

Titre: Title:	Extracteur de paramètres Delta-Lognormaux globalement optimaux par séparation et évaluation exploitant l'arithmétique par intervalles
Auteur: Author:	Simon Pierre Boyogueno Bidias
Date:	2022
Type:	Mémoire ou thèse / Dissertation or Thesis
Référence: Citation:	Boyogueno Bidias, S. P. (2022). Extracteur de paramètres Delta-Lognormaux globalement optimaux par séparation et évaluation exploitant l'arithmétique par intervalles [Thèse de doctorat, Polytechnique Montréal]. PolyPublie. https://publications.polymtl.ca/10749/

Document en libre accès dans PolyPublie

Open Access document in PolyPublie

URL de PolyPublie: <https://publications.polymtl.ca/10749/>
PolyPublie URL:

Directeurs de recherche: Réjean Plamondon, Jean Pierre David, & Yvon Savaria
Advisors:

Programme: Génie électrique
Program:

POLYTECHNIQUE MONTRÉAL

affiliée à l'Université de Montréal

**Extracteur de paramètres Delta-Lognormaux globalement optimaux par
séparation et évaluation exploitant l'arithmétique par intervalles**

SIMON PIERRE BOYOGUENO BIDIAS

Département de génie électrique

Thèse présentée en vue de l'obtention du diplôme de *Philosophiæ Doctor*

Génie électrique

Août 2022

POLYTECHNIQUE MONTRÉAL

affiliée à l'Université de Montréal

Cette thèse intitulée :

**Extracteur de paramètres Delta-Lognormaux globalement optimaux par
séparation et évaluation exploitant l'arithmétique par intervalles**

présentée par **Simon Pierre BOYOGUENO BIDIAS**

En vue de l'obtention du diplôme de *Philosophiæ Doctor*

a été dûment acceptée par le jury d'examen constitué de :

Frédéric LESAGE, président

Réjean PLAMONDON, membre et directeur de recherche

Yvon SAVARIA, membre et codirecteur de recherche

Jean Pierre DAVID, membre et codirecteur de recherche

Marco BONIZZATO, membre

Marc PARIZEAU, membre externe

REMERCIEMENTS

Je remercie Mon Papa, Monsieur Boyogueno Victor pour son support et ses encouragements dans toutes les étapes de ma scolarisation, particulièrement dans ma décision de poursuivre les études doctorales. Bien que n'étant plus de ce monde tu demeures incontestablement ma source de motivation et d'encouragement pour l'accomplissement de cette thèse. Je remercie mes directeurs de thèse les professeurs Plamondon Réjean, Savaria Yvon, David Jean-Pierre pour leur support financier et expérience dans la recherche. Je remercie aussi tous les collègues du laboratoire Scribens de Polytechnique Montréal avec qui j'ai eu à échanger pendant mon séjour dans ce laboratoire. Je ne s'aurai oublier dans cette liste mon petit frère Boyogueno Moudio Michel à qui je dis merci pour son support technique et moral.

RÉSUMÉ

Le présent projet s'inscrit dans la continuité d'un ensemble de travaux visant à concevoir un outil matériel peu couteux pouvant permettre l'analyse des mouvements manuscrits humains dans le cadre d'applications biomédicales. En effet, la Théorie Cinématique du Mouvement Humain Rapide semble apporter des éléments de réponse sur le comportement temporel du système moteur d'un être humain lors de l'accomplissement d'un mouvement. Le modèle Delta-Lognormal qui dérive de cette théorie présente sous forme d'une équation mathématique, l'interaction agoniste-antagoniste ainsi que l'ensemble des propriétés temporelles de toutes les composantes neuromusculaires étant intervenues dans l'accomplissement d'un mouvement. Les paramètres Delta-Lognormaux issus de ce modèle font l'objet de nombreuses études dans le domaine des sciences, particulièrement en génie biomédical, où ils visent à caractériser la susceptibilité d'un individu à développer certaines maladies.

Le sujet de cette thèse est la conception d'un algorithme robuste permettant d'extraire des paramètres Delta-Lognormaux globalement optimaux. L'algorithme conçu cherche un optimum global qui approche la solution exacte, dans un délai raisonnable avec une garantie de qualité afin de permettre l'étude des mouvements dans le domaine biomédical.

Le développement de cet outil d'extraction des paramètres Delta-Lognormaux à partir des mouvements de pointage provenant de gestes neuromoteurs repose sur un algorithme d'optimisation par séparation et évaluation (« Branch-and-Bound »), (**S&E**) utilisant le concept d'arithmétique par intervalles (**AI**).

L'algorithme **S&E** conçu est décrit sous forme de contributions scientifiques à l'état de l'art sur l'extraction des paramètres Delta-Lognormaux. Les quatre Publications principales sont les suivantes:

- « **On the use of Interval Arithmetic to Bound Delta-Lognormal Rapid Human Movements Models** »
- « **On the Use of Interval Arithmetic for the Branch and Bound Delta-Lognormal Parameter Extraction of Rapid Human Movements** »

- « **Combining Interval Arithmetic with the Branch and Bound Algorithm for Delta-lognormal Parameter Extraction** »
- « **A Global Optimization Algorithm for the Estimation of Delta-Lognormal Parameters Characterizing Pointing Gestures** »

Ces publications relatent de façon détaillées l'ensemble des contributions principales développées dans cette thèse.

Deux Publications secondaires ont aussi été réalisées :

- « **An Interactive Tablet-Based System to Run Neuromuscular Tests** »
- « **A New Experimental Set-Up To Run Neuromuscular Test** »

Ces publications relatent les contributions complémentaires tels que le processus de collecte et d'acquisition de données de mouvements simples permettant de tester l'algorithme.

Dans le cas des contributions principales, Les publications « **On the use of Interval Arithmetic to Bound Delta-Lognormal Rapid Human Movements Models** » et « **Combining Interval Arithmetic with the Branch and Bound Algorithm for Delta-lognormal Parameter Extraction** » traitent de l'aspect séparation (**Branch**) de l'algorithme. On y retrouve l'ensemble des techniques relatives à la conception d'un espace de recherche dans lequel s'opérera l'exploration des solutions. Le chapitre « **On the Use of Interval Arithmetic for the Branch and Bound Delta-Lognormal Parameter Extraction of Rapid Human Movements** » dans le livre « *The Lognormality Principle and its applications in E-Security, E-learning and E-Health* » et la publication « **A Global Optimization Algorithm for the Estimation of Delta-Lognormal Parameters Characterizing Pointing Gestures** » traitent de l'évaluation (**Bound**). Dans ces deux publications, différentes contributions pour chaque étape de l'évaluation sont proposées et insérées dans la partie séparation pour former l'extracteur des paramètres Delta-Lognormaux.

L'arithmétique par intervalles est utilisée dans cette thèse comme outil mathématique central pour les différents calculs dans les quatre publications principales. Une bibliothèque non publiée contenant toutes les fonctions arithmétiques et transcendantales de bases se trouvant dans les fonctions delta et Sigma Lognormale a été entièrement développée en langage de programmation évolué (C++).

L'algorithme utilise l'équation Delta-Lognormale reconnue comme la meilleure pour la reproduction des mouvements de pointage rapides, afin d'identifier les meilleurs paramètres pour les décrire et de permettre leur utilisation dans le domaine biomédical.

Au-delà des applications médicales, la possibilité d'extraire des solutions globalement optimales, selon un degré d'optimalité à préciser, ouvre de nouvelles perspectives, à savoir :

- La réalisation de nouvelles études fondamentales sur les propriétés des modèles Lognormaux appliqués à l'analyse des mouvements humains.
- La caractérisation des outils d'extraction Delta-Lognormaux.
- L'utilisation de la modélisation Delta-Lognormale dans des applications où une solution localement optimale n'est pas suffisante.

En plus des contributions principales et secondaires, cette thèse comporte des textes inédits qui détaillent les développements d'intérêts généraux associés à la modélisation lognormale, et à l'élaboration des banques de données de mouvements manuscrits rapides (synthétiques et réels) pour tester l'extracteur conçu.

ABSTRACT

This thesis aims at designing a hardware tool that can analyze handwritten movements performed by patients in clinics. Indeed, the Kinematic Theory of Rapid Human movement provides an accurate model of the temporal behaviour of the human motor system. The Delta-Lognormal model that derives from this theory provides a mathematical equation reflecting the antagonist-agonist interaction as well as the temporal properties of all the neuromuscular components involved in the accomplishment of a movement. The Delta-lognormal parameters obtained from the model was used in numerous studies in science, particularly in medicine, where they are used to characterize people susceptibility to develop some diseases.

The project aims to design a robust algorithm to extract globally optimal parameters for the Delta-Lognormal model. The algorithm must be of global nature and guarantee an exact solution within a reasonable processing time to allow studies of movements in the biomedical field.

The proposed Delta-Lognormal parameters extractor tool analyzes pointing movements coming from captured neuromotor gestures. This tool is based on an optimization algorithm that exploits the concept of Separation and Evaluation (“Branch-and-Bound”) using interval arithmetic. The proposed extractor is described in this thesis in the form of scientific contributions. The following four main publications :

- **« On the use of Interval Arithmetic to Bound Delta- Lognormal Rapid Human Movements Models »**
- **« On the Use of Interval Arithmetic for the Branch and Bound Delta- Lognormal Parameter Extraction of Rapid Human Movements »**
- **« Combining Interval Arithmetic with the Branch and Bound Algorithm for Delta-lognormal Parameter Extraction »**
- **« A Global Optimization Algorithm for the Estimation of Delta- Lognormal Parameters Characterizing Pointing Gestures**

related in detail the main ideas developed in this thesis. The contributions **« On the use of Interval Arithmetic to Bound Delta- Lognormal Rapid Human Movements Models »** and

« **Combining Interval Arithmetic with the Branch and Bound Algorithm for Delta-lognormal Parameter Extraction** » handle with the branch (Separation) aspect of the algorithm. They contain all the techniques relating to the design of the search space in which the search for solutions will be performed. The chapter « **On the Use of Interval Arithmetic for the Branch and Bound Delta-Lognormal Parameter Extraction of Rapid Human Movements** » in the book “The Lognormality Principle and its applications in E-Security, E-learning and E-Health” and the publication « **A Global Optimization Algorithm for the Estimation of Delta-Lognormal Parameters Characterizing Pointing Gestures** » handle with the evaluation (Bound). In these two contributions, different rules for each step of the Bound (evaluation) are proposed and inserted in the Branch (separation) part to form the Delta-Lognormal parameters extractor.

The following two secondary publications:

- « **An Interactive Tablet-Based System to Run Neuromuscular Tests** »
- « **A New Experimental Set-Up To Run Neuromuscular Test** »

Detail the process of collecting and acquiring data from simple handwriting movements to test the extractor.

Interval arithmetic is used in this thesis as the mathematical tool for the various calculations in the four main contributions. An unpublished library containing all the basic arithmetic and transcendental functions found in the delta and Sigma-lognormal functions have been entirely developed in advance programming language (C++). Details on the design of the library are part of the unpublished text in this thesis.

The algorithm uses the Delta-Lognormal equation considered to be the most accurate for representing fast pointing movements to analyze some neurological disorders.

Beyond medical applications, the possibility of extracting globally optimal solutions opens new perspectives to:

- Carrying out fundamental studies on the properties of Lognormal models in human movement's analysis.
- Characterize accuracy of other Delta-Lognormal extraction tools.

- Further applications where a locally optimal solution to that class of problem is no longer sufficient.

In addition to the main and secondary contributions, this thesis includes the technical details around the development of the **S&E** algorithm and the development of the databases of fast handwritten movements (synthetic and real) to test the designed extractor.

TABLE DES MATIÈRES

REMERCIEMENTS	III
RÉSUMÉ	IV
ABSTRACT.....	VII
TABLE DES MATIÈRES	X
LISTE DES TABLEAUX.....	XIV
LISTE DES FIGURES	XVII
LISTE DES SIGLES ET ABRÉVIATIONS	XXI
LISTE DES ANNEXES.....	XXII
CHAPITRE 1 INTRODUCTION	1
1.1 Présentation du cadre de la thèse	1
1.2 Modèle de conception Delta-Lognormal	3
1.3 Structure de la thèse	4
CHAPITRE 2 REVUE DE LITTÉRATURE	6
2.1 L'étude des mouvements humains basée sur la théorie cinématique des mouvements rapides et les modèles lognormaux qu'elle propose.....	6
2.1.1 Présentation des modèles lognormaux	7
2.1.2 Grandeur cinématique de la modélisation lognormale	11
2.1.3 Modélisation du Bruit d'ajustement de la lognormale	12
2.1.4 Avantage de la lognormalité pour la modélisation de mouvements simples.	13
2.1.5 Synthèse des travaux sur les mouvements humains basée sur la théorie cinématique des mouvements rapides et les modèles lognormaux qu'elle propose.	15
2.2 Généralités sur l'arithmétique par intervalles	16
2.2.1 Concepts et termes de bases	16
2.2.2 Operations arithmétiques.....	20

2.2.3 La Dépendances	21
2.3 L'analyse Lognormale des mouvements humains à partir des données provenant de mouvements d'écritures manuscrites.....	21
2.3.1 Extracteur Sigma-lognormaux	22
2.3.2 Extracteurs Delta-Lognormaux	26
2.3.3 Évaluation des extracteurs Delta-Lognormaux Actuels et objectifs de recherches	35
CHAPITRE 3 ÉTAPE SUIVIES EN VUE D'ÉLABORER UN OUTIL D'ANALYSE AMÉLIORÉ POUR L'EXTRACTION DES PARAMÈTRES DELTA-LOGNORMAUX EN EXPLOITANT UN ALGORITHME DE SÉPARATION ET ÉVALUATION	42
3.1 Banque de données : mouvements réels et synthétiques.	43
3.1.1 Banque de données réelles	43
3.1.2 Données synthétiques	46
3.2 Extracteurs	47
3.2.1 Définition et réalisation de l'espace de recherche : utilisation de l'arithmétique par intervalles	49
3.2.2 La détermination des bornes supérieures et inférieures d'un espace en utilisant l'arithmétique par intervalles sur neuf classes de vitesses synthétiques.	51
3.2.3 Séparation des espaces en sous-espaces et évaluation comparative des sous-espaces	52
3.3 Banc de test:.....	53
3.4 Synthèse des étapes requises pour la réalisation d'un outil capable d'extraire les paramètres Delta-Lognormaux des mouvements manuscrits simples	55
CHAPITRE 4 ARITHMÉTIQUE PAR INTERVALLES ET THÉORÈME FONDAMENTAL	56
4.1 Opération sur les fonctions transcendentales de bases	56
4.1.1 Dépendances de l'AI dans le cas des fonctions transcendentales	58
4.1.2 Propriétés et définitions de bases nécessaires à L'algorithme S&E du chapitre 5	59

4.2 L'implémentation sur ordinateur de la librairie des fonctions de bases	61
4.3 Théorème fondamentale de l'AI	61
CHAPITRE 5 ARTICLE 4: GLOBAL OPTIMIZATION ALGORITHM FOR THE ESTIMATION OF THE DELTA-LOGNORMAL PARAMETERS CHARACTERIZING POINTING GESTURES	63
5.1 Introduction.....	64
5.2 The Delta-Lognormal model	66
5.3 State-of-the-art Delta-Lognormal extractors	68
5.4 Definitions, notations, and fundamental theorem.....	70
5.5 The multidimensional Interval Arithmetic Branch and Bound Delta-Lognormal Parameter Extractor:	72
5.6 Testing the algorithm with ideal velocity profiles	83
5.6.1 Tests conditions.....	84
5.6.2 Evaluation of the extractor under ideal conditions.....	84
5.6.3 Evaluation of the Extractor under Noisy Conditions	87
5.7 Performance Evaluation with Real Human Movements.....	88
5.8 Conclusion	92
CHAPITRE 6 DÉVELOPPEMENT INÉDITS ET COMPLÉMENTAIRES À L'ALGORITHME DE SÉPARATION ET ÉVALUATION	96
6.1 Élaboration des mouvements synthétiques simples	96
6.1.1 Variabilité dans le temps des mouvements manuscrits simples et classifications des profils de vitesses.	97
6.1.2 Construction des différentes classes de profils de vitesses Delta-Lognormales	98
6.1.3 Résultats du processus de génération de profils de vitesses synthétiques.....	105
6.1.4 Synthèse des travaux sur la génération de profils de vitesse synthétiques.....	106

6.2 Résultats détaillés des délais d'extractions des paramètres Delta-Lognormaux par l'algorithme S&E	107
6.2.1 Échantillons de tests pour l'évaluation des délais d'extractions des profils de vitesses synthétiques par l'algorithme S&E conçus	107
6.2.2 Échantillons de tests pour l'évaluation des délais d'extractions des profils de vitesses synthétiques par l'algorithme S&E conçus	108
6.2.3 Résultats des tests sur les données synthétiques	109
6.2.4 Résultats des tests sur les données réelles	111
6.2.5 Synthèse des résultats sur les temps d'extractions des différents profils.	113
6.3 Synthèse des développements inédits	114
CHAPITRE 7 DISCUSSION GÉNÉRALE.....	115
7.1 Ensemble Φ de la procédure de séparation	115
7.2 Sous-ensemble réduit de Φ de la procédure de séparation	116
7.3 Discussion sur la procédure d'évaluation	117
7.4 Discussion relative au modèle Delta-Lognormal de la théorie cinématique du mouvement rapide et l'extracteur S&E	121
CHAPITRE 8 CONCLUSION ET RECOMMANDATIONS	122
RÉFÉRENCES	131
ANNEXES	144

LISTE DES TABLEAUX

Tableau 3.1 Échantillon expérimental recueillis avec le lognomètre pour entraîner l'algorithme S&E.....	45
Tableau 3.2 Échantillon expérimental des données synthétiques l'algorithme S&E	47
Tableau 5.1 Basic interval arithmetic operations and functions for the Delta-Lognormal	70
Tableau 5.2 Initial box	85
Tableau 5.3 Statistics of the absolute error on the parameters estimated by the IAB&BPE on ideal data.....	85
Tableau 5.4 Extraction (in %) of the exact value of the parameters for a snr>100db as ending criterion	86
Tableau 5.5 Parameter value of the speed profile shown in Figure 5.8.C for the gesture shown in Figure 5.8.A the extracted parameters are for a SNR =31dB	90
Tableau 6.1 Classes de profils de vitesses lognormaux idéaux.....	98
Tableau 6.2 Échantillons de synthétiques pour l'évaluation des temps d'exécution de l'algorithme S&E.....	108
Tableau 6.3 Échantillon de données réelles pour l'évaluation des temps d'exécutions de l'algorithme S&E.....	109
Tableau 6.4 Résultats détaillés des délais moyens d'extraction des paramètres Delta-Lognormaux des profils de vitesses synthétiques par l'algorithme S&E.....	110
Tableau 6.5 Résultats de l'ANOVA sur les délais moyens d'extraction des paramètres Delta-Lognormaux des profils de vitesses synthétiques par l'algorithme S&E	110
Tableau 6.6 Résultats des tests de Student sur les délais d'extraction des données synthétiques par l'algorithme S&E.....	111
Tableau 6.7 Résultats détaillés des délais moyens d'extraction des paramètres Delta-Lognormaux des profils de vitesses synthétiques par l'algorithme S&E.....	112
Tableau 6.8 Résultats de l'ANOVA sur les délais moyens d'extraction des paramètres Delta-lognormaux des profils de vitesses des données réelles par l'algorithme S&E	112

Tableau 6.9 Résultats des tests de Student sur les délais d'extractions des données réelles par l'algorithme S&E.....	112
Tableau A.1 The initial box.....	152
Tableau A.2 Statistics of the absolute error on the parameters estimated from ideal Delta-lognormal velocity profile	153
Tableau A.3 Statistics of the absolute error on the parameters estimated from ideal Delta-lognormal velocity profile (the bounding box)	153
Tableau A.4 Comparison of the number of boxes explored to extract one velocity profile and the processing time required to perform 11 iterations in the main loop	155
Tableau B.1 Basic arithmetic operations between two intervals	161
Tableau B.2 Basic arithmetic functions	161
Tableau B.3 The initial box.....	167
Tableau B.4 Statistics of the absolute error on the parameters estimated from ideal Delta- Lognormal velocity profiles	168
Tableau B.5 Statistics of the absolute error on the parameters estimated from ideal Delta- Lognormal velocity profiles (the bounding box).....	169
Tableau B.6 Comparison of the number of boxes explored to extract one velocity profile and the processing time required to perform 11 iterations in the main loop.....	170
Tableau C.1 Extraction of the exact value of the parameters for an SNR of >100db as ending criterion.....	179
Tableau C.2 Performance comparison of the number of boxes explored to extract one velocity profile and the processing time required to perform 11 iterations in the main loop	179
Tableau D.1 Result of t-test for independent group between the new and the old system with the visual cue	188
Tableau D.2 Result of t-test for independent group between the new and the old system with the sound cue	189

Tableau E.1 The Delta-Lognormal parameters extracted by the software for the stroke presented on Figure E.5.....	201
Tableau E.2 The Sigma-Lognormal parameters extracted by the software for the stroke presented on Figure E.5.....	202
Tableau E.3 Result of t-test for independent groups between the new and the old system with the visual cue using the Delta-Lognormal model	203
Tableau E.4 Result of t-test for independent groups between the new and the old system with the sound cue using the Delta-Lognormal model	203
Tableau E.5 Result of t-test for independent groups between the new and the old system with the visual cue using the Sigma-Lognormal model.....	204
Tableau E.6 Result of t-test for independent groups between the new and the old system with the sound cue using the Sigma-Lognormal model.....	204

LISTE DES FIGURES

Figure 2.1 Courbes de profil de vitesse Delta-Lognormale ainsi que ses composantes agoniste et antagoniste.....	8
Figure 2.2 Angle de départ, angle de fin et pivot associé au modèle Sigma-Lognormal	9
Figure 2.3 Intervalle de la variable X.....	16
Figure 2.4 Largeur d'un intervalle	17
Figure 2.5 Milieu d'un intervalle matérialisé par le point $m(x)$	17
Figure 2.6 Vecteur d'intervalles de dimension n . Largeur de la boite ($W(x)$).....	18
Figure 2.7 Milieu d'une boite	19
Figure 2.8 Exemple de division d'une boite en deux sous boites	19
Figure 2.9 Schéma du système d'extraction des paramètres Lognormaux.....	25
Figure 2.10 Caractéristiques globales utilisées par le modèle de Wise. (I_1, I_2 pente des tangentes aux points d'inflexion). (C, t_1, t_2) triangle permettant d'approximer une composante neuromusculaire	27
Figure 2.11 Points caractéristiques choisis (t_0, t_i, t_j, t_m) pour l'algorithme INITRI	29
Figure 2.12 Variation des paramètres de la fonction lognormale en fonction du temps : les t_i sont des intervalles de temps définissants le domaine de validités des paramètres.....	32
Figure 2.13 Borne supérieure de la lognormale	33
Figure 2.14 Borne inférieure de la lognormale	34
Figure 2.15 Enveloppe Delta-Lognormale en rouge renfermant des Delta-Lognormales en bleue	35
Figure 3.1 Architecture du système d'extraction proposée	42
Figure 3.2 Schéma bloc de la modélisation des données synthétiques	46
Figure 3.3 Schéma bloc du banc de test de l'algorithme S&E	54

Figure 5.1 A human handwriting pointing gesture: a. the (x, y) position of the pen tip movement, b. the real and reconstructed velocity profile with a 31dB SNR, c. the reconstructed velocity profile decomposed into agonist and antagonist contributions	67
Figure 5.2 Bounding box derived from the initial rule: A. Real velocity profile enclosed by its envelope computed using the NIE. B. An ideal velocity profile enclosed by its envelope computed using the NIE	75
Figure 5.3 Experimental testing of the splitting rule: the execution time and memory space used to extract a handwriting gesture when use: the gradient multiplied by the box dimension (blue (c) and red (D) line), the normalised box dimension (brown (A) and black (B))	76
Figure 5.4 Experimental validation of the Delta-Lognormal envelope	80
Figure 5.5 Efficiency of the discarding test: the number of boxes rejected by the discarding tests (blue), the number of boxes in the memory (red).....	82
Figure 5.6 Ideal velocity profile parameters extracted by IAB&BPE. A. Upper and lower envelope surrounding the ideal and extracted speed profile. B. original and extracted velocity profile, with the last one being decomposed into its agonist and antagonist neuromuscular	86
Figure 5.7 Ideal noisy velocity profile parameters extracted by IAB&B. Superior and inferior envelopes surrounding the ideal and extracted speed profiles are plotted	87
Figure 5.8 Example of human Handwriting strokes extracted by IAB&BPE : A. the (x, y) position of the pen tip movement of a writer, B. the real and reconstructed velocity profile with a 31dB SNR enclosing their envelopes, C. the reconstructed velocity profile	89
Figure 5.9 Execution time and memory space during the extraction of a real handwriting pointing gesture. Note that a same axis is used both for the time in ms and for the memory space expressed in boxes.....	91
Figure 5.10 Example of parkinsonian handwriting pointing gesture extracted by IAB&BPE for a 26dB SNR : the envelope enclosing the original (green) and the $\Delta\Lambda$ (red) velocity profile	92
Figure 6.1 Lognormale avec lobe inférieur à droite du lobe principal.....	96

Figure 6.2 Distribution lognormale avec sa surface peinte en rouge	101
Figure 6.3 Toutes les classes des courbes delta-lognormales : en traits continus les courbes Delta-Lognormales obtenues par soustraction des deux courbes en pointillés	105
Figure 7.1 Boite englobante obtenue avec l'algorithme S&E pour SNR fixé à 20dB.....	119
Figure 7.2 Boite englobante obtenue avec l'algorithme S&E pour SNR fixé à 23dB.....	119
Figure 7.3 Boite englobante obtenue avec l'algorithme S&E pour SNR fixé 28dB	120
Figure 7.4 Boite englobante obtenue avec l'algorithme S&E pour SNR fixé à 25dB.....	120
Figure A.1 Delta-lognormal velocity profile example	151
Figure A.2 Experimental testing of the Delta-Lognormal envelope	152
Figure A.3 Example of application of algorithm 1 in extracting real human pen-tip movement	154
Figure B.1 Delta-Lognormal velocity profile examples	166
Figure B.2 Experimental testing of the Delta-Lognormal envelope	168
Figure B.3 Example of application of algorithm 1 in extracting real human pen-tip movement (the top trace in blue is the fast stroke, the middle traces are respectively the original (blue solid line) and the reconstructed (red stars line) velocity profile	170
Figure D.1 The bloc Diagram of the old acquisition system	183
Figure D.2 The bloc diagram of the global system including the new acquisition system.....	184
Figure D.3 The interface displayed on the screen of the tablet including the guide sheet on the center with the visual cue on each side	186
Figure E.1 The bloc diagram of the old acquisition system.....	193
Figure E.2 The bloc diagram of the global system including the new acquisition system called the Lognometer	194
Figure E.3 The user interface displayed on the computer monitors shown to the user	196
Figure E.4 The interface displayed on the screen of the tablet including the guide sheet on the center with the fast stroke, and the visual cue on each side.....	197

Figure E.5 An original stroke executed by a subject to be reconstructed by the two models.....	199
Figure E.6 The Delta-Lognormal reconstruction of the velocity of the stroke shown of Figure E.5	200
Figure E.7 The Sigma-Lognormal reconstruction of the velocity of the stroke shown of Figure E.5	201
Figure E.8 Different guiding sheet and stimuli integrated in the new system. a. guiding sheet for the simple stroke test with the red light, this guide sheet is used with a visual and sound cue. b. A simple fast stroke displayed on the guide sheet. c. The orange cross displayed is not valid. Guiding sheets for the fast pen stroke on a choice reaction time protocol (d) , (e). f,g,h,i are guiding sheets for the speed/accuracy trade-off task. j,k,l are guiding sheets for the fast pen stroke sequences	206

LISTE DES SIGLES ET ABRÉVIATIONS

ΣΛ Somme des lognormales

ΔΛ Delta-Lognormale

AVC Accident vasculaire cérébrale

EMG Électromyographies

HWS Handwriting Samples

IAB&BPE Interval arithmetic Branch and Bound Parameter Extraction

RSB Rapport signal sur bruit

S&E Séparation et évaluation

SNM Système neuromoteur

TDA Trouble déficitaire de l'attention

TDAH Trouble déficitaire de l'attention avec Hyperactivité

TCMHR Théorie Cinématique du Mouvement Humain Rapide

LISTE DES ANNEXES

ANNEXE A Article 1: ON THE USED OF INTERVAL ARITHMETIC TO BOUND DELTA- LOGNORMAL RAPID HUMAN MOVEMENTS MODELS	144
ANNEXE B Article 2: ON THE USE OF INTERVAL ARITHMETIC FOR BRANCH AND BOUND DELTA-LOGNORMAL PARAMETER EXTRACTION OF RAPID HUMAN MOVEMENTS	156
ANNEXE C Article 3: COMBINING INTERVAL ARITHMETIC WITH THE BRANCH AND BOUND ALGORITHM FOR DELTA-LOGNORMAL PARAMETER EXTRACTION	172
ANNEXE D Article 5: A NEW EXPERIMENTAL SET-UP TO RUN NEUROMUSCULAR TESTS.....	181
ANNEXE E Article 6: AN INTERACTIVE TABLET-BASED SYSTEM TO RUN NEUROMUSCULAR TESTS.....	191

CHAPITRE 1 INTRODUCTION

1.1 Présentation du cadre de la thèse

En dépit des grands progrès réalisés au cours des dernières décennies dans la réduction de la mortalité, les maladies neurodégénératives constituent un fardeau important pour la population vieillissante. Leur grande incidence sur le vieillissement ainsi que les impacts sévères sur l'autonomie et le bien-être des patients atteints constituent des causes importantes d'invalidité. La hausse de la prévalence actuelle associée aux traitements efficaces (chirurgie) justifie la baisse de la mortalité qui s'y rapporte. En revanche, ces changements tendent à modifier le profil clinique de ces maladies et les rendent plus difficiles à traiter avec le vieillissement. Face à ces défis, il est important d'élaborer de nouvelles stratégies pour gérer les effets du vieillissement. La détection précoce et plus précise des maladies neurodégénératives apparaît comme une alternative.

