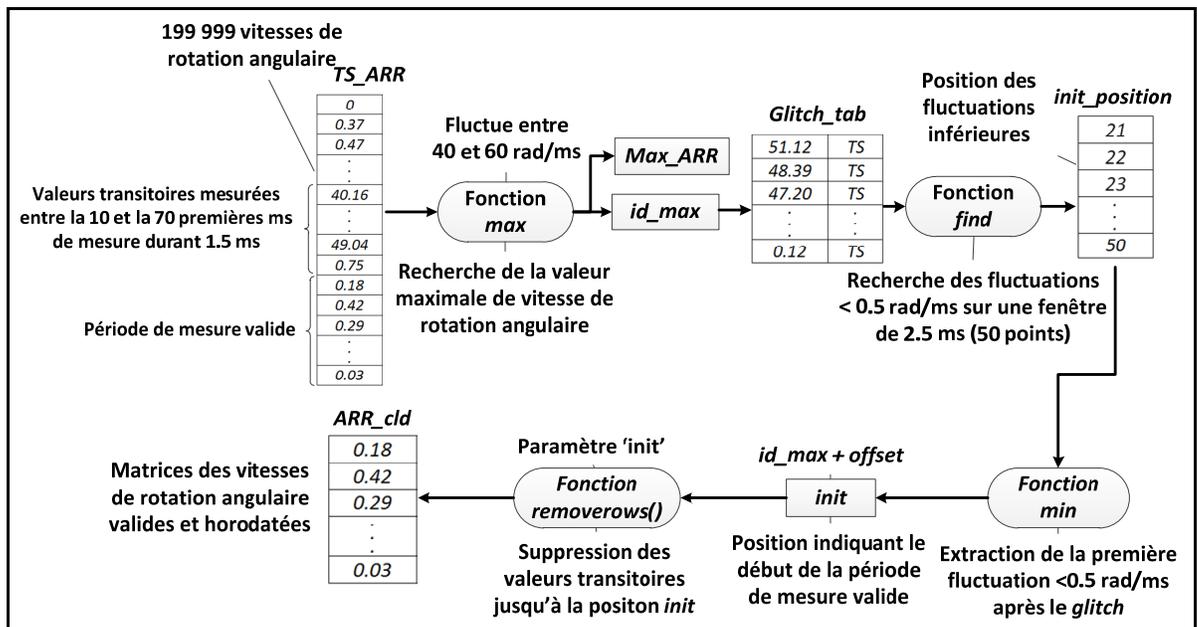


Figure-A XXXIV-1 Procédure de traitement pour la suppression des valeurs transitoires de vitesse de rotation angulaire des états de polarisation mesurés avec le polarimètre

La figure-A XXXIV-1 formalise la fonction Matlab intégrée au script de traitement de données des mesures du polarimètre (Voir ANNEXE XXX, p. 186). La position de la valeur maximale des fluctuations des états de polarisation a tout d'abord été identifiée par l'application de la fonction *max* sur la matrice *ARR*. En effet, la valeur maximale de chaque matrice *TS_ARR* définit une valeur transitoire et fluctue entre 40 et 60 rad/ms (Voir Figure-A XXXIV-1). Le résultat de la fonction *max* se présente sous la forme de la valeur maximale *Max_ARR* et de sa position dans la matrice *ARR*, appelée *id_max*. La durée de la mesure des valeurs transitoires ne dépasse jamais 1.5 ms soit 30 points de mesure. Pour cette raison, une fenêtre de 50 points a été sélectionnée à partir de la position *id_max* et a été enregistrée dans la matrice *glitch_tab* (Voir Figure-A XXXIV-1). La fonction *find* a par la suite été exécutée sur cette matrice en vue de la sélection des positions qui désignent les fluctuations inférieures à 0.5 rad/ms enregistrées dans la matrice *init_position*. La première valeur de la matrice *init_position* indique la première fluctuation inférieure à 0.5 rad/ms après le *glitch* (Voir

Figure-A XXXIV-1). Cette position particulière a été appelée *offset* et a été obtenue en appliquant la fonction *min* sur la matrice *init_position*. Les valeurs des positions *id_max* et *offset* ont par la suite été additionnées. La position *init* a de cette manière été obtenue et indique le début de la période de mesure valide (*Voir* Figure-A XXXIV-1). La fonction *removerows* a finalement été appliquée sur la matrice *ARR* en vue de la suppression des valeurs aberrantes jusqu'à la position *init*.

ANNEXE XXXV

ÉTAPE 5 : INSERTION DES VALEURS D'HORODATAGE AUX VITESSES DE ROTATION ANGULAIRE DES ÉTATS DE POLARISATION

La figure-A XXXV-1 décrit la procédure de traitement appliquée sur le répertoire *StructData* pour la création des valeurs d'horodatage en vue de l'obtention de matrices constituées des valeurs de vitesse rotation angulaire valides appelées *ARR_cld*.

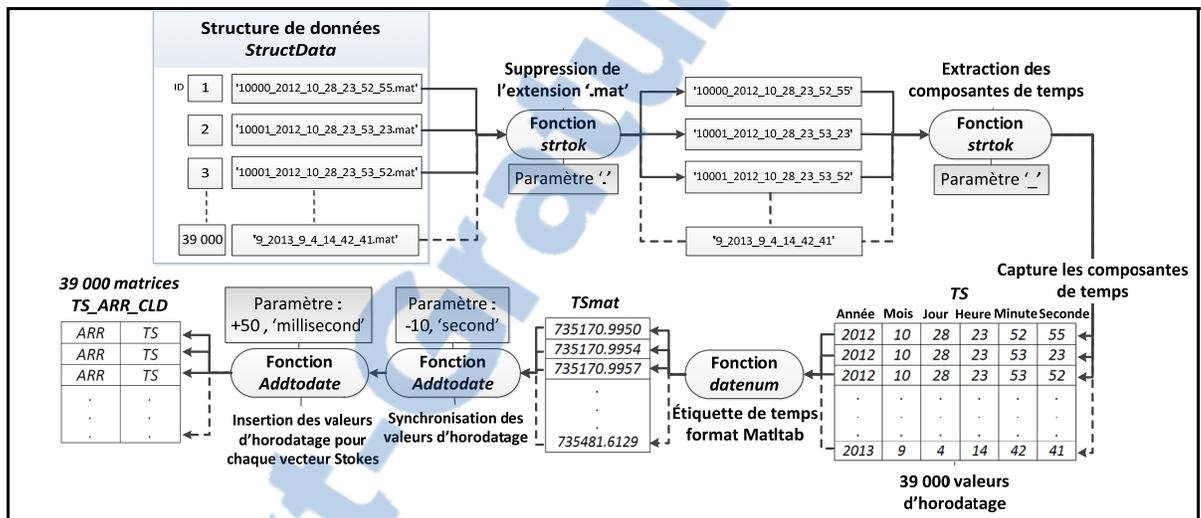


Figure-A XXXV-1 Procédure de création et d'insertion des valeurs d'horodatage dans les fichiers de mesure enregistrés par le polarimètre

Tout d'abord, la fonction *strtok* paramétrée par l'option ('.') a été appliquée sur le titre de chaque fichier de mesure en vue de la suppression de l'extension du fichier de mesure (.mat) et de l'extraction de la chaîne de caractères correspondant au temps d'enregistrement du fichier de mesure (Voir Figure-A XXXV-1). La fonction *strtok*, paramétrée dorénavant par l'option ('_'), a une seconde fois été appliquée sur chaque chaîne de caractères afin que chaque composante temporelle associée au *timestamp* puisse être extraite. La seconde utilisation de la fonction *strtok* a pour objectif de décomposer chaque valeur de la chaîne de caractères dans le but de récupérer chaque composante temporelle (P. ex. année, mois, jour, etc.). Ces composantes temporelles ont été par la suite enregistrées dans une matrice appelée *TS* (Voir Figure-A XXXV-1). 39 000 valeurs d'horodatage constituent la matrice *TS*. Chaque ligne de donnée de la matrice *TS* correspond au temps d'enregistrement inscrit sur le titre de chaque fichier de mesure Matlab.

Afin que les valeurs d'horodatage puissent être identifiées dans l'environnement Matlab, la fonction *datenum* a été appliquée sur la matrice *TS* pour la conversion des temps d'enregistrement de la matrice *TS* selon un format spécifique à Matlab. Le résultat de

l'application de la fonction *datenum* se présente sous la forme d'une matrice *TS_MAT* constituée de 39 000 valeurs d'horodatage selon le format Matlab. Une fonction a par la suite été développée en vue de la synchronisation des valeurs d'horodatage avec l'échelle des temps de mesure. En effet, la date inscrite sur chaque titre de fichier Matlab correspond au moment où un fichier de mesure est sauvegardé après 10 s de capture de données. Le fichier étant sauvegardé à la fin de la mesure, il est nécessaire de soustraire la durée de la mesure à la valeur d'horodatage précédemment créée dans le but d'obtenir les valeurs réelles auxquelles les paramètres ont été mesurés. La soustraction d'une durée temporelle de 10 secondes sur toutes les valeurs d'horodatage de la matrice *TS_MAT* a été effectuée par l'application de la fonction *addtodate* paramétrée par l'option *second* (Voir ANNEXE XXX, p. 186). Enfin, les valeurs d'horodatage associées aux vecteur Stokes mesurés par le polarimètre ont été insérées dans la matrice *ARR* par l'incrémentaire d'une période de 50 μ s pour chaque valeur d'horodatage enregistrée dans la matrice *TS_MAT* (Voir Figure-A XXXV-1). La fonction *insert* a été appliquée sur les valeurs d'horodatage de la matrice *TS* en vue de l'association des vitesses de rotation angulaire aux valeurs d'horodatage. Le résultat de cette fonction se présente sous la forme d'une matrice *TS_ARR_CLD* constituée des vitesses de rotation angulaire des états de polarisation et du temps de mesure correspondant (Voir Figure-A XXXV-1).

ANNEXE XXXVI

ÉTAPE 6 : CONCEPTION DE LA STRUCTURE DE DONNÉES INDEXÉE « PAR CLÉ » DES MESURES DU POLARIMÈTRE

La figure-A XXXVI-1 présente le processus de création de la structure de données *Struct_ARR* des vitesses de rotation angulaire des états de polarisation mesurés avec le polarimètre.

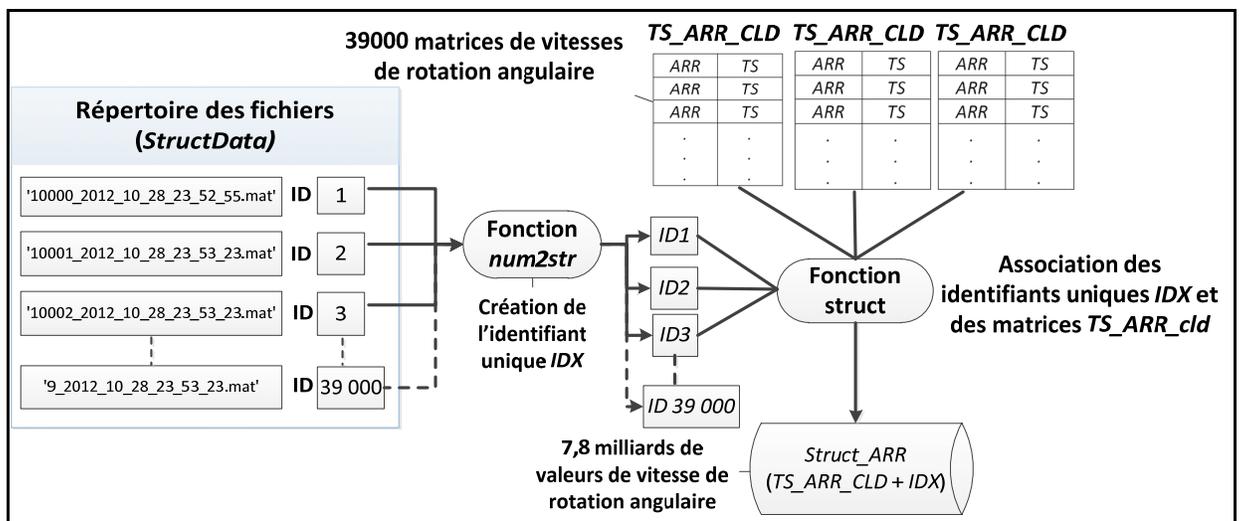


Figure-A XXXVI-1 Création de la structure de données des vitesses de rotation angulaire des états de polarisation en fonction du temps

Cette structure de données, nommée *Struct_ARR*, inscrit les entrées des fichiers de mesure correspondant aux matrices *TS_ARR_CLD* (Voir Figure-A XXXVI-1). 39 000 matrices *TS_ARR_CLD* ont été enregistrées dans cette structure de données. Cette structure de données associe un numéro d'identifiant *ID* pour chaque matrice *TS_ARR_CLD*. Pourtant, plus de 9000 numéros d'identifiant se répètent pour la totalité des fichiers sauvegardés par le polarimètre. Pour cette raison, une matrice d'identifiants uniques a été développée par l'application de la fonction *num2str* sur les entrées du répertoire *StructData* afin que chaque identifiant *IDX* soit unique. Cette fonction Matlab génère un identifiant unique pour chaque matrice *TS_ARR_cld*. L'adresse de chaque matrice *TS_ARR_cld* a été définie par la chaîne de caractère « *IDX* » où *X* désigne le numéro d'identifiant unique (Voir Figure-A XXXVI-1). La fonction *struct* a été appliquée sur chaque matrice *TS_ARR_CLD* pour la création de la structure de données associant l'identifiant unique *IDX* et la matrice *TS_ARR_CLD* correspondant (Voir ANNEXE XXX, p. 186).