La cinématique du mouvement humain pourrait apporter certaines réponses à la prévention des pathologies dégénératives. En effet, le mouvement humain est le résultat du contrôle du système moteur qui fonctionne avec plusieurs degrés de liberté. Une modélisation rigoureuse de la cinématique inverse partant des observations des membres pour remonter à l'effecteur du mouvement pourrait apporter des informations sur la variation ou l'état du système moteur.

Cette thèse fait état d'un outil d'analyse des mouvements de pointage humains basée sur la Théorie Cinématique du Mouvement Humain Rapide. Cette théorie dont les fondements sont axés sur un modèle physiologique et anatomique du système neuromoteur le représente par une équation mathématique comportant sept paramètres. Parmi eux, quatre sont assimilables directement au système neuromoteur ($\mu_1 \mu_2 \sigma_1 \sigma_2$) et trois aux commandes neuromotrices ($t_0 D_1 D_2$). Les travaux publiés en utilisant cette théorie soutiennent que le profil de vitesse est le paramètre cinématique invariant par rapport aux changements externes observés sur l'effecteur terminal. L'aspect physiologique sur lequel se fondent les équations caractéristiques de cette théorie la rend très attractive pour l'étude des mouvements chez les humains.

L'écriture manuscrite en tant que classe spécifique de mouvements rapides observables, requiert un contrôle moteur fin et une coordination neuromusculaire spécifique pour son exécution. Son apprentissage chez les humains développe une fois acquis un mécanisme de commande moteur au niveau cérébral [1] qui rend automatique l'échange entre perception visuelle et commande

musculaire pendant son exécution. De ce fait, la détérioration de l'écriture peut être le signe d'une diminution ou d'un dysfonctionnement de ce programme moteur, causé par le vieillissement ou une pathologie d'ordre neurologique.

En éducation, diverses méthodes pédagogiques interactives ont été développées pour reproduire des lettres et des mots à partir de la concaténation de traits. De même, les traits manuscrits sont analysés pour étudier les déficits d'écriture chez les enfants [2] [3], ainsi que pour entraîner les patients souffrant de crampe de l'écrivain [4]. Dans les neurosciences, l'écriture manuscrite est étudiée pour caractériser les processus neurodégénératifs comme la maladie de Parkinson, la maladie d'Alzheimer [5] [6] et la sclérose en plaques [7]. L'écriture manuscrite possède des caractéristiques invariantes telles que :

- Le temps de réaction requis par un individu pour atteindre une cible. Ce dernier est fonction de la largeur de la cible et de la distance à laquelle elle se situe par rapport à un référentiel fixé. Cette observation implique un compromis vitesse/précision pour l'émission d'un geste manuscrit. [8]
- La forme du profil de vitesse tangentielle qui se présente comme une cloche asymétrique qui s'atténue en fonction de la durée du mouvement [9] [10].
- Le lien étroit existant entre la vitesse tangentielle d'un mouvement rapide et la trajectoire effectuée [11].

Ces particularités font de l'analyse des mouvements d'écriture un terrain expérimental de choix pour l'analyse d'une catégorie de troubles neurologiques.

D'autre part, les récentes innovations dans le domaine de l'électronique et du logiciel avec la création d'appareils portables capables de numériser des expressions manuscrites la rendent encore plus attractive. En effet, ces outils évolués permettent l'acquisition des mouvements de pointage de manière précise à travers des stylos électroniques ou le bout des doigts. Les informations riches en cinématique du scripteur recueillies à partir de ces instruments nécessitent des paradigmes d'expérimentation de mouvement adaptés à leurs caractéristiques pour mieux les interpréter.

Beaucoup de ces exemples soulignent la nécessité d'étudier les propriétés de base des traits simples pour une analyse réaliste de la façon dont le système de contrôle moteur effectue des mouvements complexes.

1.2 Modèle de conception Delta-Lognormal

Parmi la vaste majorité des modèles proposés pour reproduire les caractéristiques des mouvements rapides, les modèles lognormaux de la théorie Cinématique du Mouvement Humain Rapide [12] [13] [14] proposé par Plamondon sont utilisés pour la modélisation des mouvements rapides de pointage de cette étude. Ces modèles [13] [14] peuvent les décrire avec précision, car, pendant l'écriture, le système moteur effectue une séquence rapide de mouvements fins et précis qui sont le résultat d'un apprentissage tout au long de la vie. Cette théorie présente des traits d'écritures manuscrites comme des primitives contenant de l'information sur le contrôle moteur et le système neuromusculaire. Les modèles lognormaux qui dérivent de cette théorie tiennent en compte toute la structure biomécanique du système moteur allant de l'émission du signal de base jusqu'à la production des écrits. Ils considèrent l'asymétrie des profils de vitesse, la synergie agoniste antagoniste entre les muscles en action, et l'émission d'une commande à un instant précis pour le déclenchement du mouvement. De plus, les fonctions analytiques des modèles résultent de l'émergence d'un nombre fini de sous-systèmes neuromusculaires connectés entre eux de manière hiérarchique. L'extraction des paramètres Delta-Lognormaux à partir des données de mouvement manuscrit fait l'objet de plusieurs études depuis les deux dernières décennies. En effet, les paramètres Delta-Lognormaux issus des mouvements de pointage semblent constituer une voie porteuse d'espoir pour le dépistage précoce du développement ou de la susceptibilité au développement de certaines maladies, notamment les maladies neurodégénératives [15] [16] accidents vasculaires cérébraux (AVC) [17], les troubles déficitaires de l'attention avec ou sans hyperactivité (TDA, TDAH) [18]. À ce titre, multiples extracteurs ont été élaborés pour résoudre la panoplie d'applications en reconnaissance de formes qu'est capable d'asservir le modèle Delta-Lognormale. En considérant l'ensemble des éléments mentionnés ci-dessus, le présent projet de recherche s'est donné comme objectif la conception et l'implémentation de l'algorithme de séparation et évaluation (Branch and Bound) appliquée au modèle Delta-Lognormal. Le tout permettra d'extraire des paramètres lognormaux globalement optimaux pour l'investigation des troubles neuromusculaires chez l'humain.

1.3 Structure de la thèse

La répartition et l'agencement de tous les développements de cette thèse formée d'articles techniques et de textes inédits s'échelonnent comme suit :

Le chapitre un (1) est consacré à l'introduction, il couvre l'ensemble des éléments mentionnés ci-dessus à savoir; le cadre théorique de l'application qui se situe dans l'analyse des mouvements humains rapides de pointages. La problématique à résoudre qui vise à construire un algorithme robuste, à caractère global, produisant des solutions exactes, capables de satisfaire aux contraintes de travail en milieux cliniques, afin de permettre l'avancement des études où les solutions locales sont moins efficaces.

Le chapitre deux (2) propose une synthèse de l'état de l'art sur les travaux en lien avec la problématique, on y retrouve des développements sur la théorie cinématique des mouvements rapides et ses modèles lognormaux. Une revue de la littérature sur l'ensemble des extracteurs conçus dans le cadre de différentes études sur les mouvements d'écritures manuscrites. Des algorithmes associés au modèle vectoriel multidimensionnel Sigma-Lognormal et au modèle à sept dimensions Delta-Lognormal de la théorie cinématique du mouvement humain rapide sont présentés. Le chapitre se conclut par l'identification du problème à résoudre suivie des objectifs et hypothèses nécessaires à l'atteinte des objectifs.

Le chapitre trois (3) présente la structure générale de cette thèse et il en situe les différentes contributions en lien avec la problématique à laquelle elle apporte des solutions concrètes et pratiques. On retrouve dans ce chapitre; l'architecture générale du système d'extraction, la description des différents modules le constituant, les relations entre les principaux modules, les contributions scientifiques et la problématique à résoudre.

Les chapitres quatre (4), présentent les développements algorithmiques en arithmétique par intervalles nécessaire à la compréhension des techniques proposées aux chapitres 5.

Le chapitre cinq (5), présente l'outil d'extraction des paramètres Delta-Lognormaux par un algorithme de type **S&E**.

Le chapitre six (6), fait état des textes inédits et des développements complémentaires associés aux grands blocs constituants l'outil d'extraction des paramètres Delta-Lognormaux. Il couvre les aspects complémentaires aux publications de revues et conférences présentées au chapitre 5, et aux

annexes A B C D et E. Le chapitre sept (7) discute des principales méthodes utilisées pour atteindre les objectifs fixés. Le chapitre huit (8) résume les principales contributions novatrices apportées par cette thèse. Il évalue l'atteinte des objectifs et la validité des hypothèses émises. Finalement il conclut et formule des recommandations.

Les annexes A, B, et C présentent respectivement les différentes publications liées à la problématique. Toutes les contributions techniques y sont illustrées telles que publiées dans les conférences, et livres.

CHAPITRE 2 REVUE DE LITTÉRATURE

Ce chapitre porte sur l'état de l'art de l'ensemble des éléments nécessaires à la réalisation des objectifs de cette thèse. Il se divise en trois sections distinctes.

- L'étude des mouvements humains basée sur la théorie Cinématique des Mouvements Rapides et les modèles Lognormaux qu'elle propose.

Les sujets considérés dans cette section sont les développements mathématiques liés à la modélisation Lognormale, les grandeurs cinématiques associées aux modèles lognormaux, la prédiction d'une commande cérébrale par les équations Lognormales et la possibilité de l'évaluer grâce à un ajustement de la courbe Delta-Lognormale. Nous présentons également la modélisation du bruit d'ajustement des courbes de vitesse avec les modèles lognormaux, ainsi que les avantages de la lognormalité pour la modélisation d'un mouvement simple.

- Généralités sur l'arithmétique par intervalles

Les sujets considérés sont les définitions et concepts de bases de l'arithmétique par intervalles nécessaire à la compréhension de l'algorithme S&E proposé au chapitre 5.

- L'analyse Delta-Lognormale des mouvements humains à partir des données provenant de mouvements d'écriture manuscrite.

Le sujet considéré ici est l'épineux problème d'extraction des paramètres d'un modèle non linéaire à sept dimensions. La discussion couvre différents algorithmes de régression non linéaire qui ont été proposés dans le cadre des études antérieures et dont le but était d'extraire les paramètres lognormaux à partir des mouvements numérisés.

2.1 L'étude des mouvements humains basée sur la théorie cinématique des mouvements rapides et les modèles lognormaux qu'elle propose.

L'ensemble des analyses menées dans ce projet reposent sur les modèles lognormaux et l'étude du mouvement humain. La Théorie Cinématique du Mouvement Humain Rapide (**TCMHR**) [12] présente le système neuromoteur comme un générateur de mouvements ayant à son entrée une impulsion et procurant en sortie une réponse impulsionale représentée par le profil de vitesse

tangentielle du mouvement rapide de l'effecteur terminal. Elle modélise le **SNM** par le produit de convolution d'un très grand nombre de sous-systèmes linéaires dont la réponse impulsionale représente l'effet de toute la partie neuronale prise à partir des aires motrices jusqu'aux muscles. Lors de l'exécution d'un mouvement rapide d'un point initial à une cible, le profil de vitesse résultant est une courbure en forme de cloche asymétrique [19] [14]. Cette dernière est le résultat d'une commande neuromotrice liée au mouvement volontaire émis et représentée par une impulsion pondérée dont le module du vecteur résultant correspond à la distance à parcourir. La direction est le déplacement rattaché aux deux supports fixés sur le point initial et final de la trajectoire respectivement. La quantité cinématique permettant d'étudier la réponse impulsionale du **SNM** est le profil de vitesse du mouvement de l'effecteur terminal. Ce dernier ressemble beaucoup aux réponses impulsionales du **SNM** et comporte moins d'oscillations. Dans le cas d'un mouvement rapide avec trajectoire rectiligne, le profil de vitesse tangentielle du mouvement qui le génère est le résultat de la superposition de l'action des contractions musculaires agoniste et de celle du relâchement des muscles antagonistes. Les muscles agonistes initient et maintiennent le mouvement dans la direction spécifiée par la tâche à réaliser tandis que les antagonistes représentent toutes les actions et effets qui s'opposent à l'action de l'agoniste. Ils sont orientés dans la direction opposée au mouvement. Les commandes impulsionales des deux muscles sont émises au temps t_0 .

2.1.1 Présentation des modèles lognormaux

Dans le cadre de la **TCMHR**, la réponse impulsionale du système moteur est représentée par le produit de convolution des réponses d'un grand nombre de sous-systèmes indépendants sous l'hypothèse de proportionnalité [14]. Les profils de vitesse résultants ont une forme Lognormale [12]. On peut modéliser les mouvements rapides tels que ceux de pointage comme le résultat de la soustraction entre deux Lognormales, dont l'amplitude est pondérée respectivement par les facteurs agonistes (D_1) et antagonistes (D_2). Si on ajoute à cela l'hypothèse d'émission d'une commande d'activation provenant du système nerveux central pour activer les deux groupes neuromusculaires, on obtient alors le modèle Delta-Lognormal tel que proposé dans [12] [19] [20] [14]. Son équation caractéristique se formule ainsi :

$$\Delta\Lambda(t; p) = D_1\Lambda_1(t, t_0, \mu_1, \sigma_1) - D_2\Lambda_2(t; t_0, \mu_2, \sigma_2) \quad (2.1)$$

Avec lambda la réponse impulsionale lognormale définie par :

$$\Lambda(t; t_0, \mu, \sigma^2) = \begin{cases} \frac{1}{\sigma\sqrt{2\pi}(t - t_0)} \exp^{-\frac{(\ln(t-t_0)-\mu)^2}{2\sigma^2}}, & \text{for } t_0 < t, \\ 0, & \text{ailleurs,} \\ \mu & > 0 \\ & \in \mathbb{R} \end{cases} \quad (2.2)$$

La figure 2.1 illustre le profil de vitesse lognormal ainsi que les deux composantes agoniste et antagoniste permettant l'établissement du mouvement.

Dans l'équation 2.1, D_1 et D_2 représentent les amplitudes des commandes neuromusculaires des composantes agonistes et antagonistes respectivement. t_0 est le temps d'activation du système neuromusculaire, μ_1, μ_2 , représentent les temps de délais de la réponse impulsionale des composantes agoniste et antagoniste respectivement. σ_1 et σ_2 sont les temps de réponse de la réponse impulsionale des deux composantes neuromusculaires respectivement. L'ensemble des propriétés temporelles du système sont sur une échelle logarithmique sauf le temps t_0 qui est exprimé en secondes.

Cependant, le modèle unidimensionnel Delta-Lognormal possède une limitation lorsque les mouvements deviennent plus complexes (cas d'une signature). Cette limitation provient du fait qu'elle ne permet pas de prendre en compte la direction changeante du mouvement. Néanmoins, ce problème est résolu dans [21] où une modélisation Sigma-Lognormale y est présentée. Dans ce modèle, une séquence de commande d'amplitude D_i sont émises aux temps t_{0i} , chacune d'entre

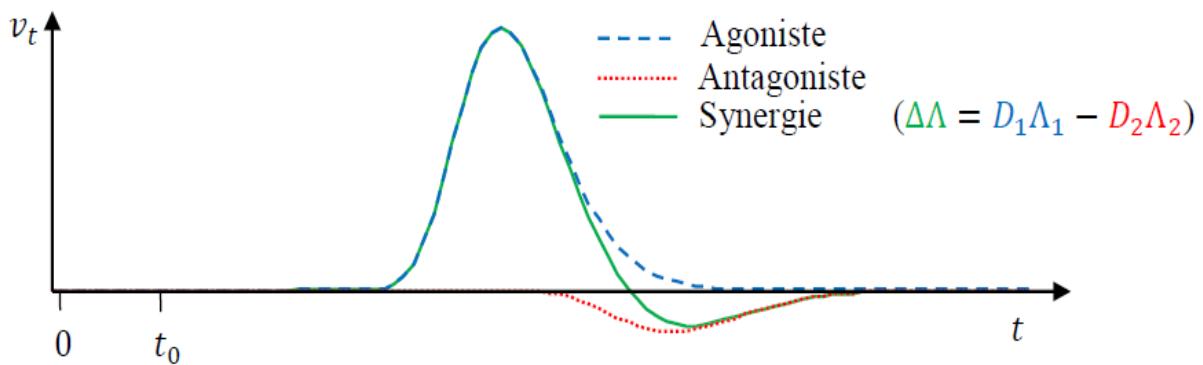


Figure 2.1 Courbes de profil de vitesse Delta-Lognormale ainsi que ses composantes agoniste et antagoniste

elles servant à activer un groupe neuromusculaire caractérisé par une réponse impulsionale

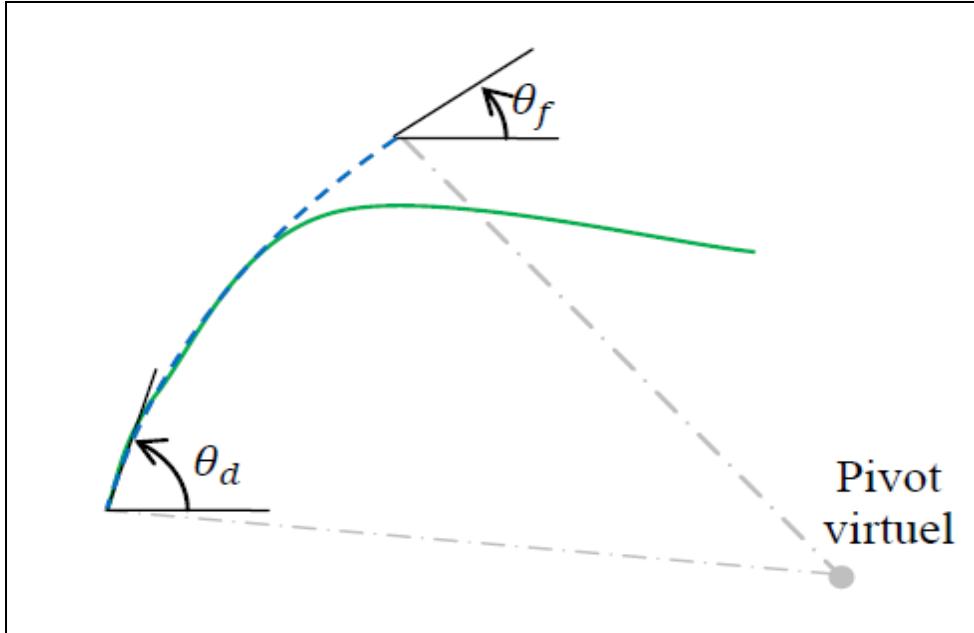


Figure 2.2 Angle de départ, angle de fin et pivot associé au modèle Sigma-Lognormal

Lognormale ayant les paramètres σ_i, μ_i , afin de reproduire des mouvements en deux dimensions, il faut de plus disposer, de paramètres donnant la direction du mouvement. Pour le modèle adopté, on suppose que chaque composante neuromusculaire agit autour d'un point pivot. Ce dernier peut être réel, comme dans le cas d'un mouvement de l'avant-bras où le coude serait fixe et ferait office de pivot.

Cependant, il s'agit d'un point virtuel. Il résulte alors que la trajectoire de chaque composante neuromusculaire décrit un arc de cercle dont les angles de départ et de fin sont respectivement θ_{di} et θ_{fi} . L'équation (2.4) ci-dessous représente la vitesse tangentielle du mouvement, tandis que l'équation 5 décrit sa direction en fonction du temps.

$$\mathbf{v}_i(t) = \Lambda_i = \frac{1}{\sigma_i \sqrt{2\pi} (t - t_{0i})} \exp^{-\frac{(\ln(t - t_{0i}) - \mu_i)^2}{2\sigma_i^2}} \quad (2.4)$$

$$\phi_i(t) = \theta_{di} + \frac{(\theta_{fi} - \theta_{di})}{D_i} \int_0^t v_{ti}(\tau; t_{0i}, \mu_i, \sigma_i) d\tau = \theta_{di} + \frac{(\theta_{fi} - \theta_{di})}{2} \operatorname{erfc}\left(\frac{\mu_i - \ln(t - t_{0i})}{\sqrt{2} \sigma_i}\right) \quad (2.5)$$

Avec : $\operatorname{erfc}(x) = \frac{2}{\sqrt{\pi}} \int_x^{\infty} e^{-t^2} dt$

Représentant la fonction d'erreur complémentaire. L'ensemble de ces composantes interagit synergétiquement à travers la somme vectorielle de manière à générer le mouvement complexe désiré.

$$\overrightarrow{v(t)} = \sum_{i=1}^N v_{ti}(t) \begin{bmatrix} \cos(\phi_i(t)) \\ \sin(\phi_i(t)) \end{bmatrix} = \begin{bmatrix} v_x(t) \\ v_y(t) \end{bmatrix}$$

Le modèle Sigma-lognormal décrit toute primitive d'un mouvement élémentaire simple ayant un profil de vitesse lognormal par la relation suivante :

$$\vec{v}(t) = \sum_{i=1}^N \vec{v}_i(t) = \sum_{i=1}^N \vec{D}_i \Lambda(t; t_0, \mu_i; \sigma_i^2) \quad N \geq 2$$

Et sa trajectoire est décrite par :

$$\vec{r}(t) = \vec{r}_0 + \int_{t_0}^t \vec{v}(\tau) d\tau$$

Avec $\vec{v}_i(t)$ le vecteur vitesse de la i^{eme} composante de $\vec{v}(t)$. Dans le cas d'un mouvement rapide effectué dans un plan, la synergie agoniste antagoniste est vectorielle et l'angle de déviation $\phi_i(t)$ constant. La seule composante angulaire considérée est la direction des deux composantes par rapport à leur position initiale et finale. Pour les angles de départ et d'arrivée égaux comme c'est le cas pour un mouvement rectiligne uniforme, $\theta_{di} = \theta_{fi} = \theta_i$ ($i=1$ pour agoniste et 2 pour antagoniste) la réponse impulsionnelle du **SNM** d'un mouvement élémentaire se traduit par la relation vectorielle :

$$\vec{v}(t - t_0) = \vec{D}_1 \Lambda_1(t, t_0, \mu_1, \sigma_1^2) + \vec{D}_2 \Lambda_2(t, t_0, \mu_2, \sigma_2^2) \quad (2.6)$$

Son module et sa direction sont données par les expressions suivantes respectivement :

$$\|\vec{v}(t - t_0)\| = \sqrt{D_1^2 \Lambda_1^2 + D_2^2 \Lambda_2^2 + 2D_1 D_2 \Lambda_1 \Lambda_2 \cos(\theta_1 - \theta_2)} \quad (2.7)$$

$$\theta = \tan^{-1} \left(\frac{D_1 \Lambda_1 \sin \theta_1 + D_2 \Lambda_2 \sin \theta_2}{D_1 \Lambda_1 \cos \theta_1 + D_2 \Lambda_2 \cos \theta_2} \right)$$

Pour le cas unidirectionnel associé au modèle Delta-Lognormal, avec les angles $\theta_{di} = \theta_{fi} = \theta_i = 0$ et $\theta_{di} = \theta_{fi} = \theta_i = \pi$ m, le module de la vitesse devient alors :

$$\|\vec{v}(t - t_0)\| = \sqrt{D_1^2 \Lambda_1^2 + D_2^2 \Lambda_2^2 - 2D_1 D_2 \Lambda_1 \Lambda_2}$$

$$\Delta\Lambda(t; p) = D_1 \Lambda_1 - D_2 \Lambda_2$$

Qui n'est autre que l'équation caractéristique du modèle Delta-Lognormal qui modélise le mouvement de pointage dans une direction.

Les modèles Delta et Sigma-Lognormaux ont en commun d'être le résultat de l'interaction synergétique des composantes lognormales pondérées et décalées temporellement. Les paramètres σ et μ possèdent la même signification que dans le modèle Delta-Lognormal, tandis que le paramètre t_0 possède une double nature. Dans un mouvement simple représenté par une Delta-Lognormale, il s'agit du temps minimum nécessaire avant de pouvoir émettre la commande motrice. En d'autres mots, t_0 est le temps d'activation des composantes musculaires en jeu dans l'accomplissement du mouvement. Dans un mouvement plus complexe, les t_0 deviennent des paramètres servant à synchroniser des sous-mouvements compris dans le mouvement d'ensemble. Ils décrivent alors exactement le moment où une commande doit être émise pour obtenir la trajectoire désirée. Cette valeur n'est plus un temps de latence associé à une limitation du système cognitif, mais une valeur imposée cognitivement dans la planification du mouvement. Avec ces deux modèles, tout mouvement de pointage peut être reconstruit. Cette reconstruction passe par la nécessité d'obtenir des paramètres maximisant l'ajustement de courbe entre le mouvement enregistré et le mouvement reconstruit par le modèle. Le choix d'une mesure permettant de répondre à cette question repose sur deux aspects physiques du problème. La première consiste à déterminer la grandeur physique par rapport à laquelle le bruit d'ajustement doit être évalué et la seconde repose sur l'évaluation de la qualité du signal reconstruit.

2.1.2 Grandeur cinématique de la modélisation lognormale

Les modèles Delta-Lognormaux modélisent les mouvements principalement au niveau de leur vitesse ou de leur position, les autres dimensions physiques comme l'accélération et les secousses ont peu d'intérêt. Le problème consiste à décider laquelle parmi la vitesse où la position est meilleure pour une modélisation du **SNM** dans le cas des mouvements rapides de pointage.

L'optimisation en position permet de retrouver des mouvements dont la trajectoire reconstruite est confondue à celle réellement parcourue. Les deux droites se collent. À l'opposé, l'optimisation en vitesse reconstruit la vitesse de manière efficace, mais, si la trajectoire reconstruite modélise mal une portion du mouvement original, il peut arriver qu'un décalage apparaisse entre les deux trajectoires et que celui-ci affecte toute la partie du mouvement suivant. Dans [22] [23], **INITRI**, **INFLEX** et autres, les auteurs proposent une méthode utilisant un ajustement par vitesses suivies d'un ajustement de la position. On peut clairement y voir le désavantage que procure l'ajustement par position sur la valeur des paramètres. Cette erreur est due au fait qu'une erreur locale dans le cas de l'optimisation en position sera compensée en aval dans le signal, de façon à annuler le décalage. Cette compensation a pour effet d'altérer la valeur des paramètres du modèle non seulement à l'endroit où le décalage s'est produit, mais aussi partout où la cinématique du mouvement reconstruit a été déformée pour compenser ce décalage. Au regard de la capacité du modèle Delta-Lognormal à modéliser la vitesse du mouvement et non sa position, notre étude reposera sur la vitesse.

2.1.3 Modélisation du Bruit d'ajustement de la lognormale

La modélisation du bruit d'ajustement et sa minimisation reposent sur la loi de la conservation de l'énergie mécanique. En effet, pendant l'accomplissement d'un geste de pointage, les muscles mis en jeux emmagasinent de l'énergie nécessaire à la réalisation de ce geste. La trajectoire décrite par ce mouvement correspond à la somme de toutes les énergies potentielles ainsi accumulées. D'après la loi sur la conservation de l'énergie, un corps ne peut transmettre intégralement l'énergie potentielle accumulée en énergie cinétique due aux pertes (frictions, effet joules), la trajectoire empruntée par le corps tend à minimiser les pertes en énergie mécanique. L'équation 2.8 avec T pour l'énergie potentielle et U pour l'énergie cinétique, traduit ce phénomène.

$$L = \int (T - U)^2 \quad (2.8)$$

La minimisation de 2.8 correspond donc à la trajectoire optimale. Dans le cas des courbes lognormales, deux mesures sont considérées ; l'erreur absolue et l'erreur quadratique. L'erreur quadratique mesure la variance résiduelle entre les grandeurs [24]. Elle favorise un écart d'ajustement également distribué sur l'ensemble du profil de vitesse. L'erreur absolue, pour sa part, permet aussi bien un écart distribué uniformément qu'un écart important localisé dans de petites sections du signal. Des tests validant l'impact des deux formes d'erreurs sur les modèles

lognormaux ont été menés dans [21]. Les résultats montrent que l'optimisation utilisant l'erreur absolue peut complètement négliger les pics secondaires et tertiaires d'un mouvement Delta-Lognormal au profit d'un ajustement optimal du pic primaire. Or, la forme des pics secondaires et tertiaires a une grande influence sur la valeur des paramètres extraits. De ces résultats découle le choix de l'utilisation d'un bruit reposant sur l'erreur quadratique.

Notons cependant qu'il est généralement préférable d'inclure l'erreur quadratique comme étant la composante bruit d'un rapport signal sur bruit (**RSB**). De cette façon, l'erreur de modélisation est mise à l'échelle relativement à l'énergie du signal étudié. On peut alors comparer la qualité des ajustements de courbe entre des mouvements d'amplitude (et donc d'énergie) différents. Une transformation logarithmique du **RSB** peut de plus faciliter la comparaison de mesures variant sur plusieurs ordres de grandeur. On définira donc ce rapport comme suit :

$$RSB = 10 \log \frac{S}{B} \quad (2.9)$$

Où S représente le signal et est donné par l'équation (2.10) et B le bruit défini comme l'erreur quadratique moyenne des écarts entre le profil de vitesse idéal et celui approximé avec le modèle Delta-Lognomal.

$$S = \int v_t^2(t) dt \quad (2.10)$$

2.1.4 Avantage de la lognormalité pour la modélisation de mouvements simples.

Deux grandes caractéristiques permettent de justifier l'emploi de la loi lognormale pour étudier la réponse impulsionnelle des mouvements rapides : les observations faites sur le profil de vitesse d'un mouvement rapide et le compromis vitesse précision impliquée dans la réalisation d'un tel mouvement.

L'utilisation de la loi lognormale pour représenter les composantes neuromusculaires du **SNM** repose en premier lieu sur les observations pertinentes que présente l'équation 2.1. C'est une fonction asymétrique, comportant la notion d'agoniste-antagoniste et formée d'un grand nombre de systèmes simples. La grande variété de courbes de mouvements rapides évaluées [25] [8] [26] dont les observations se rapprochent des caractéristiques de cette équation montre qu'elle est la mieux adaptée pour l'étude des mouvements rapides.

Pareillement, le profil de vitesse de pointage ressemble aux réponses impulsionales positives et non oscillantes des systèmes linéaires. Les caractéristiques telles que :

- La forme en cloche des enveloppes des signaux d'électromyographies (EMG) enregistrés au niveau des muscles lors de leurs contractions, [27];
- L'hypothèse de l'existence d'une relation de proportionnalité entre les temps de délai et mesurés à la sortie de chaque sous-système qui compose le **SNM**;
- La preuve mathématique formelle de la convergence vers un profil Lognormal de la réponse impulsionale du **SNM** présentée dans Plamondon et Al [14] ;

Justifient le choix d'une modélisation du **SNM** par une réponse impulsionale Lognormale.

D'autre part mais toujours dans le même ordre d'idée, la loi de Fitts sur le compromis vitesse-précision lors de l'émission d'un mouvement rapide volontaire de pointage permet la prédiction de la durée moyenne du mouvement en fonction du temps. Son équation caractéristique donnée par :

$$MT = a + b \log_2 \frac{2A}{W} \quad (2.11)$$

Où (MT) représente la durée moyenne d'une tâche de pointage réalisée à vitesse maximale, (A) l'amplitude de la tâche de pointage et (W) la largeur de la cible stipule que lors d'un mouvement de pointage d'un sujet allant d'un point initial vers une cible définie, si le sujet doit pointer le plus rapidement avec la plus grande précision possible vers la cible, la rapidité d'exécution se fait au détriment de la précision. Inversement, plus un mouvement devient précis, plus sa vitesse de réalisation diminue. En faisant une correspondance avec le modèle Delta-Lognormal, le paramètre (W) correspond au double de l'erreur relative dans le cadre de la Théorie Cinématique des Mouvements Rapides et la distance entre la zone cible et la zone de départ représentée par A qui correspond à la valeur D dans le cadre de la Théorie cinématique. D'autres formulations peuvent être trouvées dans [28] [29] [30]. Une grande variété de travaux portant sur les mouvements impliqués dans ce type de tâches rapportent comment le compromis vitesse-précision se comporte dans différentes conditions telles qu'en présence d'obstacles [31] [32], avec ou sans rétroaction visuelle [33], selon la géométrie de la zone cible [34], en présence d'illusion visuelle [35] [36] [37] et avec des cibles mobiles [38]. Différents aspects du contrôle moteur, comme l'apprentissage [39] et la présence de métacognition [40] ont aussi été évalués par ce paradigme. Cet ensemble de caractéristiques que présentent les mouvements impliqués dans un compromis vitesse-précision en

plus qu'ils soient des mouvements de pointage rapide se prêtent bien à une modélisation Delta-Lognormale. Différents tests ont été menés dans cette optique en utilisant une famille complète de modèles lognormaux et ont été proposées par Plamondon & Djoua [22] pour la représentation d'un trait de crayon.

2.1.5 Synthèse des travaux sur les mouvements humains basée sur la théorie cinématique des mouvements rapides et les modèles lognormaux qu'elle propose.

En résumé, cette section présente de façon sommaire l'état de l'art sur les modèles Lognormaux. Elle détaille les outils de modélisation qui seront utilisés pour représenter les mouvements à analyser dans ce projet. Elle étale dans un premier temps les raisons justifiant l'adoption des modèles Lognormaux pour la modélisation des mouvements de pointages rapides ainsi que la robustesse des équations caractéristiques. Finalement, la mesure qui est utilisée pour comparer la qualité de la modélisation des données expérimentales est définie et son choix justifié.

Au regard de ce qui est dit ci-dessus, l'ensemble des mouvements considéré tout au long de cette étude sont des mouvements de pointage rapides où l'on considère l'interaction synergétique de deux groupes neuromusculaires. Le choix du modèle Delta-Lognormal dans le cadre des investigations à réaliser est basé sur les besoins suivants: le modèle devrait:

- Être adapté à l'étude chez l'humain et faire intervenir les mouvements des membres supérieurs. Les modèles spécialisés dans l'analyse de la posture, de la locomotion ou des mouvements de corps non humains ne seront pas utiles.
- Permettre l'analyse cinématique des mouvements.
- Être assez polyvalent pour accommoder des mouvements complexes tels ceux associés à l'écriture manuscrite.
- Être suffisant pour réaliser des analyses efficientes de mouvements simples provenant de tâches de pointage.
- Permettre la représentation automatique de mouvements humains réels. C'est donc dire que les paramètres nécessaires à la reproduction des mouvements provenant de participants répondant à des tests neuromoteurs doivent pouvoir être extraits de façon non interactive.

L'étude des mouvements de pointage rapides nécessite des extracteurs de paramètres lognormaux des profils de vitesse des mouvements rapides. Ce sujet est abordé à la prochaine sous-section. On y retrouve deux grands groupes d'extracteurs, ceux appartenant au modèle Delta-Lognormal et ceux appartenant à la modélisation Sigma-Lognormale.

2.2 Généralités sur l'arithmétique par intervalles

Cette section résume les concepts de bases de l'arithmétique par intervalles (**AI**) nécessaires à la compréhension des techniques proposées dans la suite de la thèse, spécialement dans les chapitres 4 et 5 où ils sont utilisés pour représenter les variables de la fonction de coût à optimiser. La section se répartit sur trois sous-sections comme suit : la sous-section un (1) présente différents concepts et définitions d'utilités générales sur les intervalles. La sous-section deux (2) présente et illustre les opérations en **AI** pour des fonctions arithmétiques (+, -, \times) de la fonction de coût à optimiser. La sous-section trois met en lumière le phénomène de dépendance lié au calcul par intervalle.

2.2.1 Concepts et termes de bases

Le but de l'analyse d'intervalle est de fournir des limites supérieures et inférieures sur l'effet des différentes sources d'erreurs et d'incertitudes y compris les erreurs d'approximation et de collecte des données de capteurs numériques sur une quantité calculée. Avant de présenter les différents concepts qui utilisent l'arithmétique par intervalles, il est nécessaire d'établir une notation adaptée à la dérivation des équations à venir.

On note $I = [a, b] = \{x \in \mathbb{R} : a \leq x \leq b\}$ l'ensemble des nombres réels compris dans un intervalle fermé unidimensionnel. Un intervalle est nommé par une lettre majuscule et l'ensemble des valeurs contenues dans l'intervalle est désigné en minuscule. Ainsi,

$$X = [\underline{x}, \bar{x}] \quad (2.12)$$

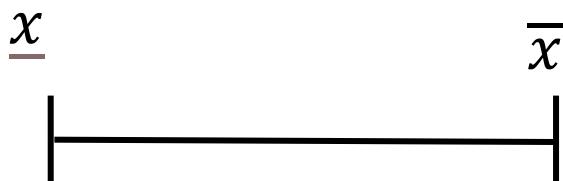


Figure 2.3 Intervalle de la variable X

est une variable d'intervalle pour laquelle \underline{x} et \bar{x} désignent les valeurs d'extrémités gauche et droite de X respectivement. La largeur de l'intervalle X est définie et noté comme suit :

$$w(X) = \bar{x} - \underline{x} \quad (2.13)$$

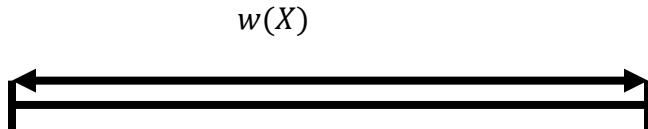


Figure 2.4 Largeur d'un intervalle

Le point milieu de X est donné par :

$$m(X) = \frac{1}{2}(\bar{x} + \underline{x}) \quad (2.14)$$

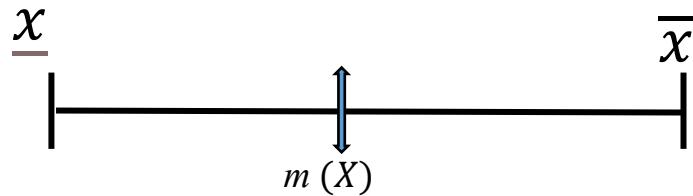


Figure 2.5 Milieu d'un intervalle matérialisé par le point $m(x)$

Le nombre réel x est équivalent à $x = [x, x]$ avec pour largeur zéro.

Lorsqu'un nombre \bar{x} ou \underline{x} n'est pas représentable sur ordinateur, on utilise l'arrondi en fixant la valeur de \bar{x} à sa valeur la plus petite représentable sur une machine et la valeur \underline{x} à la plus grande représentable sur un ordinateur de sorte que l'intervalle finale contient l'intervalle $[\underline{x}, \bar{x}]$.

Exemple 1: l'intervalle= $0.666 = [0.666, 0.666] = [0.665, 0.667]$.

$$X \cap Y = \left[\max(\underline{x}, \underline{y}), \min(\bar{x}, \bar{y}) \right] \quad (2.15)$$

Et leur union est donnée par :

$$X \cup Y = \left[\min(\underline{x}, \underline{y}), \max(\bar{x}, \bar{y}) \right] \quad (2.16)$$

L'un de nos objectifs est de développer un algorithme d'intervalle pratique capable de produire des bornes nettes (ou presque nettes) sur la solution du problème d'optimisation de la Delta-

Lognormale par un algorithme de type **S&E**. L'intersection est particulièrement une opération très importante dans ce type d'algorithme, car si nous avons deux intervalles contenant un résultat d'intérêt, leur intersection ne peut que donner un intervalle plus petit qui contient également le résultat.

Exemple 2 : prenons $X = [2, 6]$ et $Y = [4, 25]$ alors $X \cap Y = [4, 6]$ qui est l'intervalle contenu dans X et Y . Le cas de l'union pour X et Y donne $X \cup Y = [2, 25]$.

Nous remarquons dans le cas de l'union que tout sous espace borné peut être réuni en un grand ensemble. Cette remarque est d'autant plus importante dans un algorithme de **S&E** car l'espace de recherche des solutions est l'union des différents cas possibles.

1.2.1.1 Vecteur d'intervalles (Boite) et découpages des boites

Nous désignons par $\mathbf{X}^I = (X_1, X_2, \dots, X_n)$ l'ensemble de n intervalles fermés ou vecteurs d'intervalles qui définissent un hyperespace communément appelé boite. La notation des vecteurs d'intervalles se fait par une lettre majuscule en gras suivie d'un I en apostrophe en gras aussi.

Ainsi, une boite sera notée comme suit :

$$\mathbf{X}^I = (X_1, X_2, \dots) = ([\underline{x}_1, \bar{x}_1], [\underline{x}_2, \bar{x}_2], \dots) \quad (2.17)$$

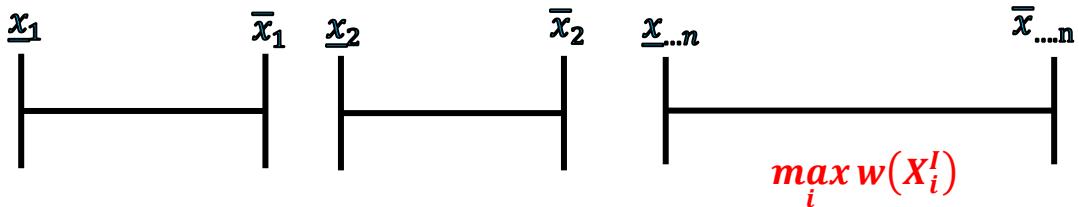


Figure 2.6 Vecteur d'intervalles de dimension n . Largeur de la boite ($W(x)$)

La largeur et le milieu d'un vecteur d'intervalle se définissent et se note respectivement comme suit:

$$w(\mathbf{X}^I) = \max_{i..N} w(X_i^I)$$

et

$$m(\mathbf{X}^I) = (m(X_1), m(X_2), \dots, m(X_n)) \quad (2.18)$$

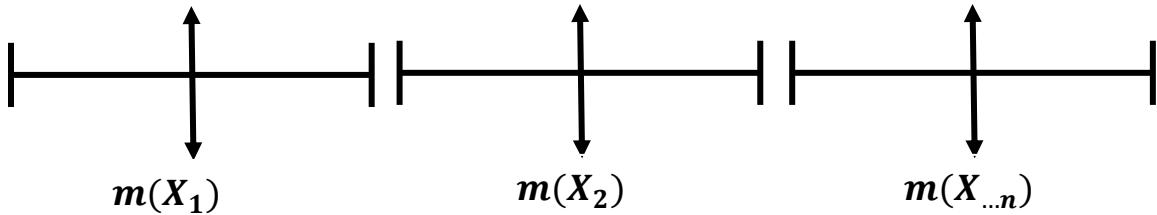


Figure 2.7 Milieu d'une boîte

La division d'un vecteur d'intervalles ou boîte X^I de dimension K est l'opération qui consiste à partitionner un intervalle ou plusieurs intervalles de l'ensemble K en N sous intervalles de sorte que la réunion des sous boîtes $X_1 \cup X_2 \cup \dots$ soit égales à l'ensemble de départ X^I . Exemple l'intervalle $X = [0,1]$ peut être divisé en n sous intervalles et donner $X_i = \left[\frac{(i-1)}{n} ; \frac{i}{n} \right]$ pour $1 \leq i \leq n$. La figure 2.8 illustre le cas du découpage de la boîte de l'équation 2.18 en deux sous boîtes.

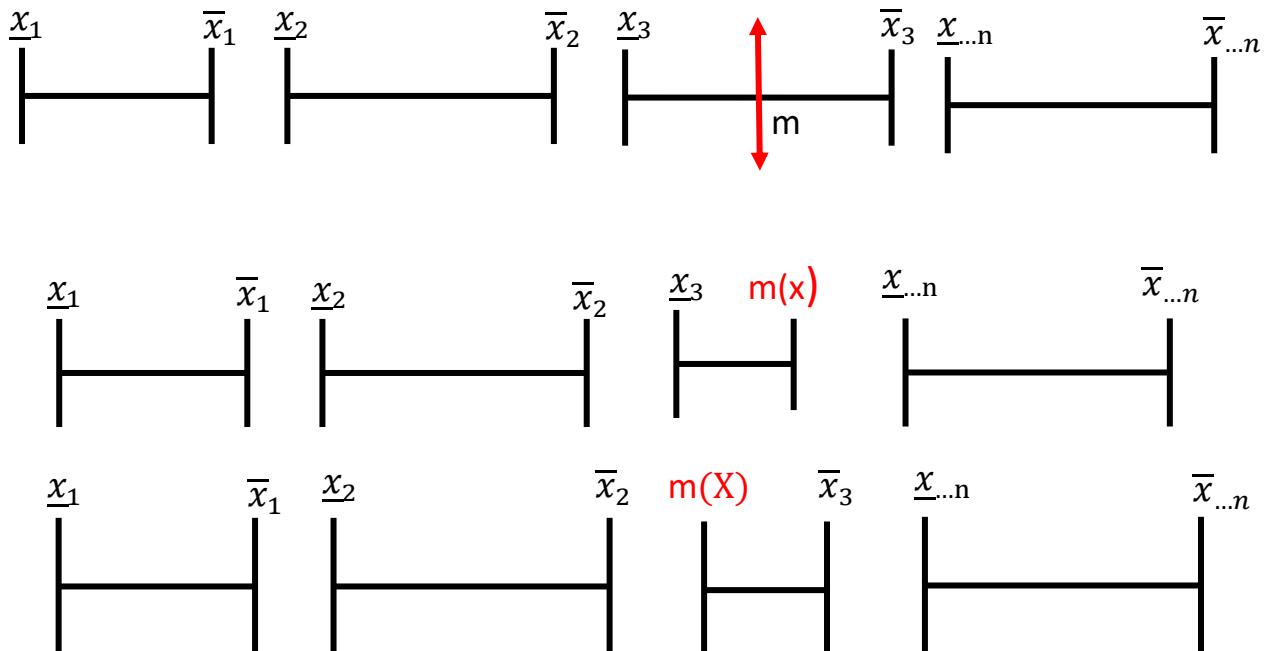


Figure 2.8 Exemple de division d'une boîte en deux sous boîtes

2.2.2 Opérations arithmétiques

Les opérations arithmétiques entre les intervalles reposent sur l'idée principale que le calcul par intervalle est une opération faite sur des ensembles, alors le résultat est un intervalle qui regroupe l'ensemble des valeurs prises par chacune des opérandes à travers l'opération. Prenons l'exemple de l'addition de deux intervalles, l'intervalle résultant est un ensemble contenant les sommes de toutes les paires de nombres des deux intervalles initiaux. Suivant cet ordre d'idée, l'ensemble des opérations arithmétiques de bases nécessaires à la réalisation de l'algorithme **S&E** proposé au chapitre 5 de cette thèse sont :

Soit $X = [\underline{x}, \bar{x}]$ et $Y = [\underline{y}, \bar{y}]$ deux intervalles avec $x, y \in \mathbb{R}$ on a :

$$X + Y = [\underline{x} + \underline{y}, \bar{x} + \bar{y}] \quad (2.19)$$

Car pour tout $x \in X \rightarrow \underline{x} \leq x \leq \bar{x}$ et pour tout $y \in Y \rightarrow \underline{y} \leq y \leq \bar{y}$ on remarque que en additionnant membre à membre les deux inégalités, la somme $x + y$ doit satisfaire l'équation 4.8. Un raisonnement analogue nous permet de déduire la soustraction d'ensemble dans un intervalle :

$$X - Y = [\underline{x} - \bar{y}, \bar{x} - \underline{y}] \quad (2.20)$$

Avec $X - Y = X + (-Y)$ pour $(-Y) = [-\bar{y}, -\underline{y}]$

Le cas de la multiplication et la division sont donnés par :

$$X \times Y = [\min S, \max S] \text{ avec } S = \{\underline{x}\underline{y}, \underline{x}\bar{y}, \bar{x}\underline{y}, \bar{x}\bar{y}\} \quad (2.21)$$

Et

$$X \div Y = X \times \left(\frac{1}{Y}\right), \text{ avec } \left(\frac{1}{Y}\right) = \left[\frac{1}{\bar{y}}, \frac{1}{\underline{y}}\right] \text{ and } 0 \notin Y \quad (2.22)$$

Les intervalles d'ensembles pour le minimum et maximum sont respectivement donnés par :

$$\min(X, Y) = [\min(\underline{x}, \underline{y}), \min(\bar{x}, \bar{y})] \quad (2.23)$$

$$\max(X, Y) = [\max(\underline{x}, \underline{y}), \max(\bar{x}, \bar{y})] \quad (2.24)$$

Fixons les idées avec l'exemple suivant : soit $X = [1, 5]$ et $Y = [2, 7]$,

$$X + Y = [3, 12] ; X - Y = [-6, 3]; X \div Y = [0.143, 2.5] ; X \times Y = [2, 35]; \min(X, Y) = [1, 5]; \max(X - Y) = [2, 7]$$

2.2.3 La Dépendances

La dépendance est la surestimation engendrée par le calcul par intervalle de certaines opérations arithmétiques. En effet lors de l'évaluation d'une fonction ayant plusieurs variables réparties dans son expression (la mise en facteur n'étant pas possible), chaque apparition d'une variable dans une opération est considérée comme une nouvelle variable. En d'autres mots, l'opération $X - X$ est calculée comme $X - Y$ avec Y numériquement égal à X mais indépendant de X . Différentes opérations en arithmétiques par intervalles présentent ce phénomène, c'est le cas notamment de la multiplication ou l'opération $X_1^2 \neq X_1 \times X_1$, la soustraction avec $X_1 - X_2 \neq [0, 0]$ pour X_1 numériquement égale à X_2 et de la division où l'opération $\frac{X}{X}$ n'est pas égale à 1.

Exemple 3: soit $X_1 = [-1, 3]; X_2 = [2, 7]$ on a :

$$X - X = [-4, 4] \neq [0, 0] ; X_1^2 = [0.0, 9] \text{ et } X_1 \times X_1 = [-3, 9] \text{ on remarque que les résultats de } X_1^2 \neq X_1 \times X_1; \frac{X_2}{X_2} = [0.29, 3.5] \neq 1$$

On remarque à travers l'exemple 3 que pour deux expressions mathématiques équivalentes on obtient un résultat global plus net dans un cas et plus large dans l'autre. Cette surestimation due aux phénomènes de dépendance peut être néfastes (augmentations des délais d'exécution) dans un algorithme de type **S&E** car l'objectif dans ce type d'algorithme est souvent de réduire l'espace de recherche afin d'atteindre l'optimum. Néanmoins, les résultats bien que surestimés englobe toujours le résultat net. Dans une séquence d'opération, l'évaluation de chaque étape de la séquence se fait en utilisant les différents modes d'arrondis selon la norme IEE-754.

2.3 L'analyse Lognormale des mouvements humains à partir des données provenant de mouvements d'écritures manuscrites.

Différents algorithmes sont proposés dans la littérature pour résoudre le problème d'extraction des paramètres Lognormaux. Cette section les regroupe en deux grandes catégories :

- Les algorithmes d'extraction des paramètres Sigma-Lognormaux dans lesquels l'extraction porte sur les mouvements complexes de pointage en deux dimensions.

- Les algorithmes d'extraction des paramètres Delta-Lognormaux où il est question d'extraire les valeurs des paramètres de mouvements de pointage unidimensionnels simples et rapides tels des traits d'écriture manuscrite.