ANNEXE XXXVII

CALCUL DE LA FONCTION DE DENSITÉ DES VITESSES DE ROTATION ANGULAIRE DES ÉTATS DE POLARISATION

Le script-A XXXVII-1 effectue le calcul et l'affichage de la fonction de densité de probabilité des valeurs de vitesse de rotation angulaire des états de polarisation mesurés avec le polarimètre et enregistrées dans la structure de données indexée par « clé ».

Script-A XXXVII-1 Calcul de la fonction de densité de probabilité des vitesses de rotation angulaire des états de polarisation

```
1.  %%Retrieve the size of all matrices stored in the OCLD data struct
2.  nMatrix = length(Struct_ARR);
3.
4.  maxARR = 0;
5.  bin_size = 0.01;
6.
7.  mean_pol_rot_rate = [];
8.  std_pol_rot_rate = [];
9.  SOP= [];
10.
11.  %%Creation of the density matrix of polarization rotation rate
12.  countMatrix_ARR= [0,0,0,0,0];
13.  taille_valARR = length(countMatrix_ARR(:,1));
14.
15.
16.  cellule = 1;
17.  ValARRScan = 0;
18.  bin_size = 0.01;
19.
20.  for Ix = 1:nMatrix
21.
22.      ID = ['id',num2str(Ix)];
23.      TS_ARR_CLD = Struct_ARR(1,2).(ID);
24.
25.      % Record the mean of rotation rate for each file
26.      mean_pol_rot_rate(Ix) = mean(TS_ARR_CLD(:,1));
27.
28.      % Record the standard deviation of rotation rate for each file
29.      std_pol_rot_rate(Ix) = std(TS_ARR_CLD(:,1));
30.
31.
32.      size_tabARR = length(TS_ARR_CLD(:,1));
33.      cellule = 1;
34.
35.      if(max(TS_ARR_CLD(:,2))>maxARR)
36.          max_bins=floor(max(TS_ARR_CLD(:,1))*(1/bin_size))+1;
37.          maxARR = max(TS_ARR_CLD(:,1));
38.          temp_countARR =...
```

```

39.
40. transpose(roundn((length(countMatrix_ARR(:,1))+1):max_bins)...
41.             *bin_size,-2));
42.             temp_countARR_deg = (180*temp_countARR)/pi;
43.
44. countMatrix_ARR((end+1):(end+length(temp_countARR)),1:5) =...
45.             [temp_countARR, temp_countARR_deg,
46.             zeros(length(temp_countARR(:,1)),3)];
47.         end
48.
49. %% Counting the occurrences of polarizaton rotation rates
50.     for index = 1 : size_tabARR
51.         valARR = TS_ARR_CLD(index,1);
52.         idARR=floor(valARR*(1/bin_size))+1;
53.         countMatrix_ARR(idARR,3)=countMatrix_ARR(idARR,3)+1;
54.     end
55. end
56.
57. %% Computing the ARR probabilities
58.
59. Sum_Prob_ARR = sum(countMatrix_ARR(:,3));
60. countMatrix_ARR(:,4) = countMatrix_ARR(:,3)/Sum_Prob_ARR;
61.
62. %% Cinquième colonne ==> Normalisation par la méthode des trapèzes
63.
64.
65. countMatrix_ARR(:,5) = ...
66.     countMatrix_ARR(:,4)* ...
67.     (1/trapz(countMatrix_ARR(:,1),countMatrix_ARR(:,4)));
68.
69. %% Cration de la figure bar
70.
71. Mean_angular = mean(mean_pol_rot_rate);
72. Mean_ARR_deg = (180*Mean_angular)/pi;
73. std_ARR = mean(std_pol_rot_rate);
74. std_ARR_deg = (180*std_ARR)/pi;
75.
76.
77. N_value_ARR = size(countMatrix_ARR(:,1));
78.
79. id_pos = find(countMatrix_ARR(:,3)~=0);
80. coutMatrix_ARR_pos = countMatrix_ARR(id_pos,:);
81.
82. figure1 = figure;
83.
84. %% Histogramme normalisé PHI !
85. axes1 = axes('Parent',figure1,'YScale','log','YMinorTick','on',...
86.             'Position',[0.0609375 0.089393939393939394 0.92890625
87.             0.886363636363636],...
88.             'FontSize',24,...
89.             'FontName','Calibri');
90. set(axes1,'XTick',0:0.5:9.2);
91. box(axes1,'on');
92. hold(axes1,'all');
93.

```

```

94. h1 =
95. bar(countMatrix_ARR(1:N_value_ARR,1),countMatrix_ARR(1:N_value_ARR,5
96. ),...
97.     'b','BaseValue',min(coutMatrix_ARR_pos(:,5))/2,'Parent',axes1);
98. LEG = legend(['{\bfStatistical Parameters :}',sprintf('\n'),'Mean =
99. {\bf',...
100.     num2str(roundn(Mean_angular,-3)), ' rad/ms}',sprintf('\n'),...
101.     'Standard deviation ({\sigma}) ={\bf', num2str(roundn(std_ARR,-
102. 3)), ' rad/ms}']);
103. set(LEG,'FontName','Calibri','FontSize',20,'Position',[0.266666666666
104. 6667 0.785663654084706 0.29296875 0.133838383838384]);
105. xlabel('Transient rotation rate
106. (rad/ms)','FontName','Calibri','fontsize',24,'fontweight','b');
107. ylabel('PDF','FontName','Calibri','fontsize',24,'fontweight','b');
108. ylim([min(coutMatrix_ARR_pos(:,5))/2
109. max(coutMatrix_ARR_pos(:,5))*1.5]);
110. xlim([0 9.2]);
111. figure(gcf);

```


ANNEXE XXXVIII

AJUSTEMENT DE LA FONCTION DE DENSITÉ DES VITESSES DE ROTATION ANGULAIRE DES ÉTATS DE POALRISATION

Le script-A XXXVII-1 ajuste la fonction de densité de probabilité des états de polarisation par la fonction de densité de Rayleigh, par la fonction de densité exponentielle et par la fonction de densité double qui est résultat du produit de convolution des deux précédentes fonctions de densité de probabilité.

Script-A XXXVIII-1 Ajustement de la fonction de densité de probabilité des vitesses de rotation angulaire des états de polarisation

```
1.  % Create only data PDF from experimental results
2.
3.  data_log = [countMatrix_ARR_pos(:,1), log(countMatrix_ARR_pos(:,4)),
4.  log(countMatrix_ARR_pos(:,5))];
5.  N_value_ARR = size(countMatrix_ARR(:,1));
6.
7.  %%True data...
8.  data_recovery = [countMatrix_ARR(:,1), countMatrix_ARR(:,4),
9.  countMatrix_ARR(:,5)];
10. id_pos = find(countMatrix_ARR(:,5)~=0);
11. data_recovery_pos = data_recovery(id_pos,:);
12.
13.
14. %% Before to do the process... compute the log weight (not the
15. log10)
16.
17. N = sum(countMatrix_ARR(:,3));
18. p = countMatrix_ARR(id_pos,4);
19. H = countMatrix_ARR_pos(:,3);
20.
21. %computing the ln weights
22. sigma_ln_v2 = sqrt((1-p(:,1))./H(:,1));
23. sigma2_ln_v2 = (sigma_ln_v2).^2;
24. weights_ln_v2 = 1./sigma2_ln_v2; % The absolute weights ln
25. mean_weights_ln_v2 = mean(weights_ln_v2);
26. W_ln_relative_v2 = weights_ln_v2 / mean_weights_ln_v2; % The
27. relative size of weights
28.
29. % computing logarithmic (base-10) weights
30. sigma_log10_v2 = (1/log(10)).*(sqrt((1-p(:,1))./H(:,1)));
31. sigma2_log10_v2 = (sigma_log10_v2).^2;
32. weights_log10_v2 = 1./sigma2_log10_v2; % The absolute weights log10
33. mean_weights_log10_v2 = mean(weights_log10_v2);
34. W_log10_relative_v2 = weights_log10_v2 / mean_weights_log10_v2; %
35. The relative size of weights
```

```

36.
37.  %%%%%%%%%%%%%% FITTING RAYLEIGH DISTRIBUTION %%%%%%%%%%%%%%
38.
39.  %Estimation of parameter alpha of the Rayleigh distribution
40.
41.  sigma_th2 = (2*var(data_recovery(:,2)))/(4-pi); % Estimate alpha of
42.  new density data
43.  sigma_th = sqrt(sigma_th2);
44.
45.
46.  id_fG = find((countMatrix_ARR_pos(:,1)<0.9));
47.  log_x = data_log(id_fG,1);
48.  log_x(1,1) = 0.005;
49.  log_y = data_log(id_fG,3); % Density
50.
51.  % modelFun_fM = @(alpha,x) log(sqrt(2/pi))- 3*log(alpha) +
52.  2*log(x)-(x/(sqrt(2)*alpha)).^2; %Log of Maxwellian distribution
53.
54.  modelFun_fG = @(sigma,x) -2*log(sigma) + log(x)-
55.  (x/(sqrt(2)*sigma)).^2; %Log of Rayleigh distribution
56.
57.  % First Option to Non linear fit...
58.
59.  % optionsLSQ = optimset('Display','Iter','TolFun',1e-250,'TolX',1e-
60.  250, 'MaxFunEvals',500000,'MaxIter',500000);
61.  % sigma_nlinfit_log = nlinfit(log_x,log_y, modelFun_fG, sigma_th,
62.  optionsLSQ, 'Weights',W_ln_relative);
63.
64.  % Second Option to Non linear fit...
65.
66.  sigma_th = 10;
67.
68.
69.  options_NLM = statset('nlinfit');
70.  options_NLM.Display = 'Iter';
71.  options_NLM.TolFun = 1e-250;
72.  options_NLM.TolX = 1e-250;
73.  options_NLM.MaxFunEvals = 500000;
74.  options_NLM.MaxIter = 500000 ;
75.
76.  NLM_log_rayleigh = NonLinearModel.fit(log_x,log_y, modelFun_fG,
77.  sigma_th,
78.  'Weights',W_ln_relative_v2(id_fG,1), 'Options',options_NLM);
79.
80.  sigma_nlinfit_log = double(NLM_log_rayleigh.Coefficients(1,1));
81.
82.  % [alpha_fminbnd,FVAL] = fminbnd(modelFun,0,100,optionsLSQ);
83.
84.
85.  x_fG = data_recovery(:,1);
86.
87.  xdata = data_recovery(:,1);
88.  xdata(1,1)=0.005;
89.
90.  fit_rayleigh_log = -2*log(sigma_nlinfit_log) + log(xdata(:,1))-

```

```

91. (xdata(:,1)/(sqrt(2)*sigma_nlinfit_log)).^2;
92. fit_rayleigh_log(:,2)= fit_rayleigh_log(:,1)-2; %Probability
93. fit_rayleigh_lin_test = exp(fit_rayleigh_log);
94.
95. y_lin = fit_rayleigh_lin_test(:,1);
96.
97. modelFun_fG_linear = @(sigma,x) (1/sigma^2) * x.* exp(-
98. (x.^2)/(2*sigma^2));
99. optionsLSQ = optimset('Display','Iter','TolFun',1e-250,'TolX',1e-
100. 250, 'MaxFunEvals',500000,'MaxIter',500000);
101.
102. [sigma_final_linear,Residual_fG,Jacob_fG,Cov_fG,mse_fG] =
103. nlinfit(xdata, y_lin, modelFun_fG_linear, 0.1, optionsLSQ);
104. %
105. [alpha_lsqcurvefit,RESNORM,Residual_fM,flag,output,Lambda_fM,Jacob_1
106. sq_fM] = lsqcurvefit(modelFun_fM,0.1,x_fM,y_fM,0.1,optionsLSQ);
107. % [alpha_fminbnd,FVAL] = fminbnd(modelFun,0,100,optionsLSQ);
108.
109. Jacobian_fG = full(Jacob_fG);
110.
111.
112. CI_fG =
113. nlparci(sigma_final_linear,Residual_fG,'jacobian',Jacobian_fG);
114. sigma_low = CI_fG(1);
115. sigma_high = CI_fG(2);
116.
117. xdata = 0:0.01:9.1;
118. xdata = transpose(xdata);
119.
120. fit_rayleigh_lin = (1/sigma_final_linear^2) * xdata.* exp(-
121. (xdata.^2)/(2*sigma_final_linear^2)); % Density
122.
123. fit_rayleigh_lin_low = (1/sigma_low^2) * xdata.* exp(-
124. (xdata.^2)/(2*sigma_low^2)); % Density
125.
126. fit_rayleigh_lin_high = (1/sigma_high^2) * xdata.* exp(-
127. (xdata.^2)/(2*sigma_high^2)); % Density
128.
129.
130. % diff = abs(max(yff = data_recovery_pos(:,4))-
131. max(fit_maxwell_log(:,2)));
132.
133. numerator = sum((data_recovery(:,3) -
134. mean(data_recovery(:,3))).*(fit_rayleigh_lin(:,1) -
135. mean(fit_rayleigh_lin(:,1))));
136. denominator = sqrt(sum((data_recovery(:,3) -
137. mean(data_recovery(:,3))).^2)).*sqrt(sum((fit_rayleigh_lin(:,1) -
138. mean(fit_rayleigh_lin(:,1))).^2));
139. Pearson_rayleigh = (numerator/denominator);
140.
141. Pearson_rayleigh_2 = Pearson_rayleigh.^2;
142. Pearson_rayleigh = sqrt(Pearson_rayleigh_2);
143.
144.
145.