Ensuite une évaluation des différents extracteurs Delta-Lognormaux actuel est effectué et les objectifs sont dégagés.

2.3.1 Extracteur Sigma-lognormaux

Un extracteur de paramètres sigma-lognormaux a été conçu pour analyser les mouvements complexes tels que les signatures et les mots. IL est basé sur l'algorithme **XOR** [22] , [21] qui signifie XZéro-robuste. L'algorithme **XOR** est une amélioration de **Xzero** appliquée au modèle de somme d'une suite de Lognormales. Nous commençons dans la prochaine sous-section par décrire l'extracteur **XOR**, ensuite nous l'évaluons en regard de ses performances sur des mouvements complexes, et enfin nous concluons avec les améliorations.

1.2.1.2 XOR

Les travaux menant à la réalisation de l'algorithme Xzéro Robuste (**XOR**) ont pour but d'appliquer le cas sommes des lognormales ($\Sigma\Lambda$) à l'algorithme Xzéro [22] développé pour l'extraction delta-lognormale. Il a alors été constaté que Xzéro ne produisait pas toujours des solutions assez robustes lorsqu'utilisé dans un cadre $\Sigma\Lambda$ sur des données réelles. L'algorithme **XOR** a donc été développé en partie pour obtenir des solutions plus robustes que celles produites par Xzéro et, contrairement à Xzéro, **XOR** doit prendre en compte un nombre arbitraire de composantes lognormales pour pouvoir estimer la valeur de ses paramètres angulaires. L'idée générale repose sur le concept d'estimation et optimisation initié par Wise. Le module estimation repose sur les approximations mathématiques de l'algorithme **Xzero** [22] et des solutions de l'extracteur **IIX**.

L'extracteur opère selon deux modes. Il démarre dans un mode séquentiel dans lequel il extrait les paramètres lognormaux dans l'ordre temporel d'apparition des composantes agoniste, antagoniste du mouvement. La fin d'une extraction est marquée par l'obtention d'un rapport signal sur bruit d'au moins 25 DB sur l'ensemble du mouvement de base. Généralement, ce premier parcours sur les pics significatifs du signal dans sa reconstruction donne des résultats satisfaisants ($SNR > 25$ DB). Dans le cas contraire où l'algorithme a atteint la fin du signal sans produire un **RSB** de reconstruction moins élevée que le seuil de 25 dB, l'extracteur entre dans un second mode. Il

identifie et extrait alors les composantes lognormales restantes dans le signal de base, par ordre décroissant de l'importance de leur effet sur le mouvement. L'ordre ici est qualifié par la grandeur de l'aire sous les différentes courbes de vitesses restantes.

La phase d'identification des points caractéristiques consiste à localiser sur le profil de vitesse du mouvement cinq points caractéristiques qui sont : le début de la composante (son occurrence au temps t_0), le premier point d'inflexion, le maximum local, le second point d'inflexion et la fin de la composante. Une série de profils de vitesses dans le mouvement possédant ces points caractéristiques est alors identifiée. Les profils de vitesses retenues sont fournis à l'estimateur **Xzero** pour le calcul des paramètres (t_0, μ, σ, D) représentatif d'une composante neuromusculaire. Les paramètres θ_f et θ_d sont par la suite estimés en utilisant l'équation (5). Les détails de la méthode se trouvent dans [21]. Une fois les paramètres cinématiques de toutes les séries obtenues par **Xzero**, s'ensuit l'extraction des paramètres angulaires $\theta_f; \theta_d$. Le modèle Sigma-Lognormal considère que les composantes neuromusculaires forment une trajectoire en arc de cercle, par ce fait, l'angle d'une composante varie proportionnellement avec la distance parcourue. L'estimation des angles $\theta_f; \theta_d$ est donnée par les équations suivantes :

$$\varphi(t) \propto \int v_t dt$$

Pour une série variant de t_0 à t on a :

$$\nabla = \frac{d\varphi(t)}{dt} \text{ et } L = \int_{t_0}^t D\Lambda(t; t_0, \mu, \sigma^2) dt$$

Les valeurs des angles de début et fin sont alors données par:

$$\theta_d = \varphi(t) - \Delta L$$

$$\theta_f = \varphi(t) + \Delta(D - L)$$

Une interpolation linéaire entre chaque série de points caractéristiques d'un profil identifié est effectuée afin de reconstruire le mouvement par l'utilisation de la fonction Sigma-Lognormale et les paramètres associant chaque profil.

Notons qu'avant de procéder à l'extraction des séries de points, un prétraitement du signal est effectué afin d'avoir une estimation meilleure pour la reconstruction.

En résumé, l'algorithme **XOR** associé au processus de prétraitement et raffinement des solutions est l'extracteur des paramètres Sigma-Lognormaux. Le système débute par identifications et extraction par ordre croissant de leurs temps d'occurrence des séries de points de mouvements de bases contenues dans le mouvement d'ensemble. Puis un prétraitement suivi d'une estimation des paramètres cinématique en utilisant l'algorithme **XOR**. Une fois l'ensemble des séries traitées, alors qu'il ne reste plus de points caractéristiques dans le signal de base, le module d'optimisation non linéaire de **XOR** formé d'un algorithme de recherche par coordonnées se charge d'effectuer les ajustements nécessaires sur chaque série de points afin d'obtenir le seuil désiré.

1.2.1.3 Évaluation de XOR

Le système d'extraction **XOR** présenté ci-dessus a l'avantage d'être très polyvalent en ce sens qu'il peut être utilisé directement sur n'importe quelle trajectoire d'un mouvement de pointage humain, sans que les données aient à être prétraitées ou que des informations a priori sur la nature du mouvement ne soient nécessaires. Ceci est très utile pour différentes applications liées à la reconnaissance de formes telles que le développement d'algorithme de reconnaissance de l'écriture ou de vérification de signature. En fait, ce type d'algorithme est essentiel dans tous les cas où le mouvement est laissé libre. À l'opposé, certaines applications contraignent fortement les mouvements à traiter. C'est le cas pour les domaines dans lesquels on fait passer des tests neuromoteurs ou des consignes spécifiques et bien précises sont à respecter pour l'accomplissement du mouvement. Ce genre de test est utilisé par exemple en psychologie expérimentale ou en génie biomédical pour le développement de tests de diagnostic [41] [42]. Dans ce contexte, les données disponibles à l'analyse proviennent de mouvements stéréotypés. Bien que le système d'extraction **XOR** puisse être utilisé pour ce genre d'études (p. ex., il a été utilisé dans [21] [43], il y a au moins deux raisons pour lesquelles une autre approche puisse sembler avantageuse. Premièrement, on dispose d'une connaissance à l'avance du type de mouvement réalisé. Il est donc possible de mettre à profit cette connaissance supplémentaire pour obtenir des résultats d'extraction supérieurs. Deuxièmement, le type d'application utilisant ces tests nécessite généralement un type d'analyse statistique qui peut être difficile de réaliser sur les paramètres extraits à l'aide du système décrit précédemment puisque celui-ci ne peut pas assurer une extraction ayant un haut degré de cohérence.

1.2.1.4 Réunification de XOR ET IIX

Bien que l'extracteur **XOR** affiche plusieurs caractéristiques désirables, les paramètres qu'il fournit peuvent être difficiles à analyser statistiquement quand ceux-ci proviennent d'un échantillon de plusieurs mouvements stéréotypés (p. ex. un ensemble de mouvements oscillatoires, de triangles, de carrés, etc.). Ceci découle du fait que chacun des mouvements reconstruits peut alors être décrit par des plans d'action très différents comportant un nombre variable de paramètres. Le problème visant à établir une correspondance entre les paramètres provenant des différents mouvements est alors épique et les techniques statistiques semblent être peu utiles. Pour répondre à ce problème, un algorithme d'extraction basé sur l'utilisation d'un prototype de mouvement a été élaboré. Cet algorithme est la réunion de l'ensemble de tous les extracteurs lognormaux ayant vu le jour. Le schéma bloc de ce système est présenté à la figure 2.9.

Il réunit en un seul bloc la robustesse de la conception **IIX** pour les mouvements Delta-Lognormaux et l'efficacité vectorielle de l'extracteur **XOR** pour les modèles Sigma-Lognormaux. Pour extraire les paramètres lognormaux associés aux tâches offrant un compromis vitesse-précision, le système **IIX** [22] est utilisé en parallèle avec un extracteur appliquant le principe d'estimation **XOR** au cas delta-lognormal. La solution produisant le meilleur **RSB** d'ajustement est conservée et optimisée en utilisant l'algorithme de Marquadt [44] sur les coordonnées. Ce processus est représenté à la Figure 2.9.

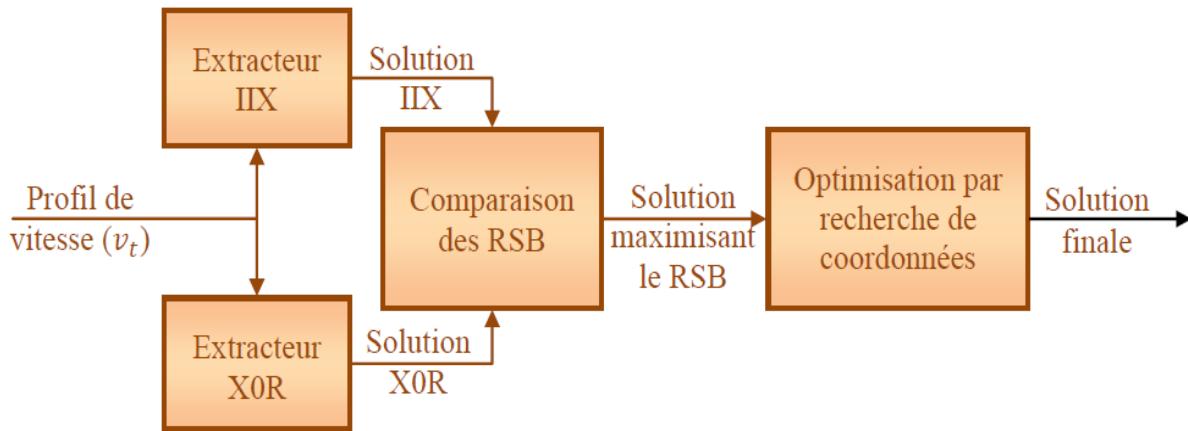


Figure 2.9 Schéma du système d'extraction des paramètres Lognormaux

2.3.2 Extracteurs Delta-Lognormaux

Les méthodes d'extraction actuelles peuvent être regroupées en quatre grandes catégories : statistique, déterministe, évolutive et séparation & évaluation. Les méthodes statistiques reposent sur des algorithmes génériques d'estimation de paramètres, qui utilisent les fonctions de densités de probabilités de variables aléatoires lognormales [45] [46]. Les méthodes déterministes se concentrent dans un premier temps sur des points particuliers de la courbe de vitesse pour estimer les paramètres, puis elles utilisent des algorithmes d'optimisation pour ajuster les valeurs estimées afin de converger vers des solutions idéales [47] [48]. Les méthodes évolutionnistes utilisent quant à elle, des algorithmes génétiques pour explorer l'espace des paramètres [49]. La méthode de [50] utilise une approche géométrique pour construire l'espace de recherche des solutions potentielles puis se sert des propriétés de la fonction de coût pour énumérer et élaguer les solutions non prometteuses. Des études comparatives entre les approches déterministes, évolutives et statistiques ont montré que les méthodes déterministes sont plus efficaces [49]. L'ajustement des courbes de vitesse simples en utilisant le modèle Delta-Lognormal débute par un premier algorithme nommé **INFLEX** proposé par [48]. Dans le but de l'améliorer, l'algorithme **INITRI** a par la suite été conçu [47]. Toujours dans un souci d'amélioration, l'algorithme **Xzero** a finalement été conçu et intégré dans un système d'extraction nommé **IIX** qui utilise ces trois estimateurs en parallèle [22].

Nous commençons dans la prochaine sous-section par décrire les extracteurs (**INFLEX**, **INITRI**, **XZERO** **IIX** et **S&E**) ensuite nous les évaluerons en regard leurs performances sur des mouvements de pointages simples à une dimension, et enfin nous concluons avec les améliorations.

1.2.1.5 Algorithme INFLEX

L'algorithme proposé par Guerfali et Li [48] est la base de la majorité des algorithmes proposés pour la résolution du problème d'extraction des paramètres lognormaux. Il est constitué d'un module d'estimation et d'un module d'optimisation non linéaire de Levenberg Marquardt [44]. Le module d'estimation repose sur une approximation des valeurs des paramètres Lognormaux à partir de certains points caractéristiques du profil de vitesse proposé par Wise [51]. Dans son algorithme d'estimation des paramètres d'un mouvement simple (profil de vitesse Delta-Lognormal), Wise a développé une méthode graphique qui lie les paramètres aux caractéristiques globales du profil de

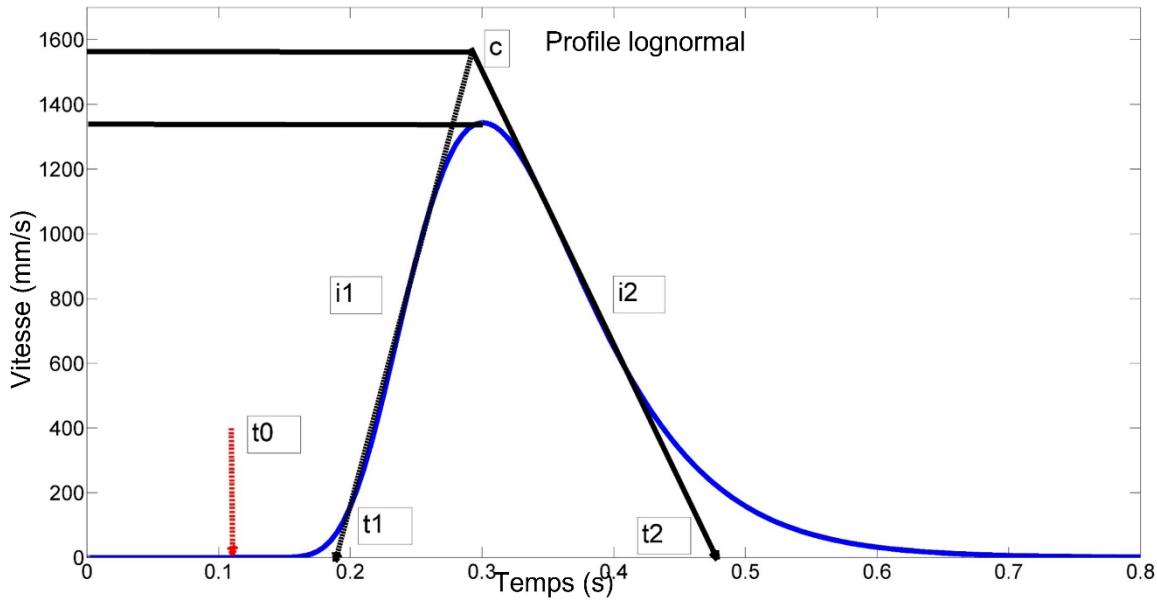


Figure 2.10 Caractéristiques globales utilisées par le modèle de Wise. (I1, I2 pente des tangentes aux points d'inflexion). (C, t1, t2) triangle permettant d'approximer une composante neuromusculaire

vitesse du mouvement de pointage à évaluer. Les points particuliers choisis sont : les pentes des tangentes aux points d'inflexion et le point d'amplitude maximale du profil. La figure 2.10 résume les points utilisés dans cet algorithme. On peut remarquer que la figure géométrique (triangle) formée par les points c, t1 et t2 permettant d'établir un ensemble d'équations qui conditionnent la portion de la courbe incluse dans le triangle.

L'estimation commence par le calcul du paramètre sigma du profil de vitesse considérée sous l'hypothèse que le mouvement dans son ensemble représente une lognormal qui est la composante agoniste. Des équations basées sur les points d'inflexion représentés par I1 et I2 sur la figure 2.10 permettent de calculer le paramètre (σ). Ensuite, on calcule les valeurs des paramètres t_0, μ , à l'aide des équations proposées par Wise [51]. Cette approche permet d'estimer les quatre paramètres d'une Delta-Lognormale en considérant que la composante antagoniste devient significative seulement après le maximum du pic principal du profil de vitesse. La composante antagoniste est alors déduite de la soustraction entre le mouvement et la composante agoniste reconstruite par échantillonnage du temps et évaluation de l'équation 2.1 pour une lognormale avec les paramètres préalablement calculés. La vitesse résultante de la soustraction est attribuée à la contribution de la composante antagoniste et les mêmes étapes du processus d'extraction graphique se répètent.

La démarche graphique de Wise associée aux points caractéristiques de Guerfali est la pierre angulaire des développements algorithmiques des extracteurs qui suivent. Son avantage est la simplicité de ses équations mathématiques associée à sa rigueur géométrique. Malgré les bonnes performances de cet extracteur pour certains profils de vitesse de pointages, il ne pouvait malheureusement pas répondre à la panoplie de profils de vitesse lognormaux. Sa principale lacune est l'identification de l'agoniste par rapport à l'antagoniste afin d'estimer le mouvement. De plus l'hypothèse du premier pic consacré à l'agoniste reste discutable. Cette méthode est difficilement adaptée pour les profils réels qui sont souvent accompagnés de distorsions. Dans ces cas, de petites erreurs d'estimation graphique des positions des points d'infexion sur le profil engendrent des erreurs non négligeables dans le calcul des valeurs des tangentes en ces points. Ce qui entraîne des résultats non conformes au modèle Delta-Lognormal. Dans le cas d'une faible asymétrie du profil de vitesse, cet algorithme peut donner une valeur négative du paramètre t_0 , qui représente le temps d'occurrence dont la valeur est positive.

1.2.1.6 Algorithme INITRI

L'algorithme **INITRI** est dû à Plamondon et Li, il est conçu pour renforcer la robustesse du module estimation de l'algorithme de Wise [51]. Il repose sur des relations entre certaines caractéristiques du profil de vitesse Delta-Lognormal tel que proposé dans [47]. Dans la méthode analytique développée, on considère que l'activité du système antagoniste devient significative après que le mouvement a atteint sa vitesse maximale à l'instant t_m . Dans ce cas, la portion du profil de vitesse située en amont du maximum est exclusivement constituée de sa composante agoniste. Les points caractéristiques sont arbitrairement choisis de sorte que deux points du profil d'indices temporels t_i et t_j soient alignés et respectent la condition $t_0 < t_i, t_j < t_m$ (figure 2.11). À partir de ces trois quantités temporelles (t_i, t_j, t_m) , des relations mathématiques sur la fonction Delta-Lognormale permettent d'estimer les quatre paramètres d'une composante.

La dérivation des équations caractéristiques et les détails de leurs implémentations peuvent être trouvés dans [47]. **INITRI**, malgré le fait qu'il complète **INFLEX**, présente le désavantage de sous-estimer l'information liée aux paramètres σ_1 qui représente le coefficient d'asymétrie de la lognormale. L'algorithme d'estimation considère que la composante antagoniste n'est significative qu'en aval du maximum du profil de vitesse. Ceci limite aussi ses possibilités

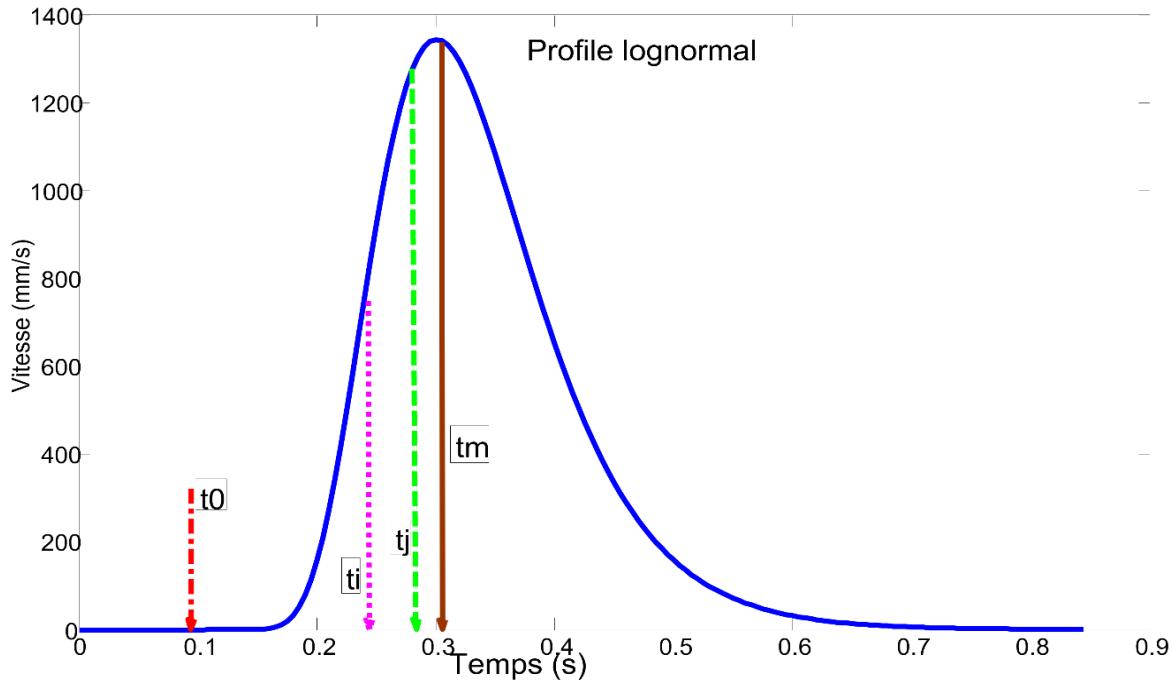


Figure 2.11 Points caractéristiques choisis (t_0, t_i, t_j, t_m) pour l'algorithme **INITRI**

d'extraction en ce qui a trait à l'ensemble des profils de vitesses lognormaux de mouvements rapides existants dans la base de données.

1.2.1.7 Algorithme Xzero et IIX

L'algorithme **Xzero** est dû à [22]. Tout comme **INFLEX** et **INITRI**, il consiste à établir des relations analytiques à partir de certains points caractéristiques du profil de vitesse d'un mouvement de pointage. Les points choisis sont les extrêmes et les deux points d'inflexion du profil de vitesse. La recherche des passages par zéro de la première et de la seconde dérivée du mouvement de vitesse est ainsi mise à l'épreuve. De façon générale, il fusionne les deux idées maîtresses des algorithmes **INFLEX** et **INITRI**. Le problème considéré est la dépendance des deux précédents algorithmes dans la détermination de la position de l'agoniste par rapport à celle de l'antagoniste dans un profil de vitesse afin de procéder aux estimations. Cette méthode permet d'offrir un estimateur fiable indépendamment de la position de la composante antagoniste par rapport à l'agoniste. Elle évite la sous-estimation du paramètre σ_1 (dû à **INITRI**) et le mauvais choix d'une tangente quand vient le temps d'une modélisation graphique (cas **INFLEX**). Les détails mathématiques sont résumés dans [22].

L'algorithme commence par une estimation de la composante agoniste du profil de vitesse Delta-Lognormal en prenant pour hypothèse que cette composante est la plus significative dans le mouvement de vitesse et que le pic principal lui appartient. Ensuite, des équations basées sur les points d'inflexions, maximum et passage par zéro du profil de vitesse permettent de calculer les différents paramètres de la composante agoniste. De façons pratique, les temps t_{max} , t_{Inf1} et t_{Inf2} , qui sont respectivement le temps d'occurrence du maximum, du premier et deuxième point d'inflexion sont déterminés sur le profil de vitesse; t_{max} se calcule en annulant la dérivée première de la fonction lognormale, les deux autres points (t_{Inf1} et t_{Inf2}) se déterminent en calculant les zéros de la dérivée seconde de la fonction lognormale. Les équations ci-dessous présentent les résultats des expressions pour les différents temps (t_{max} , t_{Inf1} et t_{Inf2} ,):

$$t_{max} = t_0 + e^{\mu - \sigma^2}$$

$$t_{Inf1} = t_0 + e^{\mu - \sigma \left(\frac{3\sigma + \sqrt{\sigma^2 + 4}}{2} \right)}$$

$$t_{Inf2} = t_0 + e^{\mu - \sigma \left(\frac{3\sigma - \sqrt{\sigma^2 + 4}}{2} \right)}$$

Ces trois expressions sont obtenues sous l'hypothèse que $t_{max} < t_{Inf1} < t_{Inf2}$, car sur un profil lognormal ces différents temps (t_{max} , t_{Inf1} et t_{Inf2} ,) se présentent dans cet ordre. La résolution de ces trois équations en remplaçant les valeurs des différents temps (t_{max} , t_{Inf1} et t_{Inf2} ,) identifiés sur le profil de vitesse permettent d'obtenir les valeurs de t_0 , μ , et σ pour une lognormale. Plutôt que de procéder à la détermination directe de ces valeurs, des intervalles de confiances autour de ces valeurs sont construites afin d'éviter des approximations plus ou moins éloignées. Le premier intervalle est construit sur le paramètre sigma d'une lognormale, car c'est lui qui fait défaut dans les deux précédents algorithmes. La détermination de l'intervalle $[\underline{\sigma}; \bar{\sigma}]$ est basée sur l'intervalle de temps qui couvre 99.97% de la surface totale recouverte par une courbe lognormale. Cet intervalle est donné par l'expression:

$$I = [e^{\mu - 3\sigma}, e^{\mu + 3\sigma}] \quad (2.25)$$

Posons $t_{min} = t_0 + e^{\mu - 3\sigma}$ et $t_{max} = t_0 + e^{\mu + 3\sigma}$;

$$t_{max} - t_{min} = e^{\mu + 3\sigma} - e^{\mu - 3\sigma} = \Delta t \quad (2.26)$$

En faisant la soustraction entre le temps du maximum et le temps t_{min} on obtient la relation suivante :

$$F(\sigma) = \frac{\sinh 3\sigma}{e^{3\sigma} - e^{\sigma^2}} - \frac{t_{max} - t_{min}}{2(t_{min} - t_{maximum})}$$

Qui est fonction du paramètre σ . Pour t_{max} et t_{min} se trouvant par exemple à 5% de la valeur du $t_{maximum}$ (temps d'occurrence de V_{max}) de part et d'autre du mode, nous obtenons une estimation de σ_{inf} et pour $t_{min} = 0$ et $t_{max} = T_{max}$ = période d'échantillonnage maximale du signale, on obtient une borne supérieure σ_{sup} de σ .

Les valeurs estimées d'intervalles du paramètre μ sont obtenues de façon analogue en remplaçant dans t_{inf1} et t_{inf2} les valeurs estimées de sigma $\hat{\sigma}$ prises dans l'intervalle des valeurs de sigma précédemment calculées. On obtient :

$$t_{inf2} - t_{inf1} = Ae^{\hat{\mu} - \hat{\sigma}^2} \quad (2.27)$$

Avec A :

$$A = e^{-\hat{\sigma}\left(\frac{\hat{\sigma}-\sqrt{\hat{\sigma}^2+4}}{2}\right)} - e^{-\hat{\sigma}\left(\frac{\hat{\sigma}+\sqrt{\hat{\sigma}^2+4}}{2}\right)}$$

en résolvant A pour mu, on obtient :

$$\hat{\mu} = \hat{\sigma}^2 + \ln \frac{t_{inf2} - t_{inf1}}{A} \quad (2.28)$$

Le \hat{t}_0 s'obtient directement en remplaçant $\hat{\mu}$ et $\hat{\sigma}$ trouvés dans l'équation du mode (t_{max}). On obtient :

$$\hat{t}_0 = t_{max} + e^{\hat{\mu} - \hat{\sigma}^2} \quad (2.29)$$

Concernant l'amplitude D on l'estime en remplaçant dans l'équation de $D\Lambda(t: t_0, \mu, \sigma^2)$ le temps maximum et les valeurs approximées de $\hat{\mu}$ et $\hat{\sigma}$ \hat{t}_0 et en égalisant à la valeur de V_{max} sur la courbe, on obtient :

$$\hat{D} = V_{max} \hat{\sigma} \sqrt{2\pi} e^{\hat{\mu} - \frac{\hat{\sigma}^2}{2}} \quad (2.30)$$

Par analogie aux deux précédents systèmes, l'antagoniste est déduit par soustraction au mouvement principale les valeurs échantillonnées de l'agoniste.

L'algorithme **IIX** est le regroupement des extracteurs **INFLEX**, **INITRI**, et **XZERO** en un seul algorithme fonctionnant en parallèle. Ces trois approximations constituent le module d'approximation de **IIX**. La solution retenue pour l'optimisation par l'algorithme de Marquardt (qui constitue le module d'optimisation) sont les valeurs des paramètres d'un des trois algorithmes ayant obtenues la meilleure erreur quadratique au sens des moindres carrés entre la vitesse à reconstruire et son approximation par l'équation du delta-lognormale.

La réunification des trois algorithmes en un a pour objectif de mettre en œuvre la complémentation que présentent l'un pour l'autre. L'objectif est de réunir l'efficacité des trois extracteurs en un seul afin de garantir l'extraction de l'ensemble des profils de vitesses connus de notre base de données. Le système **IIX** constitue l'extracteur Delta-Lognormal le plus polyvalent qui existe. Il permet d'offrir un estimateur fiable indépendamment de la position de la composante antagoniste par rapport à l'agoniste.

1.2.1.8 Approche séparation et évaluation appliquée au modèle Delta-Lognormal

Des travaux visant à ajuster le profil de vitesse Delta-Lognormal par une approche par séparation et évaluation ont été initiés dans [50]. Dans cette publication, O'Reilly & Al proposent un extracteur des paramètres Delta-Lognormaux par séparation et évaluation dont la technique principale consiste à bâtir des enveloppes Delta-Lognormales pour ensuite effectuer la recherche de la meilleure enveloppe qui minimise l'erreur quadratique entre le signal et l'enveloppe la plus proche. Dans cet algorithme, l'enveloppe est conçue en déterminant les variations de la fonction lognormale par rapport à ces paramètres pour chaque composante neuromusculaire de l'équation

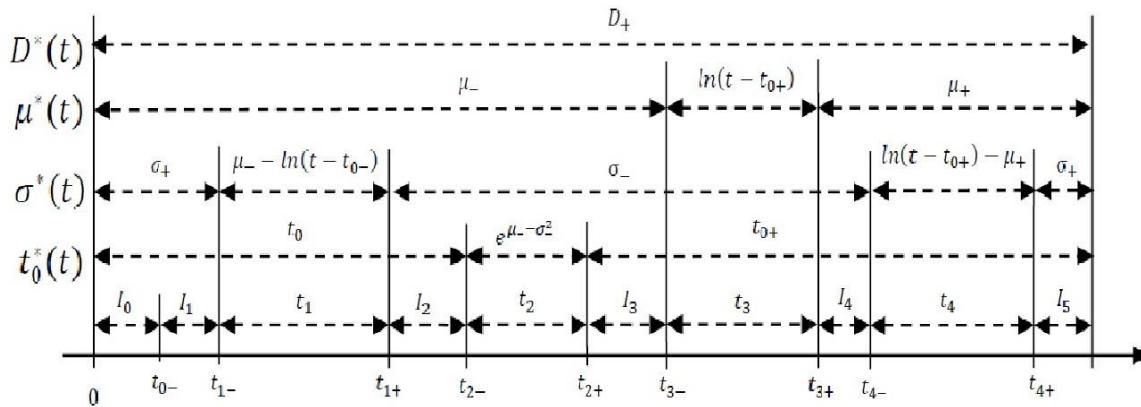


Figure 2.12 Variation des paramètres de la fonction lognormale en fonction du temps : les t_i sont des intervalles de temps définissant le domaine de validité des paramètres

de la Delta-Lognormale. Pour ce faire, les instants où le gradient de la fonction lognormale s'annule pour chaque paramètre sont calculés. Un système d'équations caractéristique des extrema sont établi pour toute la plage d'échantillonnage du profil de vitesse à étudier. L'ajustement de la courbe se fait en calculant l'erreur quadratique sur l'enveloppe construite. Les développements mathématiques menant à l'obtention des zéros du gradient peuvent être trouvés dans [50]. La figure 2.12 résume les intervalles de variations des paramètres pour une lognormale. Dans ce graphique, les instants t_i sont des intervalles de temps qui sont déterminés comme suit. Prenons le cas de t_1 par exemple, pour des valeurs des paramètres lognormaux (μ et σ), on a :

$$t_1 = t_{0-} + e^{\mu- - \sigma} = t_{0-} + [e^{\mu- - \sigma_+}, e^{\mu- - \sigma_-}] \quad (2.31)$$

Dans cet intervalle, la valeur du paramètre changeant (σ dans ce cas) peut être obtenue en isolant de l'équation 2.31 la valeur de σ :

$$t = t_{0-} + e^{\mu- - \sigma_-} \Leftrightarrow \sigma = \mu_- - \ln(t - t_{0-}) \quad (2.32)$$

En appliquant la même logique, on obtient l'expression du paramètre variant dans chaque intervalle t_i . À partir de ce diagramme, l'expression de l'enveloppe maximale recherchée et des développements similaires à 2.31 et 2.32 on obtient la borne supérieure pour une lognormale.

Un raisonnement analogue permet de déterminer la borne inférieure.

$$\boxed{\bar{\Lambda}} = \begin{cases} 0 & t \in I_0 = [0, t_{0-}] \\ \Lambda(t - t_{0-}; \mu_-, \sigma_+) & t \in I_1 = t_{0-} + [0, e^{\mu- - \sigma_+}] \\ \Lambda(t - t_{0-}; \mu_-, \sigma = \mu_- - \ln(t - t_{0-})) & t \in I_2 = t_{0-} + [e^{\mu- - \sigma_+}, e^{\mu- - \sigma_-}] \\ \Lambda(t - t_{0-}; \mu_-, \sigma_-) & t \in I_3 = t_{0-} + [e^{\mu- - \sigma_-}, e^{\mu- - \sigma_-^2}] \\ \Lambda(e^{\mu- - \sigma_-^2}; \mu_-, \sigma_-) & t \in I_4 = t_{0-} + [e^{\mu- - \sigma_-^2}, e^{\mu- - \sigma_-^2}] \\ \Lambda(t - t_{0+}; \mu_-, \sigma_-) & t \in I_5 = t_{0+} + [e^{\mu- - \sigma_-^2}, e^{\mu- - \sigma_-}] \\ \Lambda(t - t_{0+}; \mu = \ln(t - t_{0+}), \sigma_-) & t \in I_6 = t_{0+} + [e^{\mu- - \sigma_-}, e^{\mu- - \sigma_-}] \\ \Lambda(t - t_{0+}; \mu_+, \sigma_-) & t \in I_7 = t_{0+} + [e^{\mu- - \sigma_-}, e^{\mu+ + \sigma_-}] \\ \Lambda(t - t_{0+}; \mu_+, \sigma = \ln(t - t_{0+}) - \mu_+) & t \in I_8 = t_{0+} + [e^{\mu+ + \sigma_-}, e^{\mu+ + \sigma_+}] \\ \Lambda(t - t_{0+}; \mu_+, \sigma_+) & t \in I_9 = t_{0+} + [e^{\mu+ + \sigma_+}, t_e] \end{cases}$$

Figure 2.13 Borne supérieure de la lognormale

$$\boxed{\underline{\Lambda}} = \begin{cases} 0 & t \in I_0 = [0, t_{0+}] \\ \Lambda(t - t_{0+}; \mu_+, \sigma_-) & t \in I_1 =]t_{0+}, t_1^*] \\ \Lambda(t - t_{0+}; \mu_+, \sigma_+) & t \in I_2 =]t_1^*, t_2^*] \\ \Lambda(t - t_{0-}; \mu_+, \sigma_+) & t \in I_3 =]t_2^*, t_3^*] \\ \Lambda(t - t_{0-}; \mu_-, \sigma_+) & t \in I_4 =]t_3^*, t_4^*] \\ \Lambda(t - t_{0-}; \mu_-, \sigma_-) & t \in I_5 =]t_4^*, t_e] \end{cases}$$

Figure 2.14 Borne inférieure de la lognormale

À partir de ces deux bornes, on déduit l'enveloppe Delta-Lognormale borné en remplaçant les valeurs de lambda supérieur et inférieur dans l'équation 2.1, ce qui donne :

$$\overline{\Delta\Lambda} = \overline{D_1} \ \overline{\Lambda_1} - \underline{D_2} \ \underline{\Lambda_2} \quad (2.33)$$

$$\underline{\Delta\Lambda} = \underline{D_1} \ \underline{\Lambda_1} - \overline{D_2} \ \overline{\Lambda_2} \quad (2.34)$$

La réalisation de l'espace de recherche par la séparation de l'agoniste et l'antagoniste dans l'équation Delta-Lognormale donne des résultats assez satisfaisants du point de vue de son approche théorique. Dans la pratique, l'élaboration de l'enveloppe est très dispendieuse en temps d'exécution. Chaque itération prend en moyenne 25 ms pour l'évaluation des bornes minimales et maximales de la fonction lognormale telles que présentées par les équations 2.33 et 2.34. Néanmoins, ces outils mathématiques restent valides pour les hypothèses émises et produisent de bons résultats dans certains cas. Comme le témoigne la figure 2.15 pour un test que nous avons effectué avec la boite :

$$\mathbf{P}^I = \{[\underline{D}_1, \overline{D}_1], [\underline{\mu}_1, \overline{\mu}_1], [\underline{\sigma}_1, \overline{\sigma}_1], [\underline{D}_2, \overline{D}_2], [\underline{\mu}_2, \overline{\mu}_2], [\underline{\sigma}_2, \overline{\sigma}_2], [\underline{t}_0, \overline{t}_0]\}$$

$$= \{[55.68, 93.95], [-0.789, -0.707], [0.206, 0.283], [0.70, 31.32], [-2.2, -0.1], [0.04, 0.55], [0.013, 0.023]\}$$

Toutes les courbes en bleu ne sortent pas de l'enveloppe en rouge. Les courbes en bleues sont les sous-ensembles de \mathbf{P}^I tel que $\mathbf{P}_i^I \leq \mathbf{P}^I$.

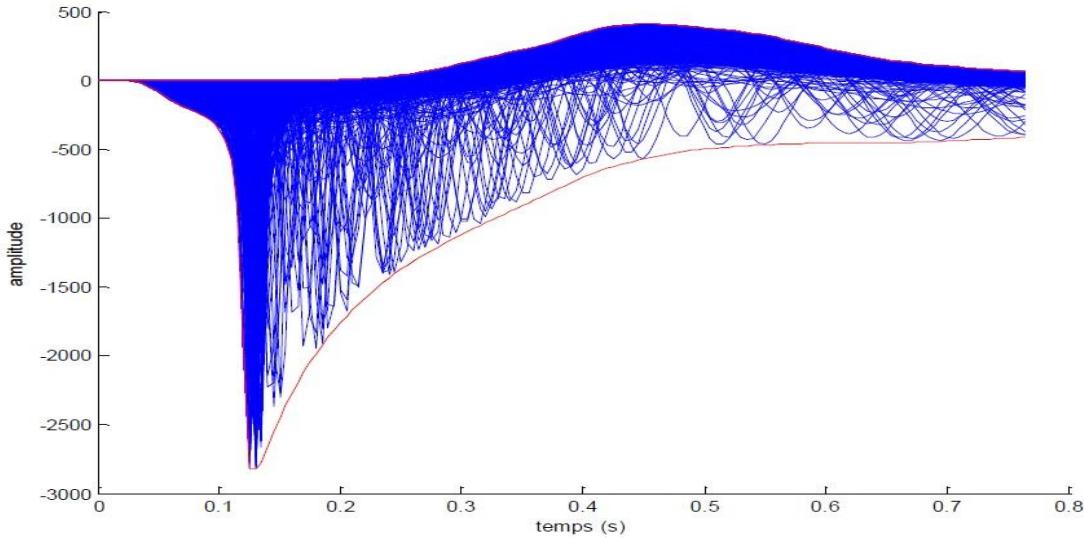


Figure 2.15 Enveloppe Delta-Lognormale en rouge renfermant des Delta-Lognormales en bleu

Pareillement, les auteurs de [52] proposent un algorithme pour l’ajustement de courbe d’un modèle basé sur une addition de fonctions lognormales tronquées et dont la qualité d’ajustement est évaluée grâce à l’erreur absolue (EA).

2.3.3 Évaluation des extracteurs Delta-Lognormaux Actuels et objectifs de recherches

L’ensemble des extracteurs à approche déterministe que propose l’état de l’art (**Xzero** et **IIX** [53]) repose sur une architecture commune composée de deux grands modules : un module d’estimation des paramètres en entrée, suivis d’un module d’optimisation non linéaire utilisant l’algorithme de Levenberg-Marquardt [44]. L’extracteur **IIX** est un regroupement de trois extracteurs en parallèle [48] [47] [53] (**INFLEX**, **INITRI**, **XZERO**) chacun avec sa particularité. Leurs différents modules d’approximation sont formés d’équations conçues à partir des morphologies des profils de vitesses Delta-Lognormaux.

Le fonctionnement général des algorithmes déterministes est le suivant; étant donné un profil de vitesse $v(t)$ provenant d’un mouvement manuscrit simple, les algorithmes commencent par estimer les paramètres lognormaux du système agoniste en considérant l’ensemble du profil de vitesse comme une seule lognormale. La lognormale associée au système antagoniste est par la suite déduite de la différence entre le signal de base $v(t)$ et la lognormale agoniste dont les paramètres ont été préalablement trouvés. Les paramètres de la Delta-Lognormale obtenus par ces

approximations servent par la suite de valeurs initiales à un algorithme d'optimisation non linéaire de Levenberg et Marquardt où il y est effectué une régression non linéaire entre les données expérimentales et les paramètres Delta-Lognormaux préalablement obtenus. La sortie de l'algorithme de Marquardt sont les sept paramètres Delta-Lognormaux représentant le profil de vitesse approximé ainsi que l'erreur quadratique moyenne entre les données observées et ce profil approximé.

Le fonctionnement de l'algorithme de type **S&E** est le suivant ; partant d'un profil de vitesse d'un geste manuscrit simple $v(t)$, les techniques faites par les algorithmes déterministes permettent d'obtenir les valeurs des paramètres initiaux. À partir de ces derniers, des ensembles bornés (boîtes et enveloppes associées) sont formés en utilisant les propriétés géométriques de la fonction de coût. Un processus d'énumération basé sur des heuristiques ou propriétés de la fonction de coût permet de créer des sous-ensembles qui sont évalués de façon optimiste entre eux afin de recouvrir le meilleur des sous-ensembles qui minimise l'erreur quadratique entre les valeurs mesurées et celles approximées.

Les points critiques des algorithmes déterministes précédents commencent par le processus d'estimation des paramètres initiaux : meilleures sont les solutions de départ meilleures sont les résultats de l'optimisation. L'algorithme **INFLEX** [48] utilise des tables de conversion et une méthode graphique initialement proposée par Wise [51] pour exploiter les caractéristiques des droites tangentes aux points d'inflexion d'une même lognormale. Cette approche permet d'estimer les quatre paramètres d'une Delta-Lognormale en considérant que la composante antagoniste devient significative seulement après le maximum du pic principal du profil de vitesse.

L'algorithme **INITRI** [47] utilise des relations analytiques établies entre les paramètres et l'occurrence temporelle du maximum de vitesse, en plus de deux autres points arbitraires situés avant le maximum, plus précisément dans la phase montante de la courbe de vitesse.

L'algorithme **Xzero** exploite les relations analytiques existantes entre une lognormale et différents points de passage par zéro du profil lognormale. Les temps d'occurrences du point maximum ainsi que ceux des deux points d'inflexion caractérisant une lognormale sont choisis pour déterminer les valeurs des quatre paramètres initiaux d'une lognormale. De façon générale, **Xzero** fusionne les deux idées maîtresses des algorithmes **INFLEX** et **INITRI**. Le problème considéré est la dépendance des deux précédents algorithmes dans la détermination de la position de l'agoniste par

rapport à celle de l'antagoniste dans un profil de vitesse afin de procéder aux estimations. Cette méthode permet d'offrir un estimateur fiable indépendamment de la position de la composante antagoniste par rapport à l'agoniste.

L'algorithme **IIX** regroupe dans son module d'approximation les algorithmes **INFLEX**, **INITRI**, et **XZERO** fonctionnant en parallèle [48] [47] [53]. La solution retenue pour l'optimisation par l'algorithme de Marquardt (qui constitue le module d'optimisation) est la valeur des paramètres d'un des trois algorithmes ayant obtenu la meilleure erreur quadratique au sens des moindres carrés entre la vitesse à reconstruire et son approximation par l'équation du Delta-Lognormale. **IIX** utilise un filtre passe-bas en amont du module d'estimation pour minimiser les petites variations en amplitudes dans le profil de vitesse afin de mieux l'ajuster pour l'application sollicitée.

Des études dans notre laboratoire ont montré que les algorithmes déterministes actuels échouent, notamment lorsque la composante antagoniste est prépondérante avant que la composante agoniste n'intervienne dans le profil de vitesse, et lorsqu'un bruit important est présent dans le signal [47]. Il est vrai que pour certaines applications en reconnaissance de forme manuscrite, une extraction des paramètres convergente avec **IIX** ne présente pas de grandes différences entre le mouvement émis et celui reproduit par le modèle Delta-Lognormale. Cependant, qu'en est-il de l'utilisation de ces paramètres dans des applications médicales pour des diagnostics ou caractériser dans un intervalle de confiance l'état de santé d'un groupe d'individus sujet à une étude? Cette information filtrée ou l'obtention d'un minimum local ou global, mais sans certitude sont-elles encore acceptables? De toute évidence, la réponse est non. Probablement qu'une statistique adaptée autour des solutions obtenues avec **IIX** puisse apporter des réponses avec des précisions plus ou moins acceptables, mais une solution exacte est généralement souhaitée.

Par ailleurs, bien qu'il existe un nombre impressionnant d'algorithmes d'optimisation globale, il est raisonnable de penser que l'utilisation d'un algorithme de recherche exact sans recours à une statistique ou probabilité quelconque est de mise afin de permettre une analyse plus discriminante. À cet effet, les algorithmes de type séparation et évaluation, dont la recherche s'effectue par énumération implicitement exhaustive de toutes les solutions possibles dans un espace fini sont adoptés dans cette thèse.

Les critères conduisant au choix de ce type d'algorithme sont les suivants :

- L'exactitude du résultat obtenu.

- La stratégie de recherche qui s'appuie uniquement sur un ensemble des paramètres mathématiques ou hypothèses réelles du problème.

Le tout premier extracteur développé avec un algorithme dit global à solution exacte est l'extracteur de O'Reilly [50]. L'objectif visé par sa conception est d'étendre l'utilisation des paramètres lognormaux à des applications où une solution locale n'est plus suffisante et référencer l'état de l'art sur l'extraction des paramètres Delta-Lognormaux. Les temps d'exécution excessifs pour recourir à une solution avec cet algorithme [50] ont donné naissance au présent projet. Il vise dans un premier temps à diminuer les délais d'exécution de l'algorithme précédent afin de faciliter et accroître son utilisation en clinique. Il est vrai que conceptuellement l'idée de base entre les deux algorithmes **S&E** et [50] est la même, mais en pratique les règles autour de chaque étape dans les deux algorithmes sont différentes et dans certains cas incompatibles.

L'algorithme **S&E** proposé dans cette thèse ne contient pas de filtre en amont, il travaille sur la totalité du signal de base sans hypothèses a priori sur une quelconque partie du signal. Chaque étape de l'architecture approximation/optimisation des algorithmes déterministes est accompagnée dans l'algorithme **S&E** d'une preuve de l'existence de la solution globale. L'architecture de l'algorithme **S&E** n'est pas directement du type approximation/optimisation, mais on pourrait assimiler les deux grands blocs qui le constituent à ces deux modules de base des algorithmes **IIX**. Dans cet ordre d'idée, l'approximation dans **IIX** et **XZERO** se rapporte à la séparation dans **S&E** et l'optimisation est associée à l'évaluation dans **S&E**. Dans la séparation, il s'y retrouve un hyper espace de recherche fini ou boîte, associée à une enveloppe garantissant l'inclusion de tous les points du profil de vitesse à l'étude. L'enveloppe fournit une borne maximale et minimale globale avec preuve de l'existence de la solution globale. À l'opposé du module d'approximation de **IIX** qui ne donne que des valeurs de base des paramètres sous les contraintes établies à partir de quelques points caractéristiques du signal de vitesse de base, l'algorithme **S&E** fournit une solution initiale complète dans laquelle il est garanti de retrouver le signal de base.

Le module de séparation de **S&E** est largement plus inclusif que l'approximation incluse dans **IIX**. En effet, dans son application, il n'exclut aucune partie du signal de base et un de ses critères de validation est l'inclusion complète du signal à l'étude dans l'enveloppe Delta-Lognormale conçue. Au-delà de l'enveloppe, les valeurs initiales de recherche de la séparation sont des intervalles garantissant une borne minimale et maximale globale, assez large et inclusive de la

solution optimale. Ces garanties autour de l'enveloppe du module de séparation s'appuient sur le théorème fondamental de l'arithmétique par intervalles. La séparation est par ailleurs la clef de la réussite pour un algorithme de type **S&E**.

Inspiré par les travaux de ces auteurs et malgré le fait que les systèmes **IIX** et **XOR** réunis ensemble offrent des performances intéressantes pour une panoplie d'applications en reconnaissance de formes et en analyse du mouvement, aucune certitude ne peut être offerte sur l'optimalité globale des solutions extraites. Cependant, l'utilisation des paramètres Delta-Lognormaux globalement optimaux est plus exigeante dans les cas où la connaissance de l'intervalle de confiance sur les solutions obtenues est nécessaire. Ceci est le cas lors de l'étude fondamentale du mouvement et lors du développement d'applications à caractère médical. C'est dans ce contexte que la présente thèse propose une approche par séparation et évaluation pour optimiser l'ajustement Delta-lognormal de façon à obtenir les paramètres Delta-Lognormaux globalement optimaux.

2.3.3.1 Objectifs et hypothèses de recherches

L'objectif principal de ce projet de recherche est la conception et l'implémentation d'un algorithme de type séparation et évaluation (Branch and Bound) pour le modèle Delta-Lognormal de la théorie cinématique du mouvement humain rapide. L'algorithme conçu permettra d'extraire des paramètres Delta-Lognormaux globalement optimaux pour l'investigation des troubles neuromusculaires chez l'humain.

Cet objectif général peut se répartir en cinq objectifs spécifiques :

- 1. Collecter des mouvements humains rapides chez des individus saints et malades pour entraîner l'extracteur.**
- 2. Créer des mouvements humains rapides synthétiques avec table de vérité. Ces mouvements devraient être semblables aux mouvements collectés sur des personnes saines et malades.**
- 3. Proposer un estimateur global des paramètres Delta-lognormaux en utilisant l'algorithme de séparation et évaluation (S&E) communément appelé (Branch and Bound).**
- 4. Établir des contraintes à cet extracteur global pour obtenir des délais d'extraction meilleurs que ceux obtenus dans [50] pour le problème considéré.**

5. Développer un banc de test pour la validation des méthodes proposées.

La réalisation de ce projet passe par la formulation de deux hypothèses de recherche sur lesquelles s'appuie l'ensemble des objectifs à atteindre. Ces dernières forment la pierre angulaire de ce projet et s'énoncent comme suit :

Hypothèse #1 : « *Il est possible de créer des profils de vitesse de mouvements synthétiques simples, semblables aux profils de vitesse des mouvements de pointage manuscrits collectés à partir des tablettes numériques et d'en extraire leurs paramètres Delta-Lognormaux à l'aide d'un algorithme de type S&E.* »

Cette première hypothèse relative à la conception des profils de vitesses de mouvements synthétiques à partir des mouvements manuscrits simples et la fonction Delta-Lognormale fait état des outils nécessaires à l'entraînement de l'algorithme afin de valider la précision des extractions qui y seront faites. Les profils de vitesse synthétiques avec table de vérité sont non seulement représentatifs des mouvements humains, mais permettent d'entraîner l'algorithme en lui fournissant une large banque de données très variées sans avoir recours à l'humain pour les effectuer. Ces profils conçus de façon algorithmique sur la base des observations faites sur des profils humains sont lisses, sans bruit, et possèdent des valeurs de paramètres connues qui permettront de valider l'exactitude du résultat d'une extraction et les ajustements à apporter à l'extracteur **S&E** pendant l'entraînement afin d'obtenir des extractions dans des délais raisonnables. Cette hypothèse sera réfutée si pour un profil de vitesse synthétique l'algorithme **S&E** ne trouve pas les paramètres de la table de vérité.

La seconde hypothèse fait état de l'exactitude des résultats fournis par l'algorithme **S&E**. Elle s'énonce comme suit :

Hypothèse #2 : « *il est possible d'obtenir automatiquement un intervalle de confiance sur la combinaison des paramètres approximant un profil de vitesse de mouvement simple de pointage manuscrit avec l'algorithme S&E. Les intervalles de confiance sur chaque paramètre ou vecteur d'intervalles (boîte) produisent une enveloppe Delta-Lognormale unique qui contient la totalité du profil de vitesse de la courbe à approximer.* »

Cette hypothèse novatrice fait état du caractère unique du vecteur d'intervalles des paramètres Delta-Lognormaux représentatif d'un mouvement manuscrit simple quelconque. Cette unicité n'a jamais été explicitement démontrée antérieurement.

En outre elle stipule que : une modélisation Delta-Lognormale, définie dans un espace de recherche fini et utilisé dans un algorithme de recherche exact, donne une boîte de solutions unique si elle existe. Cette hypothèse sera réfutée si :

1. Pour une même tâche motrice on retrouve un vecteur de paramètres ayant des valeurs différentes pour un même rapport signal sur bruit élevé.
2. Les valeurs des paramètres sont aberrantes ou irréalistes du point de vue physiologique de sorte que l'enveloppe Delta-Lognormale construite n'inclut pas la totalité du profil de vitesse à approximer.