```

```

146. %%%%% FITTING EXPONENTIAL DISTRIBUTION %%%%%%%%%%%%%%
147.
148. data_high_value = [countMatrix_ARR(111:201,1),
149. countMatrix_ARR(111:201,3),countMatrix_ARR(111:201,4),countMatrix_AR
150. R(111:201,5)];
151. id_pos = find(data_high_value(:,4)~=0);
152. data_high_value_pos = data_high_value(id_pos,:);
153.
154. data_log = [data_high_value_pos(:,1), log(data_high_value_pos(:,3)),
155. log(data_high_value_pos(:,4))];
156.
157. % Find the weights of the high value
158. H_high = data_high_value_pos(:,2);
159. p_high = data_high_value_pos(:,3);
160.
161. % sigma_high_ln = log(1+(1./sqrt(H_high(:,1))));
162.
163.
164. sigma_high_ln = 1./sqrt(H_high(:,1));
165. sigma2_high_ln = (sigma_high_ln).^2;
166. weights_high_ln = 1./sigma2_high_ln; % The absolute weights ln
167. mean_weights_high = mean(weights_high_ln);
168. W_high_relative = weights_high_ln / mean_weights_high; % The
169. relative size of weights
170.
171.
172. % Second version of computing high ARR value
173. sigma_high_ln_v2 = sqrt((1-p_high(:,1))./H_high(:,1));
174. sigma2_high_ln_v2 = (sigma_high_ln_v2).^2;
175. weights_high_ln_v2 = 1./sigma2_high_ln_v2; % The absolute weights ln
176. mean_weights_high_v2 = mean(weights_high_ln_v2);
177. W_high_relative_v2 = weights_high_ln_v2 / mean_weights_high_v2; %
178. The relative size of weights
179.
180.
181. x_log = data_log(:,1);
182. y_log = data_log(:,3);
183.
184. % modelFun_fE = @(lambda,x) log(lambda) - (lambda*x);
185. modelFun_fE = @(lambda,x) (lambda(1)*x) + (lambda(2));
186.
187.
188. % Second option
189. NLM_log_exp = NonLinearModel.fit(x_log, y_log, modelFun_fE, [0,0.5],
190. 'Weights',weights_high_ln_v2,'Options',options_NLM);
191. lambda_log = double(NLM_log_exp.Coefficients(:,1));
192. CI_fE = coefCI(NLM_log_exp);
193. %%%
194. a_exp = exp(lambda_log(2));
195. b_exp = lambda_log(1);
196. b_exp_low_log= CI_fE(1,1);
197. b_exp_high_log= CI_fE(1,2);
198. a_exp_low_log = CI_fE(2,1);
199. a_exp_high_log= CI_fE(2,2);
200.

```

```

201. % [alpha_lsqcurvefit, RESNORM, Residual_fm, flag, output, Lambda_fm, Jacob_
202. lsq_fm] = lsqcurvefit(modelFun_fm, 0.1, x_fm, y_fm, 0, 1, optionsLSQ);
203. % [alpha_fminbnd, FVAL] = fminbnd(modelFun, 0, 100, optionsLSQ);
204.
205. a_exp_low = exp(a_exp_low_log);
206. b_exp_low = b_exp_low_log;
207. a_exp_high = exp(a_exp_high_log);
208. b_exp_high = b_exp_high_log;
209.
210. lambda_final_linear = -b_exp;
211. lambda_low = -b_exp_low;
212. lambda_high = -b_exp_high;
213.
214. rho_exp_final_linear = a_exp/lambda_final_linear;
215. rho_exp_low = a_exp_low/lambda_low;
216. rho_exp_high = a_exp_high/lambda_high;
217.
218.
219. % Calcul des densités...
220. % xdata_high = data_high_value(:,1);
221.
222. xdata_high = 1.1:0.01:9.1;
223. xdata_high = transpose(xdata_high);
224.
225. yfit_exp = a_exp*exp(b_exp*xdata_high);
226. yfit_exp_low = a_exp_low*exp(b_exp_low*xdata_high);
227. yfit_exp_high = a_exp_high*exp(b_exp_high*xdata_high);
228.
229. numerator_exp = sum((countMatrix_ARR(111:end,5) -
230. mean(countMatrix_ARR(111:end,5))).*(yfit_exp(:,1) -
231. mean(yfit_exp(:,1))));
232. denominator_exp = sqrt(sum((countMatrix_ARR(111:end,5) -
233. mean(countMatrix_ARR(111:end,5)).^2)).*sqrt(sum((yfit_exp(:,1) -
234. mean(yfit_exp(:,1)).^2)));
235. Pearson_coeff_exp1 = (numerator_exp/denominator_exp);
236.
237. %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% fGEI DISTRIBUTION   VERSION 1.0
238. %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
239.
240.
241. sigma_th=sigma_final_linear;
242. lambda_th= lambda_final_linear;
243. rho_th = 0.000001;
244.
245. init_parameter = [rho_th,lambda_th,sigma_th];
246.
247. % result_fit_max = fit_maxwell_pdf(
248. countMatrix_ARR(:,1),countMatrix_ARR(:,4),0);
249. % alpha_th = result.a;
250. % ==> Need to use the logarithmique scale !
251.
252. id_fGEI = find((countMatrix_ARR_pos(:,1)<0.51) |
253. (countMatrix_ARR_pos(:,1)>1.1 & countMatrix_ARR_pos(:,1)<2.01));
254. data_log = [data_recovery_pos(id_fGEI,1),
255. log10(data_recovery_pos(id_fGEI,2)),log10(data_recovery_pos(id_fGEI,

```

```

256. 3)]];
257.
258. x = data_log(:,1);
259. x(1,1) = 0.005;
260.
261. y = data_log(:,3);
262.
263. init_rho = 0.0000001;
264.
265.
266. modelFun_fGEI = @(rho,x) rho*((lambda_th * exp((lambda_th^2 *
267. sigma_th^2)/2) * exp(-lambda_th*x)) .* ...
268. ( ( sqrt(pi/2) * lambda_th * sigma_th * erf((x-
269. (lambda_th*sigma_th^2))/(sqrt(2)*sigma_th)) - ...
270. exp(-((x-
271. (lambda_th*sigma_th^2)).^2)/(2*sigma_th^2)) ) - ...
272. ( sqrt(pi/2) * lambda_th * sigma_th * erf((0-
273. (lambda_th*sigma_th^2))/(sqrt(2)*sigma_th)) - ...
274. exp(-((0-
275. (lambda_th*sigma_th^2)).^2)/(2*sigma_th^2)) ) )...
276. + (1-rho)*((1/sigma_th^2) * x.* exp(-
277. (x.^2)/(2*sigma_th^2)));
278.
279. modelFun_fGEI_Advance = @(p,x) p(1)*((p(2) * exp((p(2)^2 *
280. p(3)^2)/2) * exp(-p(2)*x)) .* ...
281. ( ( sqrt(pi/2) * p(2) * p(3) * erf((x-
282. (p(2)*p(3)^2))/(sqrt(2)*p(3)) ) - ...
283. exp(-((x- (p(2)*p(3)^2)).^2)/(2*p(3)^2)) ) -
284. ...
285. ( sqrt(pi/2) * p(2) * p(3) * erf((0-
286. (p(2)*p(3)^2))/(sqrt(2)*p(3)) ) - ...
287. exp(-((0- (p(2)*p(3)^2)).^2)/(2*p(3)^2)) )
288. ) )...
289. + (1-p(1))*((1/p(3)^2) * x.* exp(-
290. (x.^2)/(2*p(3)^2)));
291.
292.
293. optionsLSQ = optimset('Display','Iter','TolFun',1e-250,'TolX',1e-
294. 250, 'MaxFunEvals',500000,'MaxIter',500000,'FunValCheck','on');
295.
296.
297.
298. %% NONLINFIT
299.
300. % First option
301.
302. % [rho_final,Residual_rho,Jacob_rho,Cov_rho,mse_rho] = nlinfit(x,
303. y,@(p,x)log10(modelFun_fGEI(p,x)),init_rho,
304. optionsLSQ,'Weights',W_log10_relative(id_fGEI,1));
305. [rho_final,Residual_rho,Jacob_rho,Cov_rho,mse_rho] = nlinfit(x,
306. y,@(p,x)log10(modelFun_fGEI(p,x)),init_rho, optionsLSQ);
307. Jacobian_rho = full(Jacob_rho);
308. CI_rho_final =
309. nlparci(rho_final,Residual_rho,'jacobian',Jacobian_rho,
310. 'alpha',0.05);

```

```

311.
312.   %%%
313.   rho_final_low = CI_rho_final(1);
314.   rho_final_high = CI_rho_final(2);
315.   %
316.   %Second option
317.
318.   sigma_final = sigma_final_linear;
319.   lambda_final = lambda_final_linear;
320.
321.   % rho_final = 0.0038;
322.
323.
324.   x = 0:0.01:9.1;
325.   x = transpose(x);
326.
327.   fit_fGEI_lin = rho_final*((lambda_final * exp((lambda_final^2 *
328.   sigma_final^2)/2) * exp(-lambda_final*x)) .* ...
329.   ( ( sqrt(pi/2) * lambda_final * sigma_final *
330.   erf((x-(lambda_final*sigma_final^2))/(sqrt(2)*sigma_final)) - ...
331.   exp(-((x-
332.   (lambda_final*sigma_final^2)).^2)/(2*sigma_final^2)) ) - ...
333.   ( sqrt(pi/2) * lambda_final * sigma_final *
334.   erf((0-(lambda_final*sigma_final^2))/(sqrt(2)*sigma_final)) - ...
335.   exp(-((0-
336.   (lambda_final*sigma_final^2)).^2)/(2*sigma_final^2)) ) ))...
337.   +(1-rho_final)*((1/sigma_final^2) * x.*
338.   exp(-(x.^2)/(2*sigma_final^2)));
339.
340.
341.   numerator = sum((data_recovery(:,3)-
342.   mean(data_recovery(:,3))).*(fit_fGEI_lin(:,1) -
343.   mean(fit_fGEI_lin(:,1))));
344.   denominator = sqrt(sum((data_recovery(:,3) -
345.   mean(data_recovery(:,3))).^2)).*sqrt(sum((fit_fGEI_lin(:,1) -
346.   mean(fit_fGEI_lin(:,1))).^2));
347.   Pearson_fGEI = (numerator/denominator);
348.
349.   fit_fGEI_lin_low = rho_final_low*((lambda_low * exp((lambda_low^2 *
350.   sigma_low^2)/2) * exp(-lambda_low*x)) .* ...
351.   ( ( sqrt(pi/2) * lambda_low * sigma_low *
352.   erf((x-(lambda_low*sigma_low^2))/(sqrt(2)*sigma_low)) - ...
353.   exp(-((x-
354.   (lambda_low*sigma_low^2)).^2)/(2*sigma_low^2)) ) - ...
355.   ( sqrt(pi/2) * lambda_low * sigma_low *
356.   erf((0-(lambda_low*sigma_low^2))/(sqrt(2)*sigma_low)) - ...
357.   exp(-((0-
358.   (lambda_low*sigma_low^2)).^2)/(2*sigma_low^2)) ) ) ))...
359.   +(1-rho_final_low)*((1/sigma_low^2) * x.*
360.   exp(-(x.^2)/(2*sigma_low^2)));
361.
362.
363.
364.   fit_fGEI_lin_high = rho_final_high*((lambda_high *
365.   exp((lambda_high^2 * sigma_high^2)/2) * exp(-lambda_high*x)) .* ...

```

```

366.      ( ( sqrt(pi/2) * lambda_high * sigma_high *
367. erf((x-(lambda_high*sigma_high^2))/(sqrt(2)*sigma_high)) - ...
368.      exp(-((x-
369. (lambda_high*sigma_high^2)).^2)/(2*sigma_high^2)) ) - ...
370.      ( sqrt(pi/2) * lambda_high * sigma_high *
371. erf((0-(lambda_high*sigma_high^2))/(sqrt(2)*sigma_high)) - ...
372.      exp(-((0-
373. (lambda_high*sigma_high^2)).^2)/(2*sigma_high^2)) ) )...
374.      +(1-rho_final_high)*((1/sigma_high^2) * x.*
375. exp(-(x.^2)/(2*sigma_high^2)));

```

ANNEXE XXXIX

CALCUL DES BARRES D'ERREURS ET DES INCERTITUDES D'UN HISTOGRAMME AVEC UNE ÉCHELLE LOGARITHMIQUE

Navidi (2010, p.210) démontre que l'erreur statistique σ_{px} de chaque classe d'une distribution statistique est formulée par l'équation (A XXXIX-1).

$$\sigma_{px} = \sqrt{\frac{\hat{p}_x \times (1 - \hat{p}_x)}{n}} \quad (\text{A XXXIX-1})$$

Tirée de Navidi (2010, p.210)

La valeur de chaque probabilité P_x est formulée par la valeur de probabilité observée (ou estimée) \hat{p}_x et de son incertitude σ_{px} selon l'équation (XXXIX.2).

$$P_x = \hat{p}_x \pm \sigma_{px} \quad (\text{A XXXIX-2})$$

Plus de $7,7 \times 10^9$ valeurs de vitesse de rotation angulaire ont été calculées dans cette expérience de *monitoring*. Lorsque la méthode pondérée des moindres carrés est effectuée au moyen d'une échelle logarithmique, Crawford *et al.* (2013) propose de calculer le logarithme des valeurs d'incertitude σ_{px} . Le calcul des incertitudes au moyen d'une échelle logarithmique se justifie par l'obtention de très faibles valeurs de probabilités obtenues. Le seuil minimal de probabilité est proche de 10^{-10} après que la méthode de calcul de la distribution statistique a été appliquée (Voir Figure 4.4). Les équations (A XXXIX-3- A XXXIX-5) présentent la démonstration du calcul du logarithme des valeurs de probabilité \hat{p}_x d'une variable x réalisée par Crawford *et al.* (2013).

$$\log(P_x) = \log\left(\hat{p}_x + \sqrt{\frac{\hat{p}_x \times (1 - \hat{p}_x)}{n}}\right) \quad (\text{A XXXIX-3})$$

$$\log(P_x) = \log\left(\hat{p}_x \left(1 + \sqrt{\frac{(1 - \hat{p}_x)}{\hat{p}_x \times n}}\right)\right) \quad (\text{A XXXIX-4})$$

$$\log(P_x) = \log(\hat{p}_x) + \log\left(1 + \sqrt{\frac{(1 - \hat{p}_x)}{N_x}}\right) \quad (\text{A XXXIX-5})$$

L'équation (A XXXIX-5) présente la formule du logarithme des valeurs de probabilité P_x en fonction des valeurs de probabilités estimées \hat{p}_x et du nombre d'occurrences N_x associées à chacune des classes de la distribution statistique. Crawford *et al.* (2013) proposent de simplifier l'équation $\log(1 + X)$ en utilisant le développement en série de Taylor. L'équation (A XXXIX-6) formule la simplification de l'expression logarithmique $\log(1 + X)$ par le développement en série de Taylor.

$$\ln(1 + x) \cong x - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{2} \dots \cong x \quad (\text{A XXXIX-6})$$

Tirée de la démonstration de Travis Crawford (2013)

Crawford *et al.* (2013) formulent l'expression simplifiée des valeurs de probabilité au moyen d'une échelle logarithmique à partir des équations (A XXXIX-5) et (A XXXIX-6).

$$\ln(P_x) = \ln(\hat{p}_x) \pm \sqrt{\frac{(1 - \hat{p}_x)}{N_x}} \quad (\text{A XXXIX-7})$$

Le calcul de la pondération sous une échelle logarithmique (à base 10) est formulé selon l'équation (A XXXIX-8).

$$\log_{10}(P_x) = \frac{\ln(P_x)}{\log 10} = \ln(\hat{p}_x) \pm \frac{1}{\ln 10} \times \sqrt{\frac{(1 - \hat{p}_x)}{N_x}} \quad (\text{A XXXIX-8})$$

L'équation (A XXXIX-9) déduit le calcul des barres d'erreurs $\sigma_{\log_{10}}$ à partir de l'équation (5.18). Cette formule a finalement été énoncée par Crawford *et al.* (2013).

$$\sigma_{\log_{10}} = \frac{1}{\ln 10} \times \sqrt{\frac{(1 - \hat{p}_x)}{N_x}} \quad (\text{A XXXIX-9})$$

Plus la valeur de probabilité estimée d'une classe de vitesse de rotation angulaire est faible, plus la valeur du poids $\omega_{\log_{10}}$ de cette même classe est faible aussi. Wolberg (2005, p.49) énonce que les poids de chaque valeur de l'histogramme sont proportionnels aux occurrences N_x des vitesses de rotation angulaire lorsque celles-ci sont calculées au moyen d'une échelle logarithmique. L'équation (A XXXIX-10) formule le calcul des poids logarithmiques (à base-10) de chaque classe en fonction des occurrences N_x et des probabilités observées \hat{p}_x des vitesses de rotation angulaire.

$$\omega_{\log_{10}} = \frac{1}{\sigma^2} = (\ln 10)^2 \times \frac{N_x}{(1 - \hat{p}_x)} \quad (\text{A XXXIX-10})$$

ANNEXE XL

PROPRIÉTÉS DU MODÈLE *NONLINEARMODEL* DU MODULE *STATISTICS TOOLBOX*

NLM_log_rayleigh <1x1 NonLinearModel>			
Property ^	Value	Min	Max
MSE	0.0393	0.0393	0.0393
Iterative	<1x1 struct>		
Robust	[]		
Residuals	<90x4 dataset>		
Fitted	<90x1 double>	-12.4070	1.3488
RMSE	0.1983	0.1983	0.1983
WeightedResiduals	<90x4 dataset>		
Formula	<1x1 classreg.regr.NonLinearFormula>		
LogLikelihood	-24.7374	-24.7374	-24.7374
DFE	89	89	89
SSE	3.4994	3.4994	3.4994
SST	147.2289	147.2289	147.2289
SSR	171.6117	171.6117	171.6117
CoefficientCovariance	6.4194e-07	6.4194e-07	6.4194e-07
CoefficientNames	<1x1 cell>		
NumCoefficients	1	1	1
NumEstimatedCoefficients	1	1	1
Coefficients	<1x4 dataset>		
Rsquared	<1x1 struct>		
ModelCriterion	<1x1 struct>		
VariableInfo	<2x4 dataset>		
ObservationInfo	<90x4 dataset>		
Variables	<90x2 dataset>		
NumVariables	2	2	2
VariableNames	<2x1 cell>		
NumPredictors	1	1	1
PredictorNames	<1x1 cell>		
ResponseName	'y'		
NumObservations	90	90	90
ObservationNames	<0x0 cell>		

Figure-A XL-1 Propriétés et extensions du modèle *NonLinearModel.fit*

ANNEXE XLI

AFFICHAGE DE LA FONCTION DE DENSITÉ DE PROBABILITÉ DES VITESSES DE ROTATION ANGULAIRE ET DES AJUSTEMENTS

Le script-A XL-1 permet l'affichage de la fonction de densité de probabilité des vitesses de rotation angulaire des états de polarisation ainsi que des courbes d'ajustement soit la fonction de densité de Rayleigh, exponentielle et la fonction de densité double qui est le résultat de la convolution des deux précédentes fonctions de densité.

Script-A XLI-1 Affichage de la fonction de densité de probabilité des vitesses de rotation angulaire et des courbes d'ajustement

```
1. figure1 = figure;
2.
3. h1 =
4. bar(data_recovery(1:N_value_ARR,1),data_recovery(1:N_value_ARR,3),...
5.
6.     'b','BaseValue',min(data_recovery_pos(:,3))/3);
7.
8. hold on;
9.
10.
11. h2 = plot(xdata(:,1), fit_rayleigh_lin(:,1));
12. set(h2,'color','r');
13. set(h2,'linewidth',2);
14. set(h2,'marker','none');
15. hold on
16.
17. h3 = plot(xdata(:,1), fit_rayleigh_lin_low(:,1),'--');
18. set(h3,'color','r');
19. set(h3,'linewidth',2);
20. set(h3,'marker','none');
21. hold on
22.
23. h4 = plot(xdata(:,1), fit_rayleigh_lin_high(:,1),'--');
24. set(h4,'color','r');
25. set(h4,'linewidth',2);
26. set(h4,'marker','none');
27. hold on
28.
29. h5 = plot(xdata_high(:,1), yfit_exp(:,1));
30. set(h5,'color','m');
31. set(h5,'linewidth',2);
32. set(h5,'marker','none');
33. hold on
34.
35. h6 = plot(xdata_high(:,1), yfit_exp_low(:,1),'--');
36. set(h6,'color','m');
37. set(h6,'linewidth',2);
```

```

38. set(h6, 'marker', 'none');
39. hold on
40.
41. h7 = plot(xdata_high(:,1), yfit_exp_high(:,1), '--');
42. set(h7, 'color', 'm');
43. set(h7, 'linewidth', 2);
44. set(h7, 'marker', 'none');
45. hold on
46.
47. xlabel('Transient rotation rate (rad/ms)',...
48. 'FontName', 'Calibri', 'fontsize', 22, 'fontweight', 'b');
49. ylabel('PDF', 'FontName', 'Calibri', 'fontsize', 24, 'fontweight', 'b');
50. ylim([min(data_recovery_pos(:,3))/3
51. max(data_recovery_pos(:,3))*1.5]);
52. xlim([-0.01 9.2]);
53.
54. set(gca, 'XTick', 0:0.5:9.2);
55. set(gca, 'YTick', [1e-8 1e-7 1e-6 1e-5 1e-4 1e-3 1e-2 1e-1 1e0]);
56. set(gca, 'Position', [0.0609375 0.089393939393939394 0.92890625
57. 0.88636363636363636]);
58. set(gca, 'FontSize', 20, 'FontName', 'Calibri');
59. set(gca, 'YScale', 'log');
60. set(gca, 'YMinorTick', 'on');
61. grid on;
62.
63. hold off;
64.
65.
66. %% FIGURE 2 ==> PLOT fMEI fonction with uncertainties
67.
68. figure2 = figure;
69.
70. g1 =
71. bar(data_recovery(1:N_value_ARR,1), data_recovery(1:N_value_ARR,3), ...
72. 'b', 'BaseValue', min(data_recovery_pos(:,3))/3);
73.
74. hold on;
75.
76.
77. g2 = plot(xdata(:,1), fit_fGEI_lin(:,1) );
78. set(g2, 'color', 'r');
79. set(g2, 'linewidth', 2);
80. set(g2, 'marker', 'none');
81. hold on
82.
83.
84. g3 = plot(xdata(:,1), fit_fGEI_lin_low(:,1), '--');
85. set(g3, 'color', 'r');
86. set(g3, 'linewidth', 2);
87. set(g3, 'marker', 'none');
88. hold on
89.
90. g4 = plot(xdata(:,1), fit_fGEI_lin_high(:,1), '--');
91. set(g4, 'color', 'r');
92. set(g4, 'linewidth', 2);

```

```
93. set(g4,'marker','none');
94. hold on
95.
96.
97. xlabel('Transient rotation rate (rad/ms)','FontName',...
98.        'Calibri','fontSize',22,'fontweight','b');
99. ylabel('PDF','FontName','Calibri','fontSize',24,'fontweight','b');
100. ylim([min(data_recovery_pos(:,3))/3
101.        max(data_recovery_pos(:,3))*1.5]);
102. xlim([-0.01 9.2]);
103.
104. set(gca,'XTick',0:0.5:9.2);
105. set(gca,'YTick',[1e-8 1e-7 1e-6 1e-5 1e-4 1e-3 1e-2 1e-1 1e0]);
106. set(gca,'Position',[0.0609375 0.0893939393939394 0.92890625
107.                    0.886363636363636]);
108. set(gca,'FontSize',20,'FontName','Calibri');
109. set(gca,'YScale','log');
110. set(gca,'YMinorTick','on');
111. grid on;
112.
113.
114. hold off;
```


ANNEXE XLII

DÉTECTION ET ENREGISTREMENT DES EFFETS TRANSITOIRES DE POLARISATION

Le script-A XLII-1 effectue le calcul et la détection des effets transitoires de polarisation parmi les vitesses de rotation angulaire des états de polarisation mesurés avec le polarimètre.