CHAPITRE 3 ÉTAPE SUIVIES EN VUE D'ÉLABORER UN OUTIL D'ANALYSE AMÉLIORÉ POUR L'EXTRACTION DES PARAMÈTRES DELTA-LOGNORMAUX EN EXPLOITANT UN ALGORITHME DE SÉPARATION ET ÉVALUATION

Cette thèse constituée de texte non publié et de publications techniques (chapitres 5 et annexes A B C D et E) a pour objectif l'élaboration d'un extracteur de paramètres Delta-lognormaux globalement optimaux pour des tâches neuromotrices. Cet objectif principal tel que présenté au chapitre 2 peut être divisé en trois grands modules. Le premier module est l'élaboration d'une banque de données de tâches neuromotrices pour entraîner l'extracteur. Il consiste à capturer des mouvements manuscrits de pointages à partir d'outils matériels tels que des tablettes numérisées et créer des profils de vitesse de pointages manuscrits synthétiques. Le deuxième module est l'algorithme de séparation et évaluation appliqué au modèle Delta-Lognormal de la théorie cinématique du mouvement rapide pour extraire des paramètres Delta-Lognormaux des mouvements de pointages manuscrits. Ce module consiste à élaborer différentes contraintes sur la fonction Delta-Lognormale et les utiliser dans un algorithme de type S&E. Le dernier module est la construction d'un Banc d'essai pour effectuer des tests automatisés sur l'algorithme S&E. Il consiste à réunir en un seul bloc, le matériel de capture de mouvements manuscrits, l'algorithme S&E et la base de données des mouvements et solutions obtenues.

Ces trois modules pour lesquels plusieurs fonctionnalités sont associées représentent l'architecture du système d'extraction conçus. Tel qu'illustré à la figure 3.1, des mouvements manuscrits synthétiques (idéaux avec table de vérité) et réels (provenant de mouvements rapides d'humains) sont recueillis et insérés à l'entrée de l'extracteur Delta-Lognormal pour l'obtention des

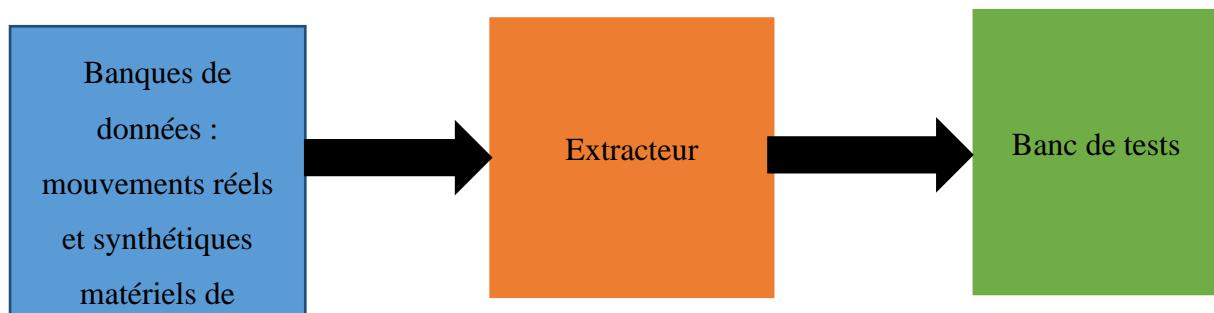


Figure 3.1 Architecture du système d'extraction proposée

paramètres. Un banc de test permet l'entraînement de l'extracteur et sa mise en condition exploitable dans diverses applications. La réalisation des différents modules sous forme de contributions scientifiques produites est décrite ci-dessous.

3.1 Banque de données : mouvements réels et synthétiques.

La banque de données est l'unité informatique (serveurs) d'entreposage des mouvements manuscrits réels et synthétiques qui serviront à tester l'algorithme **S&E**. Les mouvements de pointage manuscrit sont collectés à l'aide des tablettes numériques dans lesquelles est introduit un logiciel de capture des mouvements de pointage. Les mouvements sont capturés suivant un protocole de test neuromoteurs établis par les cliniciens. Les tests neuromoteurs sont des mouvements de pointage utilisés en clinique pour le diagnostic des maladies ayant une incidence sur la motricité. Les mouvements synthétiques sont produits par des algorithmes qui conçoivent des profils de vitesses synthétiques avec leur table de vérité à partir des caractéristiques observées sur des profils de vitesses d'écritures manuscrites réelles. Cette section porte sur la réalisation des données réelles et synthétiques et se divise en deux sous-sections :

- La création de la banque des données réelles
- Création de la banque des données synthétiques.

3.1.1 Banque de données réelles

Les traits d'écriture manuscrite qui constituent une classe spécifique de mouvements rapides de pointages [54] sont utilisés comme unités élémentaires du mouvement à l'étude. Leurs choix se justifient par la disponibilité du matériel permettant leur capture et leur reproduction par la fonction Delta-Lognormale. Dans ce contexte, les données réelles sont obtenues à partir de mouvements humains captés à l'aide d'un stylo numérique et d'une tablette numérisée (Wacom Cintiq) [55] dans laquelle est inséré un logiciel de capture que nous avons adapté et amélioré. Le logiciel de capture est un algorithme permettant la détection et l'acquisition des mouvements de pointages effectués une tablette numérisée branchée à un ordinateur par connexion USB. L'acquisition est effectuée selon un protocole fixé par l'utilisateur pendant un temps fini à une fréquence de 200Hz. Le protocole de manière générale est l'ensemble constitué de la trajectoire à parcourir, les cibles à atteindre et les signaux de début et fin de séquence respectivement. Le système de capture ou

lognomètre formé de l'association stylo, tablette, et logiciel fait l'objet de deux publications (copie en annexe D et E) dans lesquelles sont détaillées les techniques de conception, les tests et les résultats obtenus.

Notons cependant que le système de capture de mouvements manuscrits proposé est une version améliorée d'un ancien système obsolète avec les nouveaux systèmes d'exploitation. L'ancien système avait été préalablement conçu à l'École Polytechnique de Montréal il y a plusieurs décennies. Cette partie du projet s'inscrit donc dans l'amélioration des travaux antérieurs avec pour objectifs principaux;

- ✓ La migration de l'ancien système vers des systèmes d'exploitation récents (Windows 10)
- ✓ L'implémentation des différents tests neuromoteurs dans une nouvelle tablette choisie avec suppression de certains modules jugés moins ergonomiques.

Dans cet ordre d'idée ;

La première contribution est l'ensemble des réalisations suivantes :

- L'intégration de l'ensemble de tests neuromoteurs nécessaires pour recueillir des mouvements de pointage humain dans un système de capture de mouvement utilisant une tablette et un stylo numérique.
- L'élaboration et le respect des contraintes d'acquisition des échantillons de mouvements pour un test neuromoteur établit à une fréquence de 200HZ et une résolution de 100 points par millimètre.
- L'insertion d'une base de données conçue en langage C++ pour conserver les mouvements de pointage captés par la tablette selon différentes contraintes associées aux types de mouvement et au nom du projet pour lequel sont recueillies les données.

Ces travaux sont décrits dans l'article intitulé : « **A New Experimental Set-up to Run Neuromuscular Tests** » qui a été publié à la conférence : « International Conference on Pattern Recognition and Artificial intelligence: Workshop on the Lognormality Principles and its Applications » (**ICPRAI**) Montréal 2018. Cet article met en relief l'implémentation de la batterie de tests neuromoteurs proposée par le personnel en clinique et les ingénieurs.

La deuxième contribution est l'intégration complète de la batterie des neuf tests neuromoteurs dans la tablette numérique, la transformation en logiciel de l'outil matériel externe permettant de réaliser certains tests neuromoteurs nécessitant un stimulus visuel, l'intégration de la version logicielle du stimulus visuel dans la tablette, et la mise à niveau de l'ensemble du système pour une utilisation dans un système d'exploitation moderne (Window 10)

Cette contribution a fait l'objet d'un article: « **An Interactive Tablet-based System to Run Neuromuscular Tests** » qui a été publié dans le livre « The Lognormality Principle and its applications in E-Security, E-learning and E-Health ». Cet article poursuit dans la même veine d'idées que la publication de conférence précédente. Elle met de l'avant un système d'acquisition complet et autonome.

Le tableau 3.1 présente l'ensemble des mouvements manuscrits recueillis avec le nouveau système d'acquisition pour entraîner l'algorithme **S&E**. Les différents groupes de participants à la collecte de données sont : des athlètes de football universitaire, des personnes atteintes de la maladie de Parkinson, des personnes présentant des troubles de l'attention ainsi que des personnes ayant manifesté des symptômes d'une commotion cérébrale. Chaque participant devait effectuer sur la tablette deux mouvements rapides par tests neuromoteurs avec chacun des bras.

Table 3.1 Échantillon expérimental recueillis avec le lognomètre pour entraîner l'algorithme S&E

Types de participants	Nombres de participants	Nombres de traits simples	Age (année)	Totale de traits
Athlètes de football	18	20/personnes	18-25	360
Personnes atteintes de TDAH	2	20	18-25	40
Personnes atteintes de la maladie de Parkinson	3	20	50 et plus	60
Personnes ayant subi une commotion cérébrale	2	20	18-30	40

Le nombre total des mouvements pour un test avec un bras est de 30 traits d'au moins 130mm. Cette expérience a été faite conjointement avec le personnel de l'Université de Montréal sur approbation du comité d'éthique de l'École Polytechnique, des participants, et de l'Université de Montréal. Un grand nombre de mouvements manuscrits simples ont été recueillis et emmagasinés dans nos serveurs. De cette banque, cinq cents (500) mouvements provenant des 5 groupes de participants ont été prélevés pour constituer notre échantillon de test.

3.1.2 Données synthétiques

Les données synthétiques ou artificielles sont des profils de vitesse de mouvements de pointage semblables aux profils de vitesses réels et ayant des valeurs de paramètres Lognormaux connus. Elles sont utilisées dans cette thèse comme modèle de base pour tester et évaluer la capacité de l'algorithme **S&E** à produire des solutions globalement optimales. Les profils de vitesses synthétiques sont créés à partir des variabilités observées sur des mouvements manuscrits de pointage collectés sur une tablette numérique et les propriétés de la fonction Delta-Lognormale selon la théorie cinématique du mouvement humain rapide. Les variabilités identifiées et mises à l'étude sont :

- Les changements observés dans l'espace associés aux traits d'écriture
- Les changements remarqués dans le temps au niveau des profils de vitesse des traits d'écriture.

La figure 3.2 présente le schéma bloc de l'ensemble du processus de génération des données

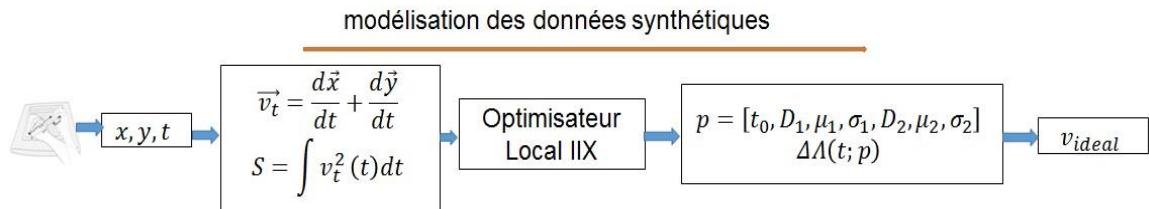


Figure 3.2 Schéma bloc de la modélisation des données synthétiques

synthétiques. Les coordonnées cartésiennes du mouvement émis sur la tablette entrent dans le système et sont transformés en vitesse. Les échantillons de vitesse obtenus par dérivation des différentes positions entrent dans l'extracteur **IIX** pour produire les sept paramètres Delta-

Lognormaux avec lesquelles la vitesse synthétique ou idéale ($V_{\text{idéal}}$) sera établie. Les vitesses synthétiques sont réalisées en évaluant l'équation Delta-Lognormale avec les paramètres de IXX échantillonné à la fréquence du mouvement réel à l'entrée. Les variabilités observées sur les traits et leurs profils de vitesses associés ont permis d'établir neuf classes différentes de profils de vitesses synthétiques. À partir de ces variabilités et des contraintes géométriques imposées à la fonction Delta-Lognormale nous avons établi deux algorithmes (6.1 et 6.2) détaillés au chapitre 6 de cette thèse. Ces algorithmes ont permis de construire automatiquement une banque de neuf mille (9000) profils de vitesse Delta-lognormaux synthétiques. La banque se compose de mille (1000) courbes pour chacune des classes identifiées et permet d'entraîner l'algorithme de **S&E** développés. Les détails sur la conception des différentes classes de profils de vitesse et l'élaboration des algorithmes permettant de générer automatiquement ces profils de vitesses sont fournis dans les textes inédits de cette thèse qui sont rapportés dans le chapitre 6 comme aspect méthodologique et résultats complémentaires pour la création de la banque des données synthétiques. Le tableau 3.2 résume les trois grands groupes de profils de vitesse synthétique de la banque de 9000 profils synthétiques. Chaque groupe représente la position significative de l'activité des deux composantes neuromusculaires impliquées dans l'accomplissement du mouvement.

Table 3.2 Échantillon expérimental des données synthétiques l'algorithme S&E

Noms des groupes des classes	Nombre total de profils de vitesse
Activités significatives de la composante antagoniste au début du mouvement.	3000
Activités significatives des composantes antagoniste et agoniste simultanément	2000
Activités significatives de la composante antagoniste à la fin du mouvement	4000

3.2 Extracteurs

Le deuxième module de la figure 3.1 est responsable de l'extraction des paramètres Delta-lognormaux globalement optimaux à partir des données des tâches neuromotrices des différentes banques de données recueillies dans le premier module. Le problème considéré dans ce module est une régression non linéaire dans laquelle on effectue l'extraction des paramètres à partir des

données brutes associées à un modèle non linéaire (delta-lognormal) à sept dimensions. Les mouvements numérisés sont reconstruits à l'aide de la fonction Delta-Lognormale sans a priori ou modification sur les données brutes. La non-modification des données brutes comme contrainte supplémentaire est associée aux exigences médicales en vue d'une utilisation de la solution en milieu clinique. En effet, selon les personnes en santé, l'acceptation d'une perte d'information ou d'une solution locale n'est plus acceptable lorsque vient le temps de poser un diagnostic. L'algorithme s'y rapportant se doit d'être non seulement un excellent ajusteur de courbe, mais aussi un filtre passe tout (si on peut le qualifier ainsi) qui permet la redistribution adéquate de l'information pour son ajustement. En considérant toutes les exigences sus mentionnées, un effort majeur a été consenti dans le développement d'un algorithme de régression à caractère global. L'algorithme d'optimisation choisi (après prototypages rapides et indices tirés de la littérature) est un algorithme de type séparation et évaluation (*Branch and Bound*).

L'algorithme de séparation et évaluation (**S&E**) est bien connu dans le domaine de la recherche opérationnelle et se définit par les deux mots qui le caractérisent :

- **Séparation** : Consiste à diviser un espace fini en sous-espace.
- **Évaluation** : revient à déterminer et comparer chaque sous-espace solution potentiel de façon objective.

Son concept repose sur la construction d'un arbre de recherche à partir d'un espace fini assez grand. Le grand espace est d'abord construit puis divisé en sous-espaces finis garantissant l'inclusion de la solution recherchée. Les sous-espaces construits sont évalués de manière '**intelligente**' afin de ne conserver que celui ou ceux qui garantissent d'aboutir à la solution globale. Dans la majorité des cas, la solution globale est soit l'ensemble des minimums ou maximums associés à une fonction de coût. Dans cette thèse, il s'agit d'effectuer une régression non linéaire à sept dimensions avec pour objectif la minimisation de l'erreur quadratique entre le modèle Delta-Lognormal et des données provenant de mouvements d'écritures manuscrites recueillis sur des tablettes numérisées. La qualité d'ajustement doit être supérieure à un seuil défini par le rapport signal sur bruit. Le signal est modélisé comme le carré de la vitesse du mouvement manuscrit numérisé et le bruit est l'erreur quadratique entre la vitesse du mouvement manuscrit numérisé et la fonction delta-lognormale.

La mise en œuvre de l'algorithme de séparation et évaluation passe par l'élaboration de quatre étapes principales :

- La définition et la réalisation de l'espace de recherche.
- La détermination des bornes supérieures et inférieures du grand espace.
- La séparation des espaces en sous-espaces
- L'évaluation des sous-espaces entre eux.

L'algorithme construit se répartit successivement sur quatre contributions scientifiques (chapitre 4, et annexe A B C) associées à chacune des étapes mentionnées ci-dessus. Les sections suivantes résument les différentes contributions ainsi que les liens entre elle et l'atteinte des objectifs de la problématique. Dans le chapitre 6 non publié, sont rapportés les résultats complémentaires aux différents développements effectués pour la réalisation de ce module.

3.2.1 Définition et réalisation de l'espace de recherche : utilisation de l'arithmétique par intervalles

L'espace de recherche ou boite englobante « bounding box » est un vecteur d'intervalles représentant les sept paramètres de la fonction Delta-lognormale. Chaque intervalle contient la valeur minimale et maximale du paramètre qui lui est associé. La caractéristique principale de l'espace de recherche est le principe d'inclusion. En effet, la réunion de sous-ensembles d'un espace englobant doit être égale à l'espace englobant lui-même. Chaque boite englobante à travers une application engendre une enveloppe. Dans le cas du problème à résoudre, l'enveloppe est l'ensemble des valeurs possibles de l'équation Delta-Lognormale pour une boite donnée. En d'autres mots, l'enveloppe est l'image d'une boîte à travers la fonction delta-lognormale. L'enveloppe englobe les valeurs possibles de l'équation Delta-Lognormale pour des paramètres variant à l'intérieur d'une boite. Donc elle respecte aussi la propriété d'inclusion.

À partir de ces définitions et propriétés, l'enveloppe supérieure et inférieure de l'ensemble des Delta-Lognormales dont la valeur est comprise dans une boite est obtenue en majorant et minorant les résultats des évaluations de chacune des sous fonctions qui compose la fonction Delta-Lognormale pour une boite donnée. Dans cette thèse, la technique développée permettant de

produire une telle enveloppe repose sur l'évaluation de la fonction Delta-Lognormale en utilisant l'arithmétique par intervalles et son théorème fondamental.

Les contributions suivantes :

1. Conception de la borne minimale et de la borne maximale pour la fonction Delta-Lognormale en utilisant l'arithmétique par intervalles et son théorème fondamental.
2. Conception de l'enveloppe Delta-Lognormale pour quatre classes de profils de vitesses synthétiques et dans un algorithme de type **S&E** pour extraction des paramètres Delta-Lognormaux.

Les articles intitulé respectivement « **On the use of Interval Arithmetic to Bound Delta-Lognormal Rapid Human Movements Models** » de la conférence « International Conference on pattern Recognition and artificial intelligence: Workshop on the Lognormality Principles and its Applications » (ICPRAI) Montréal 2018, et « **On the Use of Interval Arithmetic for the Branch and Bound Delta-Lognormal Parameter Extraction of Rapid Human Movements** » du livre « The Lognormality Principle and its applications in E-Security, E-learning and E-Health » dont des copies conformes aux originaux sont présentés aux annexes A et B de cette thèse. Ces articles dressent un portrait détaillé des grandes techniques utilisées.

On y retrouve la conception de l'espace **P** communément appelé Boite englobante (Bounding Box) qui définit l'ensemble des solutions possibles pour le problème d'extraction de paramètres Delta-Lognormaux considérés. Ces articles contiennent également la définition d'une boite englobante, la dérivation de la fonction de coût pour la problématique, les différentes opérations arithmétiques et théorèmes permettant de construire une enveloppe pour certaines classes de profils de vitesse de mouvements manuscrits à l'étude. De plus les articles rapportent les travaux qui ont conduit à la mise en œuvre d'une bibliothèque en arithmétique par intervalles avec arrondi pour les opérations arithmétiques et fonctions transcendantales qui composent la fonction Delta-Lognormale. Bien que la majorité des développements de la bibliothèque font partie des contributions non publiées de cette thèse, un ensemble d'outils informatiques et mathématiques de l'arithmétique par intervalles permettant l'élaboration des différents espaces de recherches utiles à l'algorithme de **S&E** sont proposées dans cette contribution.

Les résultats obtenus pour ces contributions sont :

- Un algorithme permettant de générer automatiquement l'hyperespace à sept dimensions et son enveloppe associée pour quatre classes de profils de vitesse synthétique Delta-Lognormaux identifiés dans le processus de conception des vitesses synthétiques.
- La preuve et la garantie de l'inclusion des sous-enveloppes dans l'enveloppe d'ensemble et une intégration de cet algorithme dans un **S&E** de base afin de valider avec succès la qualité de la précision des paramètres recouvrés dans une enveloppe pour un profil synthétique donné.

3.2.2 La détermination des bornes supérieures et inférieures d'un espace en utilisant l'arithmétique par intervalles sur neuf classes de vitesses synthétiques.

La conception des bornes pour un algorithme de type **S&E** est une étape cruciale qui nécessite plusieurs tests afin de s'assurer que les bornes conçues ne permettent en aucun cas la perte d'une solution potentielle pendant la phase d'évaluation d'un algorithme **S&E**. Cette section poursuit dans la même veine que les deux contributions présentées à la section précédente. Elle se penche davantage sur la généralisation du processus d'obtention de bornes des profils de vitesses artificiels Delta-Lognormales connues de toute la banque de mouvements enregistrés (neuf classes).

La contribution principale est l'utilisation du gradient de la fonction de coût (erreur quadratique moyenne entre données observées et fonction Delta-Lognormale) pour choisir la dimension de coupure d'une enveloppe englobante en sous enveloppes et permettre une extraction des paramètres Delta-Lognormaux sur neufs classes différentes de profils de vitesses synthétiques dans un algorithme de type **S&E** [50] en vue d'effectuer des tests sur les délais d'exécution et la précision des valeurs des paramètres recouvrés.

L'article intitulé : « **Combining Interval Arithmetic with the Branch and Bound Algorithm for Delta-lognormal Parameter Extraction** » présenté à la conférence « International Conference on Pattern Recognition and Artificial Intelligence: Workshop on the Lognormality » Cancún Mexico 2019 est fourni en annexe C de cette thèse. Cette publication propose une technique de division de l'espace en sous-espace et la définition de l'évaluation des fonctions

transcendantales par la méthode de l'extension par intervalle (natural interval extension) proposée par [56].

Le résultat obtenu est un algorithme plus inclusif que celui proposé dans [57]. Il permet la génération des enveloppes pour toute la banque de profils de vitesses synthétiques connus. Un premier test d'extraction dans un algorithme simple de **S&E** et des résultats d'extraction qui complète avec des temps encourageants, allant d'une à deux heures comparativement à des jours pour le même algorithme lorsqu'il n'utilise pas les bornes conçues.

3.2.3 Séparation des espaces en sous-espaces et évaluation comparative des sous-espaces

La séparation est le processus de division du grand espace (bounding box) dont la conception fait l'objet des contributions 1 et 2 en sous-espace. L'évaluation est le processus de compétitions par énumérations des sous-espaces (solutions potentielles) dans le but de conserver la meilleure selon notre objectif. Dans notre cas la meilleure solution est celle ayant le rapport signal sur bruit le plus élevé comparativement aux autres. Les contributions suivantes :

- ✓ La définition de l'optimum global pour le problème d'extraction des paramètres Delta-Lognormaux.
- ✓ L'élaboration et la validation de 5 différentes règles permettant l'évaluation des solutions candidates.
- ✓ L'élaboration automatique (algorithme) des données synthétiques pour toutes les classes (9) de profils de vitesses de mouvements rapides simples connus à ce jour.
- ✓ Des tests avec résultat concluant dans un algorithme de types **S&E** sur l'ensemble des données synthétiques de toutes les classes.
- ✓ Les résultats d'un ensemble de tests de mouvements de pointages humains imposés et extrait avec succès, dans des délais raisonnables par l'algorithme **S&E**.

L'article « **A Global Optimization Algorithm for the Estimation of Delta-Lognormal Parameters Characterizing Pointing Gestures** » qui a été soumis pour publication dans la revue « **Pattern Analysis and Machine Intelligence** » (**PAMI**) met de l'avant la séparation et l'évaluation. Cette publication résume et applique les différentes techniques de construction des bornes

supérieures et inférieures de la fonction Delta-Lognormale pour tout type de profil de vitesse, y compris les profils de vitesse artificiels dans lesquelles du bruit blanc gaussien a été inséré afin de simuler des mouvements manuscrits réels. Cet article propose une méthode générale de conception des bornes (minimales et maximales) des enveloppes en s'appuyant sur certains points caractéristiques des profils de vitesse. La phase d'évaluation proposée dans cette contribution est un groupe de procédures basé sur les propriétés géométriques de la fonction Delta-Lognormale et du théorème fondamental de l'arithmétique par intervalles. On y retrouve une méthode de sélection d'un sous-espace (Boite) candidat à une évaluation, une procédure qui permet de borner chaque boite candidate avec la garantie d'inclusion dans le domaine de base, des propriétés de rejets et d'acceptations des boites candidates en tant que solutions non prometteuses ou finales respectivement. La combinaison de la phase de séparation et d'évaluation que comporte cette contribution conduit à l'élaboration d'un algorithme de type **S&E** complet pour le problème d'extraction des paramètres Delta-lognormaux.

Les résultats qualitatifs et quantitatifs obtenus sur un ensemble de tests effectués en utilisant toute la gamme de traits manuscrits simples contenus dans notre base de données sont présentés en détail aux chapitres cinq (5) où est détaillée cette contribution. On peut noter des temps d'exécution de l'ordre de la dizaine à la vingtaine de minutes comparativement à des jours voire des semaines pour d'autres travaux du même type [50]. On note également une reconstruction jusqu'à la limite de précision de l'ordinateur pour toutes les classes de profils de vitesses Delta-lognormaux artificiels de la base de données.

3.3 Banc de test:

Le banc d'essai conçu pour entraîner l'extracteur **S&E** est un programme informatique en langage C++ et Matlab qui permet l'envoi automatique de tous les groupes de fichiers de mouvements manuscrits rapides simples préenregistrés dans notre base de données ou directement pris des tablettes numérisantes Wacom vers l'extracteur pour évaluation des profils de vitesses. Les résultats obtenus sont retournés sous un format de fichier choisi par l'utilisateur. La figure 3.3 présente l'architecture du Banc de test. Elle comporte une base de données, un système d'acquisition de mouvement manuscrit simple, un extracteur de paramètres Delta-lognormaux par la méthode **S&E** dans lequel est inséré un programme informatique permettant l'affichage et la sauvegarde des résultats d'extraction des profils de vitesse reconstruits par l'algorithme **S&E** dans

la base de données. Le système d'acquisition est composé d'une tablette numérisante sur laquelle des participants effectuent des mouvements rapides et d'un logiciel d'émission de stimuli pour guider l'utilisateur dans l'émission du mouvement manuscrit rapide afin de collecter des données. Les publications « **An Interactive Tablet-Based System to Run Neuromuscular Tests** » de la conférence « International Conference on pattern Recognition and artificial intelligence: Workshop on the Lognormality Principles and its Applications » (ICPRAI) Montréal 2018, et « **A New Experimental Set-Up To Run Neuromuscular Test** » du livre « The Lognormality Principle and its applications in E-Security, E-learning and E-Health » dont des copies conformes aux l'originaux sont présentée aux annexes D et E de cette thèse. Ces articles dressent un portrait détaillé des techniques mises en œuvre.