Script-A XLII-1 Calcul et détection des effets transitoires de polarisation

```
1.  %%Retrieve the size of all matrices stored in the OCLD data
2.  structure
3.  nMatrix = length(Struct_ARR);
4.
5.  threshold1=0.9;
6.  threshold2=0.45;
7.
8.  for fileIx = 1:nMatrix
9.      fileIx
10.
11.      ID = ['id',num2str(fileIx)];
12.      TS_ARR_CLD = Struct_ARR(1,2).(ID);
13.
14.      try
15.          FileData = load(chemincomplet);
16.
17.      catch
18.          fprintf('Could not load: %s\n', nomfichier);
19.          continue;
20.      end
21.
22.
23.      S = FileData.sop.';
24.      P = FileData.p;
25.
26.  %% High polarizaton rotation rate detection
27.
28.      peak_rotation_rate = find(TS_ARR_CLD(1:(end-
29.  200),1)>=threshold1);
30.
31.      if (isempty(peak_rotation_rate)==0)
32.
33.          for i = 1:length(peak_rotation_rate(:,1));
34.
35.              % Checking if the pol transient is well unique
36.
37.
38.              if(i==1)
39.                  previous_peak = -1;
40.              else previous_peak = peak_rotation_rate(i-1,1);
41.
```

```

42.         end
43.
44.         if(peak_rotation_rate(i,1)-previous_peak>200)
45.             % fprintf('plop')
46.             peak1 = peak_rotation_rate(i,1);%pic debut a 0.9
47.
48.             for j=peak1+200:-1:peak1  %% Definition de la fenetre
49. maximal de 10 ms
50.
51.                 if TS_ARR_CLD(j,1)>=threshold1
52.                     peak2=j;    %pic fin a 0.9
53.
54.                     % Identification du début d'un transient
55.                     ind_debut = peak1;
56.
57.                     while TS_ARR_CLD(ind_debut,1)>threshold2
58.                         ind_debut = ind_debut-1;
59.                     end
60.                     peakdebut = ind_debut; %% Le point de départ
61. d'un transient est egal à 0.45 rad/ms
62.
63.                     % Identification de la fin d'un d'un transient
64.                     ind_fin = peak2;
65.
66.                     while TS_ARR_CLD(ind_fin,1)>threshold2
67.                         ind_fin = ind_fin+1;
68.                     end
69.
70.                     peakfin = ind_fin;%% Le point final d'un
71. transient est egal à 0.45 rad/ms
72.
73.                     pol_transients(end+1,1:4) =
74. [fileIx,peakdebut,peakfin,TS_ARR_CLD(peakdebut,2)];
75.
76.                 end
77.
78.             end
79.
80.         end
81.
82.     end
83.
84. end
85.
86. end
87. peak_rotation_rate=[];
88.
89.
90. end

```

ANNEXE XLIII

CALCUL DES PARAMÈTRES CARACTÉRISTIQUES DES EFFETS TRANSITOIRES DE POLARISATION

Le script-A XLIII-1 effectue le calcul de la durée, de la valeur d'élasticité, de la valeur maximale des vitesses de rotation angulaire ainsi que de l'intervalle de temps séparant deux effets transitoires de polarisation durant la même période d'acquisition à partir de la matrice *pol_transients* obtenue après exécution du script-A XLII-1 (Voir ANNEXE XLII, p.).

Script-A XLIII-1 Calcul et enregistre les paramètres caractéristiques des effets transitoires de polarisation

```
1.
2.   ind=1;
3.   ind2=1;
4.   nMatrix = length(Struct_ARR);
5.   TSmat=[];
6.   pol_transient_durations=[];
7.   pol_transient_deflections = [];
8.   dt = 0.05;
9.   dt_second = 0.5/1000;
10.  max_SOP_changes=[];
11.  TS_transients = [];
12.  matdf = [];
13.
14.  for idFile = 1:nMatrix
15.
16.      idtransient = find(pol_transients(:,1) == idFile);
17.      n_pol_transient = length(idtransient);
18.
19.      if ( isempty(idtransient) == 0)
20.
21.          try
22.
23.              transientID = ['id',num2str(idFile)];
24.              TS_ARR_CLD = Struct_ARR(1,2).(transientID);
25.
26.              for ind_transient = 1:n_pol_transient
27.
28.                  % Extraction du début et de la fin de chaque effet
29.                  transitoire
30.
31.                      startTransient =
32.                      pol_transients(idtransient(ind_transient),2);
33.                      endTransient =
34.                      pol_transients(nevents(ind_transient),3);
35.
36.                      %%Computing the duration of the polarization
37.                  transient
38.                      duration = (endTransient - startTransient)*dt;
```

```

39.         pol_transient_durations(end+1,1:2) = [idFile,
40. duration];
41.
42.         %Computing the maximal polarization rotation rate
43.         max_transient_rotation_rate =
44. max(TS_ARR_CLD(startTransient:endTransient,1));
45.         max_SOP_changes(end+1,1:4) =
46. [idFile,startTransient,endTransient,max_transient_rotation_rate];
47.
48.         %% PROCESS : angular deflection computation
49.
50.         for k=startTransient+1:endTransient
51.
52. matdf(1:3,ind2)=S(1:3,startTransient).*S(1:3,k);
53.             ind2=ind2+1;
54.         end
55.
56.
57.
58.         %% Computing all transient magnitude
59.         dot_product_relative = sum(matdf); % Size 1-by-
60. (nSamples-1)
61.         dot_product_relative(dot_product_relative < -1) =
62. -1; % Avoids numerical artifacts
63.         dot_product_relative(dot_product_relative > +1) =
64. +1;
65.
66.         angular_velocity_relative =
67. acos(dot_product_relative); % Units of [rad/msec]
68.
69. angular_velocity_relative=transpose(angular_velocity_relative);
70.
71.         %% Extratcing the maximum transient deflection
72.         pol_transient_deflections(end+1,1:5)=
73. [idFile,max(angular_velocity_relative(:,1))];
74.
75.
76.         %Computing the time interval between adjacent pol
77. transients
78.         if( ind_transient >=2)
79.             ind_preTransient = idtransient(ind_transient-
80. 1,1);
81.             end_previous_peak =
82. pol_transients(ind_preTransient,3);
83.
84.             ind_nextTransient =
85. idtransient(ind_transient,1);
86.             start_next_transient =
87. pol_transients(ind_nextTransient,2);
88.
89.             time_interval = (start_next_transient-
90. end_previous_peak)*dt_second;
91.
92.             %% Time interval must exceed 10-ms
93.             if (time_interval >= 0.01)

```

```
94.             Time_intervals(end+1,1:3) = [ID,
95. ind_transient,time_interval];
96.             end
97.         end
98.
99.             %% Reinitialisation
100.            ind2=1;
101.            dot_product_relative = [];
102.            angular_velocity_relative = [];
103.            matdf = [];
104.
105.        end
106.
107.        catch (fprintf('Could not load: %i\n', ID);
108.        continue;
109.    end
110.
111.
112.    end
113.
114. end
115.
```


ANNEXE XLIV

CALCUL DE LA FONCTION DE DENSITÉ DE PROBABILITÉ DES VALEURS MAXIMALES DE VITESSE DE ROTATION ANGULAIRE ET D'ÉLASTICITÉ DES EFFETS TRANSITOIRES DE POLARISATION

Script-A XLIV-1 Calcul de la fonction de densité de probabilité associant les valeurs maximales de vitesse de rotation angulaire et les valeurs d'élasticité des effets transitoires de polarisation

```
1. %% Taille de la matrice ARR
2. countMatrix_ARR_deflections= zeros(113,83);
3. % 42 * 0.1 ==> Transient rotation rates à partir de 0.9
4. % 200*0.05 (fenêtre de 10 milliseconde pour la durée d'un
5. transient)
6. countMatrix_ARR = zeros(83,2);
7.
8. taille_valARR = length(countMatrix_ARR(:,1));
9.
10. cellule = 1,
11. ValARRScan = 0.9;
12.
13.     for cellule = 1:taille_valARR;
14.         countMatrix_ARR(cellule,1) = ValARRScan;
15.         countMatrix_ARR(cellule,2) = (180*ValARRScan)/pi;
16.         ValARRScan = ValARRScan + 0.1;
17.     end
18.
19.
20. %% CREATION DE LA MATRICE INTEGRANT LES VALEURS DES TRANSIENTS
21. DURATIONS %%
22.
23. %% Taille de la matrice ARR
24. countMatrix_Deflection= zeros(113,1);
25.
26. % Mettre en place les dimensions de la matrice en question...
27. taille_valDeflection = length(countMatrix_Deflection(:,1));
28.
29. cellule = 1;
30. ValDeflectionScan = 0;
31.
32.     for cellule = 1:taille_valDeflection;
33.         countMatrix_Deflection(cellule,1)= ValDeflectionScan;
34.         ValDeflectionScan = ValDeflectionScan + 0.0125;
35.         ValDeflectionScan = roundn(ValDeflectionScan,-3);
36.     end
37.
38. %% Obtention des valeurs mesurées de max_SOP_chages et des durées
39. respectives
40.
41. cellule=1;
```

```

42. transient_polarimeter = [];
43. dt=0.05;
44.     for cellule=1:length(pol_transient_deflections(:,1))
45.         transient_polarimeter(cellule,1:2) = ...
46.             [max_SOP_changes(cellule,4),...
47.              pol_transient_deflections(cellule,2)];
48.     end
49.
50. % Compte des couples Max_ARR & Durations_ARR
51.
52. taille_mesure = length(transient_polarimeter(:,1));
53.
54. id_transient = 1;
55. id_deflection = 1;
56.
57.     for cellule = 1:taille_mesure;
58.
59.         val_Max_ARR = transient_polarimeter(cellule,1);
60.         for ind_ARR=1:(taille_valARR-1)
61.
62.
63.             %% On compte les itérations maintenant !!
64.             if (val_Max_ARR >= countMatrix_ARR(ind_ARR,1) &&...
65.                 val_Max_ARR < countMatrix_ARR(ind_ARR+1,1));
66. %                 sprintf('OK ARR')
67.                 break;
68.             end
69.         end
70.
71.         val_deflection_ARR = transient_polarimeter(cellule,2);
72.
73.         for ind_deflection=1:(taille_valDeflection-1);
74.
75.
76.             if val_deflection_ARR >=
77. countMatrix_Deflection(ind_deflection,1) &&...
78.                 val_deflection_ARR <
79. countMatrix_Deflection(ind_deflection+1,1)
80. %                 sprintf('OK Duree')
81. %                 countMatrix_Duree(ind_duration,1)
82.                 break;
83.             end
84.         end
85.         countMatrix_ARR_deflections(ind_deflection,ind_ARR) =...
86.             countMatrix_ARR_deflections(ind_deflection,ind_ARR)+
87.             ind_ARR=1;
88.             ind_deflection=1;
89.         end
90.
91.         %% Computing Joint Probability values
92.         Sum_Prob = sum(sum(countMatrix_ARR_deflections));
93.         matrix_prob_3D = countMatrix_ARR_deflections/Sum_Prob;

```

ANNEXE XLV

CALCUL DE LA FONCTION DE DENSITÉ DE PROBABILITÉ CONJOINTE DES VALEURS MAXIMALES DE VITESSE DE ROTATION ANGULAIRE ET DES VALEURS DE DURÉE DES EFFETS TRANSITOIRES DE POLARISATION

Script-A XLV-1 Calcul de la fonction de densité de probabilité conjointe des valeurs maximales de vitesse de rotation angulaire et des valeurs de durée des effets transitoires de polarisation

```
1. countMatrix_ARR_durations= zeros(200,83);
2. % 42 * 0.1 ==> Transient rotation rates à partir de 0.9
3. % 200*0.05 (fenêtre de 10 milliseconde pour la durée d'un
4. transient)
5.
6. %%%%%% CREATION DE LA MATRICE INTEGRANT LES VALEURS DES TRANSIENTS
7. %%%%%% ROTATIONS RATES %%%%%%%
8. countMatrix_ARR = zeros(83,2);
9. taille_valARR = length(countMatrix_ARR(:,1));
10. cellule = 1,
11. ValARRScan = 0.9;
12.
13.     for cellule = 1:taille_valARR;
14.         countMatrix_ARR(cellule,1) = ValARRScan;
15.         countMatrix_ARR(cellule,2) = (180*ValARRScan)/pi;
16.         ValARRScan = ValARRScan + 0.1;
17.     end
18.
19.     %% CREATION DE LA MATRICE INTEGRANT LES VALEURS DES TRANSIENTS
20.     DURATIONS %%
21.
22.
23.     %% Taille de la matrice ARR
24.     countMatrix_Duree= zeros(200,1);
25.
26.     % Mettre en place les dimensions de la matrice en question...
27.     taille_valDuree = length(countMatrix_Duree(:,1));
28.
29.     cellule = 1;
30.     ValDureeScan = 0.05;
31.
32.     for cellule = 1:taille_valDuree;
33.         countMatrix_Duree(cellule,1)= ValDureeScan;
34.         ValDureeScan = ValDureeScan + 0.05;
35.         ValDureeScan = roundn(ValDureeScan,-3);
36.     end
37.     %% Obtention des valeurs mesurées de max_SOP_chages et des durées
38.     respectives
39.     cellule=1;
40.     transient_polarimeter = [];
41.
```

```

42. dt=0.05;
43.
44.     for cellule=1:length(pol_transient_durations(:,1))
45.         cell_duration = pol_transient_durations(cellule,2).*dt;
46.         transient_polarimeter(cellule,1:2) =
47. [max_SOP_changes(cellule,4),cell_duration];
48.     end
49.
50. % Maintenant que nous avons les matrices... nous pouvons effectuer
51. le compte des couples Max_ARR & Durations_ARR
52.
53. taille_mesure = length(transient_polarimeter(:,1));
54.
55. id_transient = 1;
56. id_duration = 1;
57.
58.     for cellule = 1:taille_mesure;
59.
60.         val_Max_ARR = transient_polarimeter(cellule,1);
61.
62.         for ind_ARR=1:(taille_valARR-1)
63.
64.             if (val_Max_ARR >= countMatrix_ARR(ind_ARR,1)...
65.                 && val_Max_ARR < countMatrix_ARR(ind_ARR+1,1));
66.                 %
67.                 sprintf('OK ARR')
68.                 break;
69.             end
70.         end
71.
72.         val_durations_ARR = transient_polarimeter(cellule,2);
73.
74.         for ind_duration=1:(taille_valDuree-1);
75.             %
76.             if (roundn(countMatrix_Duree(ind_duration,1),-2) ==
77. val_durations_ARR);
78.                 if val_durations_ARR >=
79. countMatrix_Duree(ind_duration,1)...
80.                     && val_durations_ARR <
81. countMatrix_Duree(ind_duration+1,1)
82.                     break;
83.                 end
84.             end
85.             %% Count the occurrences
86.             countMatrix_ARR_durations(ind_duration,ind_ARR) =
87. countMatrix_ARR_durations(ind_duration,ind_ARR)+1;
88.
89.             ind_ARR=1;
90.             ind_duration=1;
91.         end
92.         %% Computing Joint Probablity Values
93.         Sum_Prob = sum(sum(countMatrix_ARR_durations));
94.         matrix_prob_3D = countMatrix_ARR_durations/Sum_Prob;

```

ANNEXE XLVI

AFFICHAGE DE LA FONCTION DE DENSITÉ DE PROBABILITÉ CONJOINTE DES PARAMÈTRES DES EFFETS TRANSITOIRES DE POLARISATION

Script-A XLVI-1 Affichage de la fonction de densité de probabilité conjointe aux valeurs maximales de rotation angulaire et d'élasticité des effets transitoires de polarisation

```
1. %Copie de la matrice histogramme 3D
2. matrix_PDF = matrix_prob_3D;
3.
4. Mean_max_ARR = mean(transient_polarimeter(:,1));
5. Mean_ARR_deg = (180*Mean_max_ARR)/pi;
6. std_max_ARR = std(transient_polarimeter(:,1));
7. std_ARR_deg = (180*std_max_ARR)/pi;
8.
9.
10. Mean_deflection_ARR = mean(transient_polarimeter(:,2));
11. std_deflection_ARR = std(transient_polarimeter(:,2));
12.
13. N_value_ARR = length(transient_polarimeter(:,1));
14.
15. %% Displaying 3D Histogram
16.
17. figure1 = figure;
18.
19. h = bar3(matrix_PDF);
20. caxis([min(min(log(matrix_PDF))) 0])
21.
22. myColorMap = [0.862745106220245 0.862745106220245
23. 0.862745106220245;...
24. 0.878431379795074 0.878431379795074 0.878431379795074;...
25. 0.894117653369904 0.894117653369904
26. 0.894117653369904;0.909803926944733 0.909803926944733
27. 0.909803926944733;...
28. 0.925490200519562 0.925490200519562
29. 0.925490200519562;0.941176474094391 0.941176474094391
30. 0.941176474094391;...
31. 0.7843137383461 0.950980424880981
32. 0.950980424880981;0.627451002597809 0.960784316062927
33. 0.960784316062927;...
34. 0.470588237047195 0.970588207244873
35. 0.970588207244873;0.313725501298904 0.980392158031464
36. 0.980392158031464;...
37. 0.156862750649452 0.990196108818054 0.990196108818054;0 1
38. 1;0.0357142873108387 1 0.964285731315613;0.0714285746216774 1
39. 0.928571403026581;...
40. 0.107142858207226 1 0.892857134342194;0.142857149243355 1
41. 0.857142865657806;0.178571432828903 1 0.821428596973419;...
42. 0.214285716414452 1 0.785714268684387;0.25 1
43.
```

```

44. 0.75;0.28571429848671 1 0.714285731315613;0.321428567171097 1
45. 0.678571403026581;...
46. 0.357142865657806 1 0.642857134342194;0.392857134342194 1
47. 0.607142865657806;0.428571432828903 1
48. 0.571428596973419;0.46428570151329 1 0.535714268684387;...
49. 0.5 1 0.5;0.535714268684387 1 0.46428570151329;0.571428596973419
50. 1 0.428571432828903;...
51. 0.607142865657806 1 0.392857134342194;0.642857134342194 1
52. 0.357142865657806;0.678571403026581 1 0.321428567171097;...
53. 0.714285731315613 1 0.28571429848671;0.75 1
54. 0.25;0.785714268684387 1 0.214285716414452;0.821428596973419 1
55. 0.178571432828903;...
56. 0.857142865657806 1 0.142857149243355;0.892857134342194 1
57. 0.107142858207226;0.928571403026581 1 0.0714285746216774;...
58. 0.964285731315613 1 0.0357142873108387;1 1 0;1 0.944444417953491
59. 0;1 0.888888895511627 0;1 0.833333313465118 0;...
60. 1 0.777777791023254 0;1 0.722222208976746 0;1 0.666666686534882
61. 0;1 0.611111104488373 0;1 0.555555582046509 0;...
62. 1 0.5 0;1 0.444444447755814 0;1 0.388888895511627 0;1
63. 0.333333343267441 0;1 0.277777791023254 0;...
64. 1 0.22222223877907 0;1 0.16666667163372 0;1 0.11111111938953
65. 0;1 0.055555559694767 0;1 0 0;...
66. 0.974509835243225 0.0267973858863115 0;0.949019610881805
67. 0.0535947717726231 0;0.923529386520386 0.0803921595215797 0;...
68. 0.898039221763611 0.107189543545246 0;0.872549057006836
69. 0.133986935019493 0;0.847058832645416 0.160784319043159 0];
70.
71. colormap(myColorMap); % Apply the colormap
72.
73. for n=1:numel(h)
74.     cdata=get(h(n), 'zdata');
75.     cdata = log10(cdata);
76.     set(h(n), 'cdata', cdata, 'facecolor', 'interp')
77. end
78.
79. set(gca, 'ZScale', 'log');
80. set(gca, 'ZMinorTick', 'on');
81. set(gca, 'YDir', 'normal');
82. view(gca, [109.62744912545 22.6719118540499]);
83.
84.
85. % llim = 1.5e-04; %==> 05-11 Nov
86. % llim = 1.5e-04; %==> 25-30 Oct
87.
88. llim = 1/length(pol_transient_deflections(:,1));
89. h = get(gca, 'Children');
90.
91. for i = 1:length(h)
92.     ZData = get(h(i), 'ZData');
93.     ZData(ZData==0) = llim/3;
94.     set(h(i), 'ZData', ZData);
95. end
96.
97. % zdir = [0 0 1];
98. % rotate(h12, zdir, 90)

```

```

99. % view(0,90)
100. % ylim([0 10]);
101. % xlim([0.9 5]);
102. % figure(gcf);
103.
104. xlim([0 83])
105. ylim([1 113])
106. zlim([9.9999e-6 1e0])
107. set(gca,'XTick',[1 7 12 22 32 42 52 62 72 82]);
108. set(gca,'XTickLabel',{'0.9','1.5','2','3','4','5','6','7','8','9'},'
109. FontName','Calibri','fontsize',18);
110. set(gca,'YTick',[1 10 20 30 40 50 60 70 80 90 100 110 113]);
111. set(gca,'YTickLabel',{'0','0.125','0.250','0.375','0.5','0.625','0.7
112. 50','0.875','1','1.125','1.250','1.375','1.4'},'FontName','Calibri',
113. 'fontsize',15);
114. set(gca,'ZTick',[1e-5 1e-4 1e-3 1e-2 1e-1 1e0]);
115. % llim = 1.5e-04; %==> 05-11 Nov
116.
117. xlabel('Fluctuations maximales
118. {\itARR_m}(rad/ms)','FontName','Calibri','fontsize',20,'fontweight',
119. 'b');
120. ylabel('Amplitudes maximales {\it\phi_m}
121. (rad)','FontName','Calibri','fontsize',20,'fontweight','b');
122. zlabel('Fonction de probabilité
123. {\itF}_{ARR_m}_{\phi_m}','FontName','Calibri','fontsize',18,'f
124. ontweight','b');

```

Script-A XLVI-2 Affichage de la fonction de densité de probabilité conjointe aux valeurs maximales de rotation angulaire et aux valeurs de durée

```

1.
2. %% Copie de l'histogramme 3D
3. matrix_PDF = matrix_prob_3D;
4.
5.
6. Mean_max_ARR = mean(transient_polarimeter(:,1));
7. Mean_ARR_deg = (180*Mean_max_ARR)/pi;
8. std_max_ARR = std(transient_polarimeter(:,1));
9. std_ARR_deg = (180*std_max_ARR)/pi;
10.
11. Mean_duree_ARR = mean(transient_polarimeter(:,2));
12. std_duree_ARR = std(transient_polarimeter(:,2));
13.
14.
15. N_value_ARR = length(transient_polarimeter(:,1));
16.
17. figure1 = figure;
18.
19. h = bar3(matrix_PDF);
20.
21. myColorMap = ([1 1 1;0.9333333337306976 1 1;0.8666666674613953 1 1;...
22. 0.800000011920929 1 1;0.733333349227905 1 1;0.6666666686534882 1
23.