Deux grandes étapes ont été implémentées en C++ pour la création de la base de données :

1. La première étape transforme des fichiers d'écritures reçus du système d'acquisition en fichiers de type **HWS** (Hand writing Stroke) requis par l'algorithme **S&E**. Cette opération consiste à enlever du contenu des fichiers d'écritures émis par la tablette numérisante, les

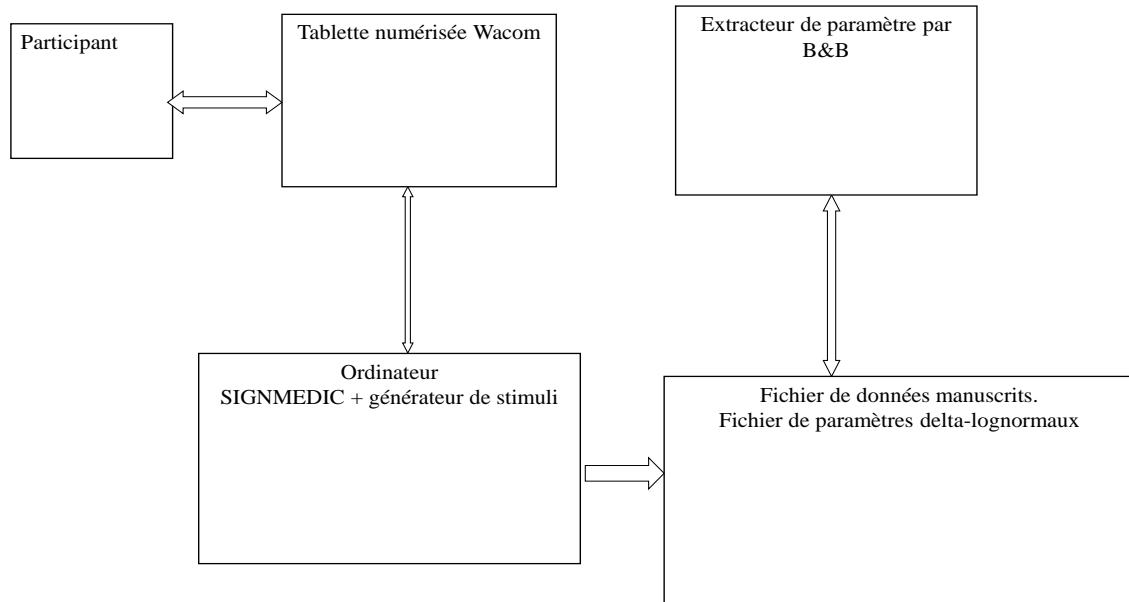


Figure 3.3 Schéma bloc du banc de test de l'algorithme **S&E**

valeurs autres que les différentes positions cartésiennes du mouvement manuscrit enregistré sur la tablette. Cette étape du traitement ajoute un entête comportant le nombre de points

recueillis par la tablette pendant l'émission du mouvement, la taille du fichier en octets et le nom du programme pour lequel l'opération d'acquisition est effectuée.

2. La seconde étape écrit les résultats de l'extraction des paramètres Delta-lognormaux par l'algorithme **S&E** dans les espaces mémoire (disque de stockage) disponibles et réservés par l'usager à cet effet.

Ce banc d'essai a permis l'entraînement de l'algorithme **S&E** afin que ce dernier soit capable d'extraire dans des délais raisonnables les paramètres Delta-lognormaux optimaux pour les neuf classes de profil de vitesses de mouvements manuscrits simples de notre base de données.

3.4 Synthèse des étapes requises pour la réalisation d'un outil capable d'extraire les paramètres Delta-Lognormaux des mouvements manuscrits simples

Ce chapitre situe le problème d'ajustement de courbes du modèle Delta-lognormal aux données empiriques pour des applications biomédicales. Les concepts fondamentaux permettant de réaliser un algorithme de type **S&E** pour extraire les paramètres Delta-lognormaux des profils de vitesses de pointage manuscrits sont présentés et illustrés. Dans le premier module de l'architecture du système, l'appareillage (matériel et logiciel) et le protocole expérimental conçus pour l'acquisition de mouvement rapide à partir de tablettes numériques ont été présentés. Les principales caractéristiques de l'échantillon des sujets ayant participé à la collecte des données avec le lognomètre conçu ont été rapportées. Le deuxième module a proposé le développement des bornes minimales et maximales avec la garantie d'inclusion des sous-espaces s'y rattachant. Les procédures de division de l'espace en sous-espaces et leurs évaluations successives. Enfin, le troisième module intègre l'ensemble pour former le banc de test afin d'extraire des paramètres Delta-lognormaux de profils de vitesses de pointages manuscrites. Les tests et résultats des différentes phases de conceptions ainsi que de l'algorithme final sont présentés dans les contributions associées. Des méthodes inédites (non publiées) et complémentaires pour l'atteinte des objectifs de cette thèse sont proposées au chapitre six (6).

CHAPITRE 4 ARITHMÉTIQUE PAR INTERVALLES ET THÉORÈME FONDAMENTAL

L'algorithme de type **S&E** proposé dans cette thèse et détaillé dans la publication « **A Global Optimization Algorithm for the Estimation of Delta-Lognormal Parameters Characterizing Pointing Gestures** » est basé sur l'arithmétique par intervalles et son théorème fondamental. Ce chapitre résume les propriétés de l'arithmétique par intervalles (**AI**) nécessaires à la compréhension des techniques proposées dans la suite du texte, spécialement dans le chapitre 5 où ces propriétés sont utilisées pour borner la fonction de coût à optimiser. Le chapitre se répartit sur trois sections comme suit : La section un (1) présente et illustre les opérations en **AI** pour des fonctions transcendantales de bases (\exp , \ln etc..) en lien avec la fonction de coût à optimiser. Par la suite, elle met en lumière le phénomène de dépendance lié au calcul par intervalle et quelques méthodes permettant de le réduire. Cette section se conclut par la proposition de diverses propriétés et définitions utiles pour l'élaboration des opérations de bornages nécessaires dans un algorithme de type **S&E**. La section deux (2) présente l'implémentation sur ordinateur en utilisant l'arithmétique usuelle. La section trois (3) conclue le chapitre en énonçant le théorème fondamental de l'**AI** qui est d'une utilité déterminante dans l'algorithme de type **S&E** proposé au chapitre 5 de cette thèse.

4.1 Opération sur les fonctions transcendantales de bases

L'arithmétique par intervalles est une méthode qui consiste à évaluer des nombres en les remplaçant par des intervalles. Une fonction d'intervalle $f(X_1, X_2, \dots, X_n)$ est une fonction à valeur d'intervalle dont les variables sont au moins un intervalle. Les opérations sur les fonctions transcendantales de bases se définissent comme suit ;

Soit f une fonction d'une variable réelle x , nous aimerais connaître la plage des valeurs prises par $f(x)$ pour x variant dans X . Il s'agit donc de déterminer l'image $f(X)$, de l'ensemble des X sous l'application définie par f . Le raisonnement est particulièrement simple pour des fonctions monotones (croissantes ou décroissantes). Par exemple soit $f(x)$ une fonction continue définie dans l'espace fini $X = [\underline{x}, \bar{x}]$. Dans le cas où f est croissante on a;

$$f(X) = [f(\underline{x}), f(\bar{x})]$$

Si f est décroissante on obtient;

$$f(X) = [f(\underline{x}), f(\bar{x})]$$

En appliquant cette logique aux fonctions régulières de bases on obtient :

Soit $X = [\underline{x}, \bar{x}] \in IR$, et $f(x) = x^2$ pour $x \in X$

$$f(X) = \begin{cases} [\underline{x}^2, \bar{x}^2] & \text{if } \underline{x} > 0 \\ [\bar{x}^2, \underline{x}^2] & \text{if } \bar{x} < 0 \\ [0, \max\{\underline{x}^2, \bar{x}^2\}] & \text{if } 0 \in X \end{cases} \quad (4.1)$$

Soit $f(x) = \exp(x)$ pour $x \in X$

$$f(X) = [\exp(\underline{x}), \exp(\bar{x})] \quad (4.2)$$

Similairement la fonction $f(x) = \ln(x)$ avec $x \in X$ et $x > 0$ on obtient :

$$f(X) = [\ln(\underline{x}), \ln(\bar{x})] \text{ pour } \underline{x} > 0 \quad (4.3)$$

Pareillement pour la fonction $f(x) = \sqrt{x}$ avec $x \in X$ et $x > 0$

$$f(X) = [\sqrt{\underline{x}}, \sqrt{\bar{x}}] \text{ pour } \underline{x} > 0 \quad (4.4)$$

Les différentes définitions assument que les opérations arithmétiques sont effectuées avec de l'arithmétique exacte. En pratique, afin de mettre à profit les propriétés de calcul par intervalle en utilisant l'arithmétique idéale disponible sur ordinateur, l'arrondi tels que spécifié par la norme IEE-754 est utilisé. Quatre modes d'arrondi sont appliqués selon le type de résultat désiré. L'arrondi vers l'infini par valeur négative ($-\infty$), l'infini par valeur positive ($+\infty$), zéro par valeur négative (-0), et zéro par valeur positive ($+0$). Chaque opération dont le résultat appartient à la borne inférieure, se voie appliquer un arrondit des calculs vers $-\infty$ et inversement pour les opérations faites sur la borne supérieure où sont l'arrondi est vers $+\infty$.

Notons que les différentes fonctions présentées aux équations 4.1 à 4.4 sont des fonctions monotones croissantes. Le cas des fonctions décroissantes s'effectue en inversant adéquatement le résultat obtenu lors de l'évaluation de la fonction dans un intervalle donné.

D'autre part, notons également que les fonctions non monotones définies dans un domaine peuvent être découpées par morceaux dans leur domaine et ainsi répondre adéquatement à la propriété de monotonie. En effet, on peut se servir des propriétés connues de la fonction afin d'effectuer un découpage par morceau du domaine de définition et rendre l'évaluation de la fonction monotone.

Par exemple la fonction $f(x) = \sin(x)$ $x \in IR$ n'est pas monotone dans IR. Néanmoins dans l'intervalle $X \in \left[\frac{-\pi}{2}, \frac{\pi}{2}\right]$ $\sin(x)$ est monotone et son évaluation dans le domaine $X \in [\underline{x}, \bar{x}]$. est $\sin(X) = [\sin(\underline{x}), \sin(\bar{x})]$. De manière plus générale, le cas des fonctions à plusieurs variables se définit comme suit : soit $f = f(x_1, x_2, \dots, x_n)$ une fonction à plusieurs variables nous cherchons à connaître l'image de l'ensemble des valeurs de f à travers les intervalles X_1, X_2, \dots, X_n pour chaque $x_1 \in X_1, x_2 \in X_2, \dots, x_n \in X_n$. Il s'agit donc de déterminer l'image $f(X_1, X_2, \dots, X_n)$, de l'ensemble des X sous l'application définie par f . Cette image lorsque calculée en utilisant l'arrondi est la fonction F qui est la fonction d'inclusion de f .

$$f(x_1, x_2, \dots, x_n) \in F(X_1, X_2, \dots, X_n)$$

4.1.1 Dépendances de l'AI dans le cas des fonctions transcendantales

Lors de l'évaluation d'une fonction ayant plusieurs variables réparties dans son expression (la mise en facteur n'étant pas possible), chaque apparition d'une variable dans une opération est considérée comme une nouvelle variable. Dans le cas des fonctions à plusieurs variables la dépendance apparaît sous la forme suivante : $f_1(X_1, X_2) = (X_1^2 + 2X_1X_2 + 3 + X_2^2)$ et non dans $f_2(X_1, X_2) = (X_1 + X_2)^2 + 3$ car dans f_1 les variables X_1, X_2 apparaissent à plusieurs reprises dans son expression.

L'exemple 2 ci-dessous pourra en convaincre le lecteur. Moore [58] montre que si dans l'expression d'une fonction, les variables n'apparaissent qu'une seule fois alors l'évaluation de la fonction en utilisant **AI** donne un résultat d'ensemble exact sans surestimation. On note à travers cette remarque que le phénomène de dépendance peut être réduit en factorisant si possible la fonction à évaluer, ou en la découplant adéquatement en une séquence regroupant ses différentes opérations arithmétiques et fonctions transcendantales élémentaires. La séquence obtenue ne doit comporter qu'une variable à la fois dans chacune de ces étapes. L'évaluation de chaque étape de la séquence se fait en utilisant les différents modes d'arrondis selon la norme IEE-754.

D'autres parts, des propriétés basées sur la monotonie des fonctions transcendantales de bases peuvent aussi être utilisées pour réduire la dépendance en **AI**. C'est le cas des fonctions quadratiques, racine carrée, logarithmique pour lesquelles l'évaluation peut être restreinte seulement au calcul de la borne supérieure (lorsqu'un des résultats d'ensemble présente une valeur négative) car nous avons déjà une borne inférieure globale finie qui est zéro. Certains exemples et

méthodes de réduction de la dépendance peuvent être trouvés dans [58] [59]. De manière générale nous pouvons retenir que L'AI garanti un résultat global au prix d'une surestimation des résultats

4.1.2 Propriétés et définitions de bases nécessaires à L'algorithme S&E du chapitre 5

Dans le but de résoudre le problème d'optimisation globale proposé dans cette thèse, la proposition, les définitions et le théorème qui suivent sont nécessaires :

Lemme : considérons la fonction d'intervalle $F(X_1, X_2, \dots, X_n)$, alors

$$F(Y_1, Y_2, \dots, Y_n) \subset F(X_1, X_2, \dots, X_n) \text{ pour tout intervalle } Y_1, Y_2, \dots, Y_n \text{ avec } Y_k \subset X_k \text{ pour } k=1 \dots n.$$

La proposition ci-dessus dont la démonstration peut être retrouvée dans [59] énonce que la monotonie d'une fonction évaluée en utilisant les opérations arithmétiques par intervalles telles que définies ci-dessus est inclusive. Cette propriété d'inclusion est importante pour borner des fonctions et joue un rôle important dans la construction de la fonction d'inclusion de l'algorithme **S&E** présenté au chapitre 5.

Définition 1 : On appelle une représentation fixe d'une fonction de n variables $f(x_1, x_2, \dots, x_n)$ la fonction $f(x_1, x_2, \dots, x_n)$ dont les variables x_1, x_2, \dots, x_n n'apparaissent qu'une seule fois dans l'expression de f .

Exemple 1 : considérons les fonctions suivantes :

$f_1(x) = 4(x^2 - x)$ et $f_2(x) = (2x - 1)^2 - 1$. Bien que la fonction $f_1(x) = f_2(x)$ $x \in IR$ $F_1(X) \neq F_2(X)$ $x \in X$. L'exemple 2 ci-dessous vous en convaincra. La fonction $f_2(x)$ est une représentation fixe en la variable x dans son expression tandis que la fonction $f_1(x)$ n'en est pas. La variable x apparaît à plusieurs reprises dans l'expression de la fonction $f_1(x)$.

Définition 2 : l'extension par intervalle d'une fonction de n variables $f(x_1, x_2, \dots, x_n)$ est la fonction d'intervalle $F(X_1, X_2, \dots, X_n)$ dont les opérations sur les fonctions arithmétiques et transcendantales de f sont remplacées par leur équivalente d'intervalle tels que définies ci-dessus (équation 2.25 à 2.24).

L'extension par intervalle a pour résultat une borne minimale et maximale générale. Ce résultat est d'intérêt dans un algorithme de type **S&E** comme nous le verrons au chapitre 5. Notons que, la plage des valeurs prises par une fonction f peut être exprimée sous forme d'intervalle comme :

$f(X) = [f(\bar{x}), f(\underline{x})]$. Cependant, $F(X)$ est la fonction d'inclusion de X calculée en utilisant l'extension par intervalle et l'arrondie. Par conséquent, $F(X)$ contient alors la plage des valeurs de $f(X)$ ainsi que les erreurs d'arrondies et toutes les autres erreurs numériques incluant le phénomène de dépendance. L'exemple 4 ci-dessous permettra d'illustrer la définition 2 et dégager certaines propriétés utiles au chapitre 5.

Exemple 2: Considérons les fonctions $f_1(x) = 4(x^2 - x)$ et $f_2(x) = (2x - 1)^2 - 1$. Leur extension par intervalle donne : $F_1(X) = 4(X^2 - X)$ et $F_2(x) = (2X - 1)^2 - 1$. Si nous faisons varier $X \in [0, 2]$ on obtient : $F_1(X) = [-8, 16]$ et $F_2(X) = [-1, 8]$. La plage $f_2(X) = [f(0), f(2)] = [0, 8]$.

Observons de cet exemple que: $F_2(X) \subset F_1(X)$ et $F_2(X) = f(X)$. Le premier résultat montre que $[-1, 8]$ est plus compacte que celui avec $[-8, 16]$ car l'expression de $f_2(X)$ est une représentation fixe en la variable x . le deuxième résultat nous montre que lorsque f à une représentation fixe alors le phénomène de dépendance est moins présent et dans certains cas pas du tout.

Définition 3: Une fonction rationnelle d'intervalle est la fonction à valeur d'intervalle dont les valeurs sont définies par une séquence spécifique et finie d'opérations d'intervalle arithmétiques.

Exemple 3: soit la fonction de deux variables $F(X_1, X_2) = \exp(2X_1 + \ln(X_2))$, Le calcul de F peut être coupé en petite séquence d'opération arithmétique d'intervalle tels que présenté ci-dessous :

$$T_1 = [2,2] \times X_1 \quad \text{Une multiplication d'intervalle}$$

$$T_2 = T_1 + \ln(X_2) \quad \text{Une addition d'intervalle}$$

$$T_3 = F(X_1, X_2) = \exp(T_2) \quad \text{Une exponentielle d'intervalle}$$

Définition 4 : Une fonction $F(X_1, X_2, \dots, X_n)$ est inclusivement isotone si pour tout intervalle $Y_i \subset X_i$ pour $i=1 \dots n$ $F(Y_1, Y_2, \dots, Y_n) \subset F(X_1, X_2, \dots, X_n)$

Théorème 1 : soit $F(X_1, X_2, \dots, X_n)$ une fonction rationnelle d'intervalle évaluée en utilisant l'extension par intervalle. Si F est évaluée en utilisant sa représentation fixe par des opérations d'intervalles impliquant uniquement l'addition, la soustraction, la multiplication et la division alors F est une fonction isotonique inclusive. Pour la preuve voir [59].

4.2 L'implémentation sur ordinateur de la librairie des fonctions de bases

L'arithmétique par intervalles est une méthode qui consiste à évaluer des nombres en les remplaçant par des intervalles. Le choix de cette arithmétique pour des problèmes d'optimisation résulte de la fiabilité des résultats qu'elle permet d'obtenir. En effet, l'AI garanti un résultat global (au prix d'une surestimation des résultats dans certaines conditions) en intégrant dans ses calculs les erreurs d'arrondis liées aux opérations à virgule flottante. À titre d'exemple, une représentation sur deux chiffres après la virgule pour π est $\pi = [3.14; 3.15]$. Pour mettre à profit cette propriété de calcul par intervalle en utilisant l'arithmétique idéale disponible sur ordinateur, nous avons utilisé l'arrondi tels que spécifié par la norme IEE-754. Quatre modes d'arrondies ont été implementés : l'arrondi vers l'infini par valeur négative ($-\infty$), l'infini par valeur positive ($+\infty$), zéro par valeur négative (-0), et zéro par valeur positive ($+0$). Des routines en langage assembleur ont été implémentées afin que les changements du registre qui contrôle les différents modes d'arrondis s'opèrent de façon rapide et efficace. Pour chaque opération dont le résultat appartient à la borne inférieure, un changement de contexte du processeur pour un arrondit des calculs vers $-\infty$ est appliqué. Le même phénomène est appliqué pour les opérations faites sur la borne supérieure. Pour ce dernier, l'arrondi est vers $+\infty$. Dans le cas des fonctions transcendentales, les propriétés liées à la monotonie des différentes fonctions accompagnées de l'arrondi sur les opérations arithmétiques ont été appliquées. Une librairie complète regroupant les fonctions arithmétiques et transcendentales pour l'ensemble des fonctions et opérations arithmétiques de la Delta-Lognormale a été conçue en C++ dans le cadre du présent projet.

4.3 Théorème fondamentale de l'AI

Théorème 2 : (théorème fondamentale) Soit $F(X_1, X_2, \dots, X_n)$ l'extension par intervalle de $f(x_1, x_2, \dots, x_n)$ alors $f(X_1, X_2, \dots, X_n) \subseteq F(X_1, X_2, \dots, X_n)$, et pour tout intervalle

$Y_k \subseteq X_k$, pour $k = 1, \dots, n$, $f(Y_1, Y_2, \dots, Y_n) \subseteq$

$F(Y_1, Y_2, \dots, Y_n)$, where $f(Y_1, Y_2, \dots, Y_n) \{ f(x_1, x_2, \dots, x_n) : x_k \in Y_k \text{ for } k = 1, \dots, n \}$.

Ce théorème dû à Moore [58] a été étendu par Hansen [59]. La preuve peut être trouvée dans [59]. Le théorème stipule que, l'évaluation d'une fonction en tout point à l'intérieur d'un espace (intervalle

ou vecteur d'intervalle) fermé ou la fonction et ses variables sont définies est garantie de produire un résultat contenu dans l'extension par intervalle F de la fonction f .

Les équations **4.1** à **4.7** et les différentes propriétés, définitions et théorèmes énoncés ci-dessus sont les outils de base nécessaires à l'élaboration et la compréhension de l'algorithme **S&E** proposé au chapitre 5. Notons que des définitions ou notations plus spécifiques seront présentées dans les sections où elles sont nécessaires, mais pour l'instant celles qui sont proposées ci-haut forment l'essentiel et la base de notre raisonnement.

CHAPITRE 5 ARTICLE 4: GLOBAL OPTIMIZATION ALGORITHM FOR THE ESTIMATION OF THE DELTA-LOGNORMAL PARAMETERS CHARACTERIZING POINTING GESTURES

Simon Pierre Boyogueno Bidias, Jean-Pierre David, Réjean Plamondon, *Life Fellow, IEEE*, Yvon Savaria, *Fellow, IEEE*

This Paper is under review in the journal « Pattern Analysis and Machine Intelligence » (PAMI) since march (2022).

The paper has been submitted on March 14, 2022.

Abstract— This paper proposes an efficient n-dimensional global optimization algorithm to compute the parameters of a model enclosing a multivariate Delta-Lognormal function with a specified maximum error. The method is based on a branch and bound algorithm, constrained in a way that takes advantage of all the information available about the objective function. The basic idea behind this new algorithm is to exploit intervals arithmetic to bound the Delta-Lognormal function and its gradient. These bounded functions are then used in several ways in a global branch and bound optimization algorithm. The different rules accompanying each step of the proposed algorithm speed up the pruning process with the guarantee of not losing the global solution. The accuracy of the extracted parameters is quantified automatically, taking advantage of the bounding box returned by the algorithm as a result. The new tool preserves confidence intervals on the extracted parameters, which might be particularly useful in medical applications, where the model could help to characterize some neuromotor diseases. The methodology and a wide variety of gesture models to be extracted is described, evaluated, and compared with two previously reported suboptimal extractors.

Index Terms— Branch and bound, Interval arithmetic, Delta-Lognormal, Parameter extraction, Kinematic theory, Pattern recognition, Artificial intelligence, Handwriting movement, Data fitting, Non-linear regression.

5.1 Introduction

Human movements can be very complex and their studies in terms of motor control is intensively used in many fields of research. For example, in the context of designing Personal Digital Bodyguards (PDBs) [1], mobile devices such as tablets or smartphones, combined with robust algorithms, aim at supplementing people's sensitive data protection with signature verification, (e-Security), enhancing human-machine interaction performances through words spotting and handwriting recognition (e-Recognition). For young children, these tools can turn into interactive toys helping to learn and master their fine motor control (e-Learning). Moreover, PDBs can also provide users with fine motor control monitoring, which can detect stress, aging and health problems (e-Health). From a broader perspective, motor control research is a corner stone that can have an impact in anthropomorphic robot and arm design, exoskeletons and prosthetics, human-like avatar gesture, gait recognition, etc. At the basis of these investigations stands the concept of handwriting or gesture primitives, simple pointing movements from which trajectories are built. Studies of this specific class of movements in terms of motor control has a long history and is still intensively exploited in many fields of research. Indeed, simple pointing movements have been used to study speed/accuracy trade-offs in target-directed movement for more than a century [60], [61] and among the main contributions, Fitts (1954) [62] proposed a formal relationship linking movement time to the relative spatial error. In subsequent years, pointing movements have been widely used in many psychometric reaction time tests. [63], [64], [65].

In this paper, we use this theory to get a better insight on single pointing gesture production. We propose a global optimization algorithm to extract Delta-Lognormal parameters of a single handwritten pointing gesture. The aim is to get the unique set of Delta-Lognormal parameters that best fit a given handwritten pointing gesture.

- *The authors are with the department of electrical engineering Polytechnique Montréal, Montréal, Canada H3C3A7. E-mail {Simon-pierre.boyogueno-bidias, yvon.savaria, rejean.plamondon, jpdavid@polymtl.ca} @polymtl.ca*

In this context, the Kinematic Theory of Human Movements (**KTHM**) [19] has been used to analyze the temporal behavior of the motor system of a human during the accomplishment of a movement. This theory presents lognormal handwriting strokes as primitives that contain information on both the central motor controller and the peripheral neuromuscular systems involved in their production [14]. The Delta-Lognormal ($\Delta\Lambda$) model derived from this theory proposes a mathematical equation describing the agonist-antagonist interaction as well as the set of temporal properties of all the neuromuscular components involved in the accomplishment of a

pointing movement. This theory, because of its ability to consider highly complex cortical cerebellar and neuromuscular properties, has proven to be one of the best in reproducing the various invariants observed in rapid pointing movements.

In this perspective, different battery of tests involving simple pointing movements have been developed [66]. and used in various applications for example to study the handwriting learning in children [67], children with **ADHD** problems [18], to track athlete performances, to detect neuromuscular fatigue [68], to monitor the evolution of normal aging of Parkinson disease [69] and the effects of brain stroke risk factors [70], to name a few. Moreover, suboptimal lognormal parameters extractor have been developed [22] , [48] , [47] , [71], and are used worldwide in numerous applications but. Except for [12], none of them can guarantee the optimality of the obtained solutions. This is not always a critical requirement for some of the above-mentioned applications where very good curve fitting may suffice. However, globally optimal solutions are important in cybersecurity, robotics, and biomedical applications, where a confidence interval on the values of extracted parameters is necessary. In this paper, we propose and validate an algorithm that extracts the global and specific timing properties of a handwriting pointing gesture. The features extracted from the pointing gesture primitives allow to reconstruct its velocity profile and represent the planning and timing used by the human who accomplish that movement [19] [14]. As this algorithm offers guaranteed accuracy, it could serve, among other things, as a benchmark for comparing sub-optimal algorithms.