```

```

24. 1;...
25. 0.600000023841858 1 1;0.533333361148834 1 1;0.466666668653488 1
26. 1;...
27. 0.400000005960464 1 1;0.333333343267441 1 1;0.266666680574417 1
28. 1;...
29. 0.200000002980232 1 1;0.133333340287209 1 1;0.0666666701436043 1
30. 1;...
31. 0 1 1;0.0416666679084301 1 0.958333313465118;...
32. 0.0833333358168602 1 0.916666686534882;0.125 1
33. 0.875;0.16666667163372 1 0.833333313465118;0.20833332836628 1
34. 0.791666686534882;...
35. 0.25 1 0.75;0.291666656732559 1
36. 0.708333313465118;0.333333343267441 1 0.666666686534882;0.375 1
37. 0.625;0.416666656732559 1 0.583333313465118;...
38. 0.458333343267441 1 0.541666686534882;0.5 1
39. 0.5;0.541666686534882 1 0.458333343267441;0.583333313465118 1
40. 0.416666656732559;0.625 1 0.375;...
41. 0.666666686534882 1 0.333333343267441;0.708333313465118 1
42. 0.291666656732559;0.75 1 0.25;0.791666686534882 1
43. 0.20833332836628;...
44. 0.833333313465118 1 0.16666667163372;0.875 1
45. 0.125;0.916666686534882 1 0.0833333358168602;0.958333313465118 1
46. 0.0416666679084301;...
47. 1 1 0;1 0.944444417953491 0;1 0.888888895511627 0;1
48. 0.833333313465118 0;1 0.777777791023254 0;1 0.722222208976746 0;1
49. 0.666666686534882 0;...
50. 1 0.611111104488373 0;1 0.555555582046509 0;1 0.5 0;1
51. 0.444444447755814 0;1 0.388888895511627 0;1 0.333333343267441 0;...
52. 1 0.277777791023254 0;1 0.22222223877907 0;1 0.16666667163372
53. 0;1 0.11111111938953 0;1 0.055555559694767 0;1 0 0;...
54. 0.974509835243225 0.0267973858863115 0;0.949019610881805
55. 0.0535947717726231 0;0.923529386520386 0.0803921595215797 0;...
56. 0.898039221763611 0.107189543545246 0;0.872549057006836
57. 0.133986935019493 0;0.847058832645416 0.160784319043159 0]);
58.
59. colormap(myColorMap); % Apply the colormap
60.
61. for n=1:numel(h)
62.
63.     cdata=get(h(n), 'zdata');
64.     cdata = log10(cdata);
65.     set(h(n), 'cdata', cdata, 'facecolor', 'interp')
66.
67. end
68.
69.
70. set(gca, 'ZScale', 'log');
71. % set(gcf, 'Position', [0.13 0.130753138075314 0.869479166666667
72. 0.778242677824267]);
73. set(gca, 'ZMinorTick', 'on');
74. set(gca, 'YDir', 'normal');
75. view(gca, [109.62744912545 22.6719118540499]);
76.
77. llim = 1/length(pol_transient_durations(:,4)); %==> 25-30 Oct
78. h = get(gca, 'Children');

```

```

79.
80. for i = 1:length(h)
81.   ZData = get(h(i), 'ZData');
82.   ZData(ZData==0) = llim/3;
83.   set(h(i), 'ZData', ZData);
84. end
85.
86. % grid(axes1,'on');
87. % hold(axes1,'all');
88. xlim([0 83])
89. ylim([0 200])
90. zlim([9.9999e-6 1e0])
91. set(gca,'XTick',[1 7 12 22 32 42 52 62 72 82]);
92. set(gca,'XTickLabel',{'0.9','1.5','2','3','4','5','6','7','8','9'},'
93.   FontName','Calibri','fontsize',18);
94. set(gca,'YTick',[0 10 20 30 40 50 60 70 80 90 100 110 120 130 140
95.   150 160 170 180 190 200]);
96. set(gca,'YTickLabel',{'0','0.5','1','1.5','2','2.5','3','3.5','4','4
97.   .5',...
98.   '5','5.5','6','6.5','7','7.5','8','8.5','9','9.5','10'},...
99.   'FontName','Calibri','fontsize',15);
100.
101. set(gca,'ZTick',[1e-5 1e-4 1e-3 1e-2 1e-1 1e0]);
102. % llim = 1.5e-04; %==> 05-11 Nov
103.
104.
105. xlabel('Fluctuations maximales {\itARR_m}
106.   (rad/ms)','FontName','Calibri','fontsize',20,'fontweight','b');
107. ylabel('Durées {\Delta}{\tau}
108.   (ms)','FontName','Calibri','fontsize',20,'fontweight','b');
109. zlabel('Fonction de probabilité
110.   {\itF}_{ARR_m}_{\Delta}_{\tau}','FontName','Calibri','fontsize
111.   ',18,'fontweight','b');
112. % zdir = [0 0 1];
113. % rotate(h12,zdir,90)
114. % view(0,90)
115. % ylim([0 10]);
116. % xlim([0.9 5]);
117. % figure(gcf);
118.
119. %% CONTOUR PROCESSING
120.
121. llim = 1/length(SOP_durations(:,4));
122.
123. for i=1:200
124.   for j=1:83
125.     if matrix_PDF(i,j) == 0
126.       matrix_PDF(i,j) = llim/3;
127.     end
128.   end
129. end
130.
131. matrix_PDF_tr = transpose(matrix_PDF);
132.
133.

```

```
134. figure3 = figure;
135.
136. logzplot('surf',matrix_PDF,'colorbar');
137.
138. xlim([1 83])
139. ylim([1 200])
140. set(gca,'XTick',[1 7 12 22 32 42 52 62 72 82]);
141. set(gca,'XTickLabel',{'0.9','1.5','2','3','4','5','6','7','8','9'},'
142. FontName','Calibri','fontsize',18);
143. set(gca,'YTick',[1 10 20 30 40 50 60 70 80 90 100 110 120 130 140
144. 150 160 170 180 190 200]);
145. set(gca,'YTickLabel',{'0.05','0.5','1','1.5','2','2.5','3','3.5','4'
146. '4.5','5','5.5','6','6.5','7','7.5','8','8.5','9','9.5','10'},'Font
147. Name','Calibri','fontsize',15);
148. xlabel('Maximum rotation rate
149. (rad/ms)','FontName','Calibri','fontsize',20,'fontweight','b');
150. ylabel('Transient duration
151. (ms)','FontName','Calibri','fontsize',20,'fontweight','b');
```

ANNEXE XLVII

CALCUL, AJUSTEMENT ET AFFICHAGE DE LA FONCTION DE DENSITÉ DE PROBABILITÉ DES INTERVALLES DE TEMPS ENTRE DEUX EFFETS TRANSITOIRES DE POLARISATION ADJACENTS

Script-A XLVII-1 Calcul, ajustement et affichage des valeurs d'intervalle de temps entre deux effets transitoires de polarisation par la fonction de densité exponentielle

```
1.  %% matrice des occurrences Time intervals
2.  countMatrix_Time_Intervals= zeros(1001,2);
3.
4.  taille_val_Time_interval = length(countMatrix_Time_Intervals(:,1));
5.  taille_mesure_Time_interval = length(Time_intervals(:,1));
6.
7.  cellule = 1;
8.  ValIntervalScan = 0.1;
9.
10.
11.  for cellule = 1:taille_val_Time_interval;
12.      countMatrix_Time_Intervals(cellule,1)= ValIntervalScan;
13.      ValIntervalScan = ValIntervalScan + 0.01;
14.
15.  end
16.
17.  cellule = 1;
18.
19.  for cellule = 1:taille_mesure_Time_interval,
20.
21.      ValIntervalScan = Time_intervals(cellule,3);
22.      for i=1:taille_val_Time_interval;
23.
24.          %% On compte les itérations maintenant !!
25.          if (ValIntervalScan >= countMatrix_Time_Intervals(i,1) &&
26.              ValIntervalScan < countMatrix_Time_Intervals(i+1,1));
27.              countMatrix_Time_Intervals(i,2)=
28.              countMatrix_Time_Intervals(i,2)+1;
29.          end
30.      end
31.      i=0;
32.  end
33.
34.
35.  %% Computing the sum of occurrences
36.  Sum_Prob_Time_Interval = sum(countMatrix_Time_Intervals(:,2));
37.
38.
39.  %% Computing the probability values
40.  countMatrix_Time_Intervals(:,3) =
41.  countMatrix_Time_Intervals(:,2)/Sum_Prob_Time_Interval;
42.
```

```

43.  %% Quatrième colonne ==> Normalisation par la méthode des trapèzes
44.  countMatrix_Time_Intervals(:,4) =
45.  countMatrix_Time_Intervals(:,3)*(1/trapz(countMatrix_Time_Intervals(
46.  :,1),countMatrix_Time_Intervals(:,3)));
47.
48.
49.  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Fitting by Exponential density function
50.
51.  %Pondération linéaire des valeurs de PDF
52.
53.
54.  id_pos = find(countMatrix_Time_Intervals(:,2)~=0);
55.
56.  N = sum(countMatrix_Time_Intervals(:,2));
57.  p = countMatrix_Time_Intervals(id_pos,3);
58.  H = countMatrix_Time_Intervals(id_pos,2);
59.
60.  sigma2_lin = p./N;
61.  sigma_lin = sqrt(sigma2_lin);
62.  weights_lin = 1./sigma2_lin;
63.  mean_weight_lin = mean(weights_lin);
64.  W_linear_relative = weights_lin / mean_weight_lin;
65.
66.
67.  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% EXPONENTIAL DISTRIBUTION   VERSION 1.0
68.  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
69.
70.
71.  modelFun_Exp = @(lambda,x) lambda*exp(-lambda*x);
72.
73.  optionsLSQ = optimset('Display','Iter','TolFun',1e-250,'TolX',1e-
74.  250, 'MaxFunEvals',5000000,'MaxIter',5000000);
75.
76.  lambda_init = 1/mean(Time_intervals(:,3));
77.  % lambda_init = 0.1;
78.
79.  % First Options - Weight Exp Fit
80.  x_pos = countMatrix_Time_Intervals(id_pos,1); % Abscisse des
81.  intervalles de temps
82.  y_pos = countMatrix_Time_Intervals(id_pos,4); % Normalisation des
83.  valeurs / densité
84.
85.  [lambda,Residual_exp,Jacob_exp,Cov_exp, mse_exp] = nlinfit(x_pos,
86.  y_pos,
87.  modelFun_Exp,lambda_init,'Weights',W_linear_relative,'options',optio
88.  nsLSQ); %
89.
90.
91.  Jacobian_Exp = full(Jacob_exp);
92.  CI_lambda =
93.  nlparci(lambda,Residual_exp,'jacobian',Jacobian_Exp,'alpha',0.05);
94.
95.  lambda_low = CI_lambda(1);
96.  lambda_high = CI_lambda(2);
97.

```

```

98. lambda_final = lambda;
99.
100. xdata = 0:0.01:10;
101. xdata = transpose(xdata);
102.
103. fit_exp = lambda_final*exp(-lambda_final.*xdata);
104.
105. fit_exp_low = lambda_low*exp(-lambda_low.*xdata);
106.
107.
108. fit_exp_high = lambda_high*exp(-lambda_high.*xdata);
109.
110. % yfit_exp1 = lambda_final_v1*exp(-lambda_final_v1.*xdata_high);
111. % yfit_exp1(:,2) = yfit_exp1(:,1)/100;
112. % yfit_exp1(:,3) = yfit_exp1(:,2)*coeff_prob;
113. numerator_exp = sum((countMatrix_Time_Intervals(:,4) -
114. mean(countMatrix_Time_Intervals(:,4))).*(fit_exp(:,1) -
115. mean(fit_exp(:,1))));
116. denominator_exp = sqrt(sum((countMatrix_Time_Intervals(:,4) -
117. mean(countMatrix_Time_Intervals(:,4)).^2)).*sqrt(sum((fit_exp(:,1)
118. - mean(fit_exp(:,1)).^2)));
119. Pearson_coeff_exp = (numerator_exp/denominator_exp);
120.
121. %% Cration de la figure bar
122.
123. Mean_time_interval = mean(Time_intervals(:,1));
124. std_time_interval = std(Time_intervals(:,1));
125.
126.
127. N_value_time_intervals= size(countMatrix_Time_Intervals(:,1));
128.
129. figure1 = figure;
130. % 'YMinorTick','on',...
131. % 'YScale','log','YMinorTick','on',...
132. %% Histogramme normalisé PHI !
133. axes1 = axes('Parent',figure1,...
134. 'YScale','log','YMinorTick','on',...
135. 'Position',[0.0609375 0.0893939393939394 0.92890625
136. 0.886363636363636],...
137. 'FontSize',16,...
138. 'FontName','Calibri');
139. box(axes1,'on');
140. hold(axes1,'all');
141.
142. min_time_interval = min(countMatrix_Time_Intervals(id_pos,4));
143. max_time_interval = max(countMatrix_Time_Intervals(id_pos,4));
144.
145. h1 =
146. bar(countMatrix_Time_Intervals(1:N_value_time_intervals,1),countMatr
147. ix_Time_Intervals(1:N_value_time_intervals,4),'b','BaseValue',min_ti
148. me_interval/1.2,'Parent',axes1);
149. hold on;
150.
151. h2 = plot(xdata(:,1), fit_exp(:,1));
152. set(h2,'color','r');

```

```
153. set(h2, 'linewidth', 2);
154. set(h2, 'marker', 'none');
155. hold on
156.
157. h3 = plot(xdata(:,1), fit_exp_low(:,1), '--');
158. set(h3, 'color', 'r');
159. set(h3, 'linewidth', 2);
160. set(h3, 'marker', 'none');
161. hold on
162.
163. h4 = plot(xdata(:,1), fit_exp_high(:,1), '--');
164. set(h4, 'color', 'r');
165. set(h4, 'linewidth', 2);
166. set(h4, 'marker', 'none');
167. hold on
168.
169. xlabel('Intervalles de temps {\Delta}T
170. (s)', 'FontName', 'Calibri', 'fontsize', 20, 'fontweight', 'b');
171. ylabel('PDF', 'FontName', 'Calibri', 'fontsize', 20, 'fontweight', 'b');
172. ylim([min_time_interval/1.2 max_time_interval*1.5]);
173. xlim([0 10]);
174. set(gca, 'XTick', (0:0.5:10));
175. grid on;
176. figure(gcf);
177. hold off;
178.
```

BIBLIOGRAPHIE

- Agilent Technologies. 2013. « Agilent N7786B Polarization Synthesizer ». Fiche technique, document PDF, USA. En ligne. 5 p. < <http://cp.literature.agilent.com/litweb/pdf/5989-8115EN.pdf> >. Consulté le 15 juillet 2014.
- Agilent Technologies. 2013. « Agilent N7781B Polarization Analyzer ». Fiche technique, document PDF, USA. En ligne. p.5.< <http://cp.literature.agilent.com/litweb/pdf/5989-8115EN.pdf> >. Consulté le 15 juillet 2014.
- Agilent Technologies. 2008. « Measuring Polarization Dependent Loss of Passive Optical Components ». Note d'application, document PDF, USA. En ligne. 16 p. < <http://cp.literature.agilent.com/litweb/pdf/5990-3281EN.pdf> >. Consulté le 20 juillet 2014.
- Bao, Xiaoyi, D.S. Waddy, L. Chen. 2004. « Polarization fluctuations in field fibers ». *Lasers and Electro-Optics Society, 2004 : the 17th Annual Meeting of the IEEE (LEOS 2004)*. (Rio Grande, Porto Rico, Nov. 7-11 2007), vol.1, p.82-83. Rio Grande : *IEEE*.
- Björck, Ake. 1996. *Numerical Methods for Least Squares Problems*, 1st ed. Philadelphia: *Society for Industrial and Applied Mathematics*, 408 p.
- Cantrell, C.D. 2004. *Polarization*. [Notes du cours EEOP 6334 Advanced geometrical and physical optics]. Richardson, Texas: PhoTEC, Erik Jonsson School, Université de Texas at Dallas.
- Chan, Calvin C.K. 2010. *Optical Performance Monitoring : advanced techniques for next-generation photonic networks*. Burlington, Massachusetts: *Academic Press/Elsevier*, 459 p.
- Crawford, T, A. Mortelette, C. Amelin, H. Dayfallah, M.A. Reimer, K. Mizerek, D. Peterson, T. J. Xia, C. Tremblay, D. Doucet, S. Duvernay, S. Wellbrock, M. O'Sullivan et M. P. Belanger. 2013. « Long term SOP evolution measurements and modeling», *paper draft v.03* for submission à *IEEE Electronics Letter*.
- Crawford, T., M.A. Reimer, H. Dayfallah, K. Mizerek, D. Doucet, M. O'Sullivan et M. P. Belanger. 2013. Communications privées. *Ciena Corporation*.
- Gavin, H. P. 2013. « The Levenberg-Marquardt method for nonlinear least squares curve-fitting problems ». Duke University, Michigan (USA). Note de Lecture PDF. En ligne. 17p. < <http://people.duke.edu/~hpgavin/ce281/lm.pdf> >. Consulté le 15 juillet 2014.

- Gisin, N. 1995. « Statistics of Polarization Dependent Losses ». *Optics Communications*, vol.114, p.399-405.
- Grinstead, C.M., et J.L. Snell. (1997). *Introduction to probability*, Providence (RI): American Mathematical Society, 510 p.
- Hui, Rongqing et Maurice O'Sullivan. 2006. *Fiber Optic Measurement Techniques*, 1st édition. Burlington (MA): Elsevier Academic Press. 672 p.
- Hui, Zhang, Guiliang Zhao et D. Wang. 1991. « Super-Gaussian window function and its applications ». In *International Conference on Circuits and Systems: Conference Proceeding0 1991*. (Shenzhen, Jun. 16-17 1991), vol.2, p.595-598. Shenzhen (China): *IEEE*.
- Kawasaki, T., W. Ichihara, T. Kataoka, et S. Matsuoka. 2007. « Over 5-months Long-Term PMD Continuous Measurement in Installed Fiber Cables with an Exposed Fiber Section ». In *Optical Fiber Communication and the National Fiber Optic Engineers Conference (OFC/NFOEC 2007)*. (Anaheim, CA, 25-29 March 2007), p.1-3. Anaheim (CA): *Optical Society of America*.
- Kreyszig, Edward, Herbert Kreyszig et Edward J. Norminton. 2011. *Advanced engineering mathematics*, 10th edition. Hoboken (N.J): Wiley, 1280 p.
- Krishnamoorthy, Kalimuthu. 2006. *Handbook of statistical distributions with applications (Statistics, a series of textbooks & monographs)*. Boca Raton: Chapman & Hall/CRC, 346 p.
- Levenberg, K. 1944. « A Method for the Solution of Certain Problems in Least Squares ». *Quarterly Journal of Applied Mathematics*, vol. 2, p.164-168.
- Liétaert, Grégory. 2005. « La PMD dans la fibre : Réalité ou pure spéculation? ». Document de présentation, PDF, *Opto 2005 - Conférence Club scientifique, JDSU*, France. En ligne. p.31. < https://www.renater.fr/IMG/pdf/Acterna_28-09.pdf >. Consulté le 15 juillet 2014.
- Marquardt, D. 1963. « An Algorithm for Least-Squares Estimation of Nonlinear Parameters ». *Journal of the Society for Industrial and Applied Mathematics*, vol. 11, n°2, p.431-441.
- Matsuda, T., T. Kawasaki, T. Kotanigawa, A. Naka et K. Oda. 2009. « Field trial of 43-Gbit/s RZ-DQPSK transmission in aerial fiber with rapidly changing SOP ». In *Optical Fiber Communication (OFC 2009) - includes post deadline papers*. (San Diego, Mar. 22-26 2009), p. 1-3. San Diego (CA): *IEEE*.

- Moler, Cleve. 2004. « *Chapter 5 : Least Squares* ». In *Numerical Computing with MATLAB* », electronic edition. Philadelphia: *Society for Industrial and Applied Mathematics*. En ligne. 27 p. < <http://www.mathworks.com/moler/leastsquares.pdf> >.
- Moon, T. K. et W. C. Stirling. 1999. *Mathematical Methods and Algorithms For Signal Processing*. N.J. (USA): Pearson Prentice Hall, 937 p.
- Navidi, William. 2010. *Statistics for Engineers*, 3rd edition. New York (USA): McGraw-Hill, 933 p.
- Nelson, Lynn E. 2012. « Polarization effects in coherent systems ». In *Optical Fiber Communication Conference and Exposition and the National Fiber Optic Engineers Conference (OFC/NFOEC 2012)*. (Los Angeles, Mar. 4-8 2012), p. 1-3. Los Angeles, (CA) : *IEEE*.
- Nelson, L. E., S. L. Woodward, S. Foo, X. Zhou, M.D. Feuer, D. Hanson, D. McGhan, H. Sun, M. Moyer, M. O'Sullivan et P.D. Magill. 2009. « Performance of a 46-Gbps Dual-Polarization QPSK Transceiver With Real-Time Coherent Equalization Over High PMD Fiber ». *Journal of Lightwave Technology*, vol. 27, n°3, p. 158-167.
- Noutsios, P.C. 2007. « In-service Measurements of Polarization Fluctuations on Field-installed OC-192 DWDM Systems ». In *International Symposium on Signals, Systems and Electronics (ISSSE '07)*. (Montreal, July 30 2007-Aug. 2 2007), p. 323-326. Montreal, (QC): *IEEE*.
- O'Sullivan, Maurice. 2011. « Implementation of multi-carrier transmission systems with coherent detection ». In *European Conference and Exposition on Optical Communications (ECOC 2011)*. (Geneva Switzerland Sept. 18-22, 2011). *Ciena corporation*, 39 p.
- O'Sullivan, Maurice. 2012. Communications privées, *Ciena Corporation*.
- Peterson, Daniel L Jr. 2004. « Estimation of outage statistics due to polarization mode dispersion ». Thèse de doctorat en génie physique, Dallas, Université du Texas, 109 p.
- Peterson, D.L., P.J. Leo et K.B. Rochford. 2004. « Field measurements of state of polarization and PMD from a tier-1 carrier ». In *Optical Fiber Communication Conference 2004 (OFC 2004)*. (Los Angeles, CA, Feb. 23-27 2004), vol.2. p. 1-4. Los Angeles (CA): *IEEE*.
- Reimer, Michael. 2012-2013. Communications privées, *Ciena Corporation*.

- Reimer, Michael A. 2007. « Modeling and Simulation of Polarization Mode Dispersion and Polarization Dependent Loss ». Mémoire de maîtrise en génie physique, Ottawa, Université d'Ottawa, 162 p.
- Roberts, K., M. O'Sullivan, W. Kuang-Tsan, S. Han, A. Awadalla, D.J. Krause et C. Laperle. 2009. « Performance of Dual-Polarization QPSK for Optical Transport Systems » *Journal of Lightwave Technology*, vol.27, n°16, p. 3546-3559.
- Smith, S. K. 1999. *The Scientist and Engineer's Guide to Digital Signal Processing*. Second edition. San Diego (CA): California Technical Publishing, 688 p.
- Tokmakoff, Andrei. 2011. *Time-correlation functions*. [Notes de cours : *Introductory Quantum Mechanics II*]. Cambridge, Massachusetts: MIT Department of Chemistry. En ligne. p.72-80 < <http://ocw.mit.edu/courses/chemistry/5-74-introductory-quantum-mechanics-ii-spring-2004/lecture-notes/11.pdf> >. Consulté le 15 juillet 2014.
- Tremblay, Christine, Andrew Cassidy, Aurélien Mortelette, Cyril Amelin, Marc Lyonnais, Thomas Tam et Michel P. Bélanger. 2012. « Advanced performance measurements using 40G coherent systems ». In *Asia Communication and Photonics Conference (ACP2012)*. (Guangzhou (Chine), Nov. 7-10 2012), p. 323-326. Guangzhou (Chine) : *IEEE*.
- Waddy, David S., Ping Lu, Liang Chen et Xiaoyi Bao. 2001. « Fast state of polarization changes in aerial fiber under different climatic conditions ». *Photonics Technology Letters, IEEE*, vol.13, n°9, p.1035-1037.
- Waddy, David D., Liang Chen et Xiaoyi Bao. 2005. « Fast PMD and PDL measurement of aerial fiber ». In *Optical Fiber Communication Conference and Exposition and the National Fiber Optic Engineers Conference (OFC/NFOEC 2005)*. (Mar. 6-11 2005), vol.3 p. 1-3.
- Walck, C. 2007. *Hand-book on STATISTICAL DISTRIBUTIONS for experimentalists*. Stockholm: University of Stockholm, 202 p.
- Walpole, Ronald E., Raymond H. Myers, Sharon L. Myers et Ye E. Keying. 2007. *Probability & Statistics for Engineers & Scientists*, 8th edition. Upper Saddle River (N. J.): Pearson Prentice Hall, Pearson Education.823 p.
- Wolberg, John. 2005. *Data Analysis Using the Method of Least Squares: Extracting the Most Information from Experiments*, 1st ed. New York: Springer. 250 p.

- Woodward, S.L., L.E. Nelson, C.R. Schneider, L.A. Knox, AT&T, M. O'Sullivan, C. Laperle, M. Moyer et S. Foo, Ciena Corporation. 2012. « Field Observations of Optical Parameters using Integrated Performance Monitoring ». In *Optical Fiber Communication Conference and Exposition and the National Fiber Optic Engineers Conference (OFC/NFOEC 2012)*. (Los Angeles, Mar. 4-8 2012), p. 1-3. Los Angeles (CA) : *IEEE*.
- Woodward, S.L.; L.E. Nelson, C.R. Schneider, L.A. Knox, M. O'Sullivan, C. Laperle, M. Moyer et S. Foo. 2014. « Long-Term Observation of PMD and SOP on Installed Fiber Routes ». *Photonics Technology Letters, IEEE* , vol.26, no.3, p.213-216.
- Xia, T.J., G. Wellbrock, M. Pollock, W. Lee, D. Peterson, D. Doucet, J. Sitch, K. Ghazian, P. Bryan et P. Rochon. 2009. « 92-Gb/s field trial with ultra-high PMD tolerance of 107-ps DGD ». In *Optical Fiber Communication Conference and Exposition and the National Fiber Optic Engineers Conference (OFC/NFOEC 2009) includes post deadline papers*. (San Diego, Mar. 22-26 2009), p. 1-3. San Diego (CA): *IEEE*.
- Xu, Li. 2010. « All-order polarization mode dispersion sensing and compensation in optical fiber communications ». Thèse de doctorat en génie physique, Lafayette Ouest (Inde), Université du Purdue, 110 p.
- Yan, L. -S, X.S. Yao, C. Yu, Y. Wang, L. Lin, Z. Chen et A.E. Willner. 2006. «High-speed and highly repeatable polarization-state analyzer for 40-Gb/s system performance monitoring ». *Photonics Technology Letters, IEEE*, vol.18, n°4, p.643-645.
- Zhang, Xu. 2012. « Digital Signal Processing for Optical Coherent Communication Systems ». Thèse de doctorat en génie physique, Kongens Lyngby, Université Technique du Danemark, 149 p.