The proposed algorithm is based on interval arithmetic in combination with the branch and bound method. The goal is to output a unique bounding box, certainly and safely enclosing the solution, resulting from a much wider but finite search space, which provides the combination of $\Delta\Lambda$ parameters that best fit the velocity profile of a simple agonist-antagonist synergy.

This new algorithm solves problems of existing extractors by producing a unique high-quality solution, with a processing time sufficiently short for practical applications. The accuracy of the extracted parameters is quantified automatically, taking advantage of the bounding box returned by the algorithm. This bounding box contains the global solution in a confidence interval such that the users can set the fitting accuracy as needed. New tools could then exploit the confidence interval on the extracted parameters to address the wide range of applications where the model can serve.

The rest of the paper is organized as follows. Section 2 presents the $\Delta\Lambda$ equation and the modeling tools related to the objective function to be minimized. Section 3 briefly reviews the status of $\Delta\Lambda$ parameter extraction methods, while Section 4 presents some theoretical aspects of interval arithmetic needed in the proposed algorithm. Section 5 presents and described the global optimization with some examples. The database used to test the algorithm is presented in section 5, while numerical results and performance comparisons with [50] and [22] are discussed in Section 6. Section 7 summarizes our main contributions and proposes future research.

5.2 The Delta-Lognormal model

The $\Delta\Lambda$ model describes how a synergetic system, made of a pair of agonist and antagonist subsystems involved in the production of a rapid human movement, can control that movement over time. The agonist and antagonist

The $\Delta\Lambda$ model describes how a synergetic system, made of a pair of agonist and antagonist subsystems involved in the production of a rapid human movement, can control that movement over time. The agonist and antagonist subsystems represented by the subscripts 1 and 2 in (5.1) have lognormal impulse responses [14]. When convolved with their corresponding input commands, it leads to:

$$\Delta\Lambda(t; P) = D_1\Lambda(t; t_0, \mu_1, \sigma_1^2) - D_2\Lambda(t; t_0, \mu_2, \sigma_2^2) \quad (5.1)$$

where

$$\Lambda(t; t_0, \mu, \sigma^2) = \begin{cases} \frac{1}{\sigma\sqrt{2\pi}(t-t_0)} \exp^{-\frac{(\ln(t-t_0)-\mu)^2}{2\sigma^2}}, & \text{for } t_0 < t, \\ 0, & \text{elsewhere.} \end{cases} \quad (5.2)$$

With

$$\sigma > 0; \mu, t_0 \in \mathbb{R}$$

$$P = [t_0, D_1, \mu_1, \sigma_1, D_2, \mu_2, \sigma_2] \quad (5.3)$$

In (5.2), t_0 represents the system activation time; $\mu_1, \sigma_1, \mu_2, \sigma_2$ correspond to the timing properties on a logarithmic scale of the two neuromuscular systems, in reaction to their input command amplitudes D_1 and D_2 respectively. Extracting the model parameters from velocity data means finding the set of seven parameters P (5.3) that best fit the original velocity profile $v_t(t)$ (collected

from human movements). This extraction is done by minimizing a nonlinear least square function f (5.4) that maximizes the signal to noise ratio (SNR) (5.5).

$$f = \int (v_t(t) - \Delta\Lambda(t; p))^2 dt \quad (5.4)$$

$$SNR = 10 \log \left(\frac{\int v_t^2(t) dt}{f} \right) \quad (5.5)$$

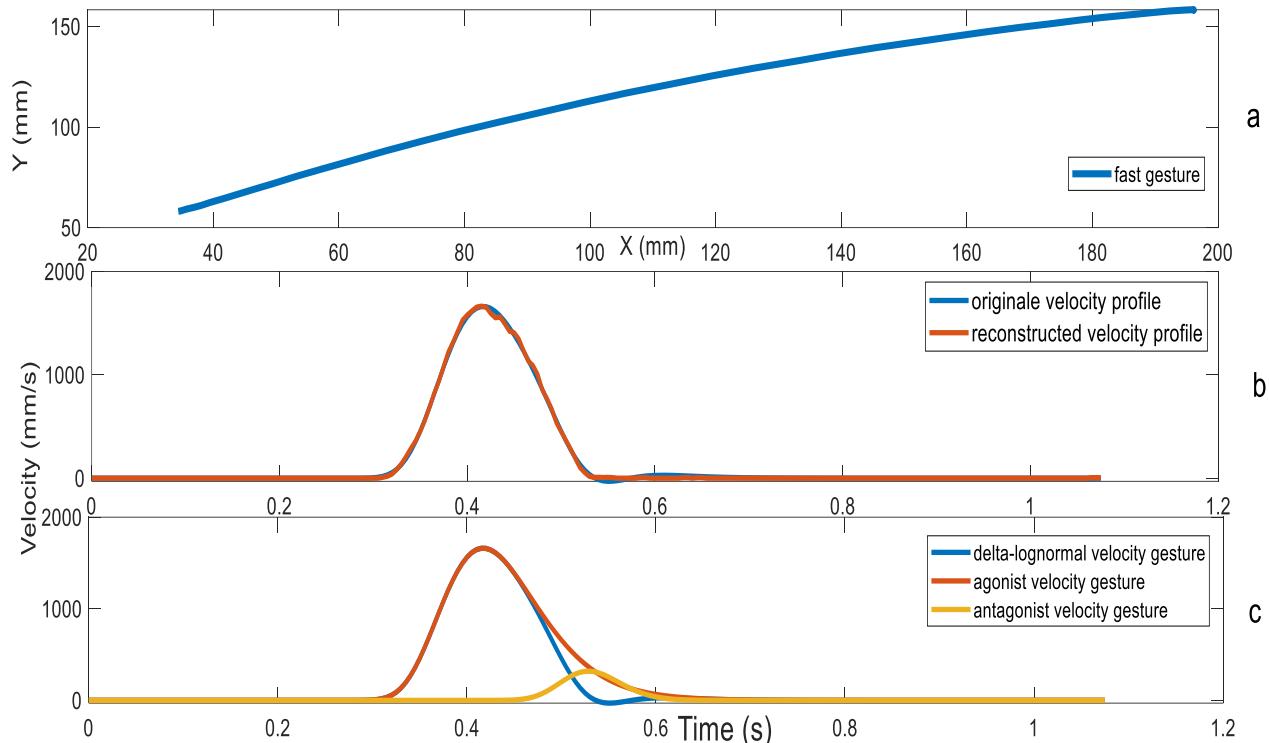


Figure 5.1 A human handwriting pointing gesture: a. the (x, y) position of the pen tip movement, b. the real and reconstructed velocity profile with a 31dB SNR, c. the reconstructed velocity profile decomposed into agonist and antagonist contributions

An example of such an extraction is shown in Figure 5.1. The $(x(t), y(t))$ position of the pen tip movements as collected with a digitizer is depicted in Figure 5.1a, followed by its velocity profile (Figure 5.1b), and the reconstructed profile produced by our algorithm is reported in Figure 5.1c. In the context of the Kinematic Theory of human movement, a $\Delta\Lambda$ profile is assumed to result from the agonist–antagonist interaction as observed for example in a pointing gesture.

5.3 State-of-the-art Delta-Lognormal extractors

algorithms proposed to solve the problem of extracting the $\Delta\Lambda$ parameters from handwriting pointing gesture. Given the large range of applications in handwriting recognition that the model can serve, various algorithms have been proposed over the past two decades. The approaches are grouped into two broad categories: evolutionist and deterministic¹. The basic idea behind these algorithms is the adjustment of the velocity profile of a simple handwriting movement, until the seven parameters characterizing a $\Delta\Lambda$ velocity profile are obtained.

The evolutionary method, as proposed in [49], is a genetic algorithm in which each gene corresponds to a lognormal parameter generated based on two genetic operators namely mutation and recombination. As presented in [49], this method does not work well for all sets of velocity profiles that the model proposes. Moreover, the heuristic base of the method can compromise the guarantees of recovering the optimal solution.

In [48], [22], [51], three deterministic algorithms more related to the geometry of the problem are proposed. They estimate the lognormal parameters by relying on certain characteristic points of the speed profile of a handwriting gesture. Based on the results of [49], this approach seems to be more robust than the evolutionary method and was thus adopted as the kernel of many algorithms found in the literature for $\Delta\Lambda$ parameters extraction.

The first algorithm developed following this reasoning is **INFLEX** [48]. It is based on an estimation module, derived from a graphical method [51] that links the lognormal function to different characteristic points of a handwritten velocity profile. This step of the method is followed by a Levenberg Marquardt nonlinear optimization Marquardt [44] .The main advantage of **INFLEX** is the simplicity and rigor of its geometric modeling. Despite that, it could not provide solutions for all types of existing lognormal speed profiles. Its primary limitation is the identification of the agonist versus the antagonist components of a velocity profile to estimate the

¹ Moreover, there is a statistical approach not relying on the geometry of the problem that has been presented in in **Source spécifiée non valide**. to process lognormal functions, but this approach has never been adapted to the extraction of Delta-Lognormal parameters.

parameters of a movement which is based on the hypothesis that the first peak is exclusively agonist, is still open question.

To mitigate those issues, the **INITRI** algorithm has been designed [47]. This algorithm, even though it complements **INFLEX** very well, has the disadvantage of underestimating the information related to the parameter (σ_1). Finally, the **Xzero** algorithm detailed in [22] has been developed. It uses an analytic expression based on the relationship between the zero crossings of the first and second derivatives of a lognormal profile of the total speed profile. This procedure avoids the underestimation of the parameter σ_1 (as in **INITRI**) and the choice of a tangent for graphical modelling (as in **INFLEX**) that can be inaccurate. To take advantage of the various algorithms, a system named **IIX** [22] whose objective is to combine the efficiency of the three extractors into one has been elaborated.

Although these prior extractors show good curve fitting, the global optimality of the solutions obtained is not guaranteed. To alleviate these problems, a branch and bound algorithm that can offer rigorous global optimization has been reported in [50]. This prior work is used in our test as the base line. It is mainly based on building a $\Delta\Lambda$ envelope that bounds all the feasible $\Delta\Lambda$ parameters into a subspace and evaluates a bound on the error between the signal and the nearest envelope. This method, even though it clearly circumscribed the challenge and bounds the solution space, suffers from lower bound estimation problems. According to [50], the conditions on which the procedure. For computing an envelope to the velocity profile do not always hold. Moreover, calculation of the estimated area of the envelope requires very long processing time.

In the present paper, we propose a novel approach in which we tackle lognormal parameter extraction using interval arithmetic combined with a branch and bound algorithm. To the best of our knowledge, there is no previous algorithm based on the concept of interval arithmetic used in combination with a branch and bound procedure that has been applied to the $\Delta\Lambda$ function. The mathematical properties developed to construct the search space as well as the evaluation method of the various functions is unique in the state of the art on extraction of lognormal parameters for handwritten gesture. The proposed algorithm produces solutions within a confidence interval that avoids relying on statistical methods to construct such an interval.

5.4 Definitions, notations, and fundamental theorem.

Before starting to detail the proposed algorithm, some definitions and notations must be presented to support it. We denote by $I = [a, b] = \{x \in \mathbb{R} : a \leq x \leq b\}$ the set of real numbers comprised in a one-dimensional closed interval. An interval is denoted by a variable in upper case, while the set of values contained in the interval is denoted by variables in lower case. An uppercase variable put in bold is an interval vector. Thus, $X = [\underline{x}, \bar{x}]$ is an interval variable where \underline{x} and \bar{x} represent the left and right ends of X respectively. The width of X is defined and denoted by $w(X) = \bar{x} - \underline{x}$ and the midpoint is given by $m(X) = 1/2(\bar{x} + \underline{x})$. We refer to $\mathbf{X}^I = (X_1, X_2, \dots, X_n)$ as the set of n dimensional interval vectors or hyper box such that:

$$\mathbf{X}^I = (X_1, X_2, \dots, X_n) = ([\underline{x}_1, \bar{x}_1], [\underline{x}_2, \bar{x}_2], \dots, [\underline{x}_n, \bar{x}_n]).$$

We define the width and the midpoint of an interval vector as:

$$w(\mathbf{X}^I) = \max_i w(X_i^I)$$

and

$$\mathbf{m}(\mathbf{X}^I) = (\mathbf{m}(X_1), \mathbf{m}(X_2), \dots, \mathbf{m}(X_n)).$$

The basic interval arithmetic operations as well as the one-variable transcendental and rational interval functions operations are described in Table 5.1. Further details can be found in [72] [58] [59].

Table 5.1 Basic interval arithmetic operations and functions for the Delta-Lognormal

Operations	Results
$X + Y$	$[\underline{x} + \underline{y}, \bar{x} + \bar{y}]$
$X - Y$	$[\underline{x} - \bar{y}, \bar{x} - \underline{y}]$
$X \times Y$	$[\min S, \max S]$ with $S = \{\underline{x}\underline{y}, \underline{x}\bar{y}, \bar{x}\underline{y}, \bar{x}\bar{y}\}$
$X \div Y$	$X \times \left(\frac{1}{Y}\right)$, with $\left(\frac{1}{Y}\right) = \left[\frac{1}{\bar{y}}, \frac{1}{\underline{y}}\right]$ and $0 \notin Y$
Functions	Results
$\exp(X)$	$[\exp(\underline{x}), \exp(\bar{x})]$

$\ln(X)$	$[\ln(\underline{x}), \ln(\bar{x})]$ for $0 < \underline{x}$
$X^{1/2}$	$[\underline{x}^{1/2}, \bar{x}^{1/2}]$
X^n	$\begin{cases} [\underline{x}^n, \bar{x}^n] & \text{if } \underline{x} > 0 \text{ or } n \text{ is odd} \\ [\bar{x}^n, \underline{x}^n] & \text{if } \bar{x} < 0 \text{ and } n \text{ is even} \\ [0, \max X^n] & \text{if } 0 \in X \text{ and } n \text{ is even} \end{cases}$

Definition 1: An interval function is defined as the range resulting from the set of interval arguments of a given function.

Definition 2: The natural interval extension (NIE) of a given function $f(x_1, x_2, \dots, x_n)$ of n variables is given by the interval function $F(X_1, X_2, \dots, X_n)$, which is obtained by replacing the real variable x with the corresponding interval variable X .

Example 1: The NIE of the non-convex function $f(x, y) = (1 - x)^2 + 100(y - x^2)^2$ for $X = [-1, 2], Y = [-1, 3]$ is $F(X, Y) = (1 - X)^2 + 100(Y - X^2)^2 = [0, 2504]$. If we take any interval, say $X' = [-1, 1] \subseteq X$, the value of $F(X', Y) = [0, 904]$ is contained in $F(X, Y)$. This important property for bounding operations using the NIE is called inclusion isotonicity.

Definition 3: An interval extension function F is said to be Lipschitz in an interval X_0 if there is a constant K such that $w(F(X)) \leq Kw(X)$ for $X \subseteq X_0$.

Lemma: If G and H are interval extension with G Lipschitz in X_0 , H Lipschitz in Y_0 and $G(Y_0) \subseteq X_0$. Then the composition $F(X) = G(H(X))$ is Lipschitz in Y_0 and its inclusion is isotonic.

Proof: $w(F(X)) = w(G(H(X))) \leq k_1 w(H(X)) \leq k_1 k_2 w(X)$.

This Lemma shows that, an interval extension whose values are defined by a finite sequence of rational interval arithmetic operations or composition of unary Lipschitz functions is Lipschitz in some region.

Example 2: let $g(x) = \exp(x)$ and $h(x) = 2x + \ln(x)$ then $goh = f(x) = \exp(2x + \ln(x))$ for $X = [1, 3]$ we have $w(F(X)) = w([7.39; 1210.29]) = 600.45 * w([1, 3])$ and similarly the sequence $w(G(H(X))) \leq 235.93 * w(2X + \ln(X)) \leq 235.93 * 2.55 * w([1, 3])$ is equal to $w(F(X))$.

We see through the example that a compound function can be decomposed into its different elementary functions and evaluated using the NIE to get the inclusion, while maintaining its properties.

Following the previous observations, we now state the so-called Fundamental Theorem of interval arithmetic.

Fundamental Theorem. *Let $F(X_1, X_2, \dots, X_n)$ be the natural interval extension of $f(x_1, x_2, \dots, x_n)$ then $f(X_1, X_2, \dots, X_n) \subseteq F(X_1, X_2, \dots, X_n)$, and for all intervals, $Y_k \subset X_k$, for $k = 1, \dots, n$, $f(Y_1, Y_2, \dots, Y_n) \subseteq F(Y_1, Y_2, \dots, Y_n)$, where $f(Y_1, Y_2, \dots, Y_n) \{ f(x_1, x_2, \dots, x_n) : x_k \in Y_k \text{ for } k = 1, \dots, n \}$.*

This theorem due to Moore [58] was extended by Hansen [59]. It states that, the evaluation of a function in any point inside a closed interval where both the function and its variables are defined is guaranteed to produce a result contained in the natural interval extension of the function. The proof can be found in [59]. We apply this theorem to the $\Delta\Lambda$ function as a specific sequence of interval arithmetic operations to bound the range of (4) within an interval vector.

5.5 The multidimensional Interval Arithmetic Branch and Bound

Delta-Lognormal Parameter Extractor:

The branch and bound technique is often used to seek an optimal solution in a combinatorial set of possible ones. The main idea behind this algorithm is to build a tree of potential candidates among which a search is performed to keep the best one. In this work, the search is carried out by browsing a tree of solutions, while removing at each step the candidates that cannot minimize an objective function. Following this idea, we seek to minimize the function f , expressed by (5.4) to obtain a SNR expressed by (5.5) that depends on the type of experiment (see section 6) for the different thresholds. That is, we seek to find the best $\Delta\Lambda$ parameters that fits an input velocity profile inside a defined finite space.

Note that defining what constitutes an optimal solution is not so simple, but in loose terms, for medical doctor or in a forensic or cybersecurity task for example, to accept extracted parameters as part of a diagnosis process requires a level of certainty such that there is no other good enough solution that departs significantly from the one provided by a parameter extraction system. Thus,

the operational optimal solution is then a bounding box that guarantees the inclusion of the total observed input speed profile of a handwritten pointing gesture, that contain all the potential solutions with a **SNR** at least equal to a user specified target. The global optimization problem that is considered is the following:

Minimize $f(x)$ subject to $x \in \mathbf{X}^I$, where f is an n dimensional continuously differentiable function $\mathbb{R}^N \rightarrow \mathbb{R}$ and $\mathbf{X}^I \subseteq \mathbb{R}^N$ is an n dimensional interval vector. Thus,

$$\begin{aligned}\mathbf{P}^I &= \left\{ \left[\underline{D}_1, \bar{D}_1 \right], \left[\underline{\mu}_1, \bar{\mu}_1 \right], \left[\underline{\sigma}_1, \bar{\sigma}_1 \right], \left[\underline{D}_2, \bar{D}_2 \right], \left[\underline{\mu}_2, \bar{\mu}_2 \right], \left[\underline{\sigma}_2, \bar{\sigma}_2 \right], \left[\underline{t}_0, \bar{t}_0 \right] \right\} \\ &= \{D_1, M_1, \Sigma_1, D_2, M_2, \Sigma_2, T_0\}\end{aligned}$$

is the bounding space of our objective function. Its variables vector is:

$$p = [t_0, D_1, \mu_1, \sigma_1, D_2, \mu_2, \sigma_2].$$

Therefore, the inclusion function of f and its gradients f' are:

$$\begin{aligned}f(\mathbf{P}^I) &\in F(\mathbf{P}^I) = [\underline{F}(\mathbf{P}^I), \bar{F}(\mathbf{P}^I)] \\ &= \int (v_t(t) - \Delta\Lambda(t; \mathbf{P}^I))^2 dt. \\ f'(\mathbf{P}^I) &\in \nabla F(\mathbf{P}^I) = F'(\mathbf{P}^I) = \frac{\partial f(\mathbf{P}^I)}{\partial \mathbf{P}_i^I} \quad i = 1..7,\end{aligned}$$

The information available during the search is:

$$f(m(\mathbf{P}^I)), F(\mathbf{P}^I) \text{ and } F'(\mathbf{P}^I)$$

Now that we have these definitions and notations, we can use interval arithmetic [58], [59] to bound the $\Delta\Lambda$ function and find the solution to the global optimization problem stated in (5.4), (5.5). The proposed algorithm is called **IAB&BPE**, which stands for Interval Arithmetic Branch and Bound $\Delta\Lambda$ Parameter Extractor. **IAB&BPE** is formulated as follows:

IAB&BPE:

1. Set the initial parameter \mathbf{P}^I **initial rule.**
 2. Set the working list $L_1 := \{ \mathbf{P}^I \}$, the final list $L_2 := \{ \}$ and the upper bound $\bar{f}^\sim = f(m(\mathbf{P}^I))$
- While ($L_1 \neq \{ \}$)

3. Select optimal box P from L_1 **Selection rule**

4. Bisect P^I into P_i^I sub boxes $i = 1..2$ **Splitting rule**

For $i = 1$ to 2

5. Compute the bound $f(m(P_i^I)), \underline{F}(P_i^I)$ **Bounding rule**

6. If $\left(\left(\underline{F}(P_i^I) < \bar{f} \right) \text{ and } \left(0 \in \nabla F(P_i^I) \right) \right)$ **Discarding rule**

7. $\bar{f} = \min \{ \bar{f}, f(m(P_i^I)) \}$; **Update upper bound** for f^*

8. If P_i^I satisfy the ending criterion **Termination rule**

$L_2 += \{ \underline{F}(P_i^I), P_i^I, \nabla F(P_i^I) \}$;

9. Else $L_1 += \{ \underline{F}(P_i^I), P_i^I, \nabla F(P_i^I) \}$; **Storing rule**

10. Return L_2, f^*

As presented above, IAB&BPE takes as input a feasible bounding-box P , a working list L_1 , a final list L_2 , the objective function f (5.4) and the upper bound \bar{f} of the cost function computed by evaluating (5.4) at the midpoint of the initial box P . The feasible domain is restricted to an initial search region wider than necessary to almost certainly safely enclose the solution. The working list is the tree generated throughout the search by the division of boxes into sub-boxes. The final list contains the solutions that meet the ending criteria presented in step 9 of the algorithm. The algorithm starts by bisecting a box (see step 4) and computing the bounds of each resulting sub-box as proposed in [57]. In step 6, the integral is computed numerically as proposed in [73] [72]. Next, the sub-boxes previously obtained are subjected to different discarding tests (see step 7) to validate if they contain, in their intervals, the current upper bound or a stationary point. If it is the case, the box is kept in the working list and the process continues. Otherwise, the box is rejected, because the test has just proven that the box cannot contain a better solution than the current best one. The stopping criterion is applied in step 9, where we evaluate (5.5) for each box that passed the discarding tests. We put in the list L_2 boxes that have satisfied a SNR of 100 dB for ideal synthetic data and 25dB for real velocity profiles. In Step 8 and 9, the subscript $+=$ serve to specify a push into a list.

IAB&BPE applies different rules based on the information available a priori and on the one obtained about the objective function during the search. Equations (5.6) and (5.7) provide the inclusion function and its gradient respectively. With the objective function defined in (4) and the information stated in (5.8), the rules at the core of our general global optimizer are as follows:

The initial rule: For simplicity, we restrict the search region to a finite box where we know that the solution can be found. The interval values of a bounding box are chosen wide enough to ensure that in practice they do not constrain the final solution. The bounds for each parameter defining the bounding box are constructed as follows:

- For synthetic or ideal velocity profiles, the initial intervals are derived from the parameters used to construct them.
- For real (human velocity profile) curves, the initial intervals are obtained from a local optimizer (**IIX**) that gives the middle value of the bound for each parameter.

In both cases, the initial interval for each parameter is established by creating a box that extend the estimated solution up and down by a multiple of an estimated weighted standard deviation. For

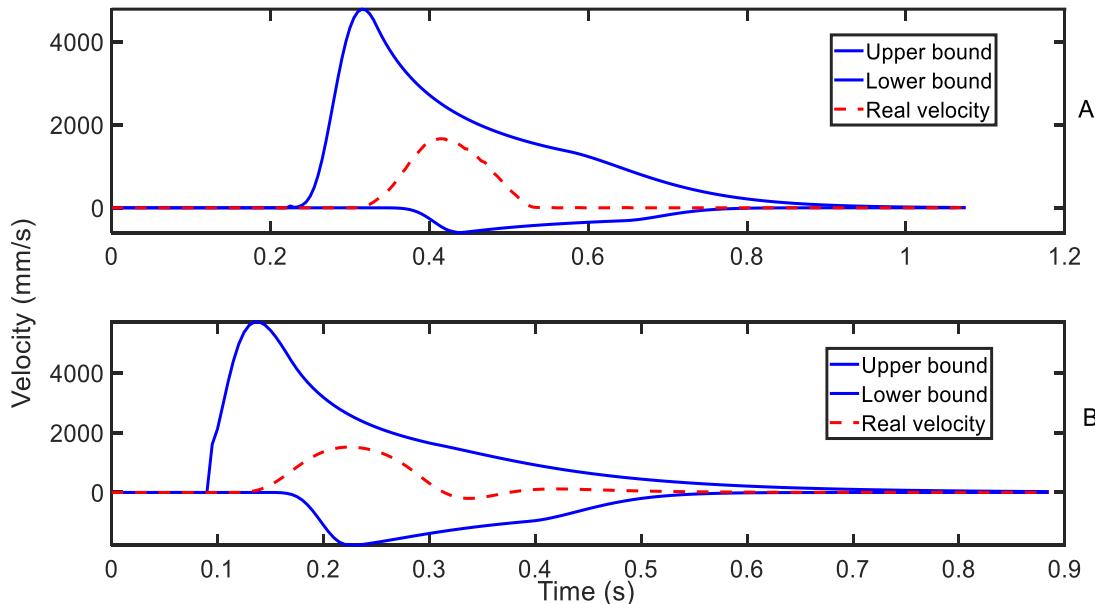


Figure 5.2 Bounding box derived from the initial rule: A. Real velocity profile enclosed by its envelope computed using the NIE. B. An ideal velocity profile enclosed by its envelope computed using the NIE

example, the resulting interval for the parameter μ_1 is $[\mu_1 - K * \text{std} ; \mu_1 + K * \text{std}]$ with μ_1 being the central value and std the standard deviation obtained according to (5.9).

$$\text{std}_i = \alpha e^{\beta \text{SNR}_i} \quad (5.9)$$

In (5.9), the coefficient α and β are the regression coefficients and the SNR_i is the SNR obtained by the local optimizer (IIX) to get the input parameter p_i . The methodology to derive (5.9) can be found in [22]. To obtain the bounding Box, the different intervals obtained are thus iterated K times (K is an integer) until at least the entire speed profile to be adjusted is included in the envelope produced by the resulting bounding box. In practice we take $K = K^* + m$ where K^* is the constant value of the box that strictly bounds the original signal and m is a positive integer specified by the user. When creating the bounding box, the operations in $\Delta\Lambda(t; \mathbf{P}^I)$ are calculated using the NIE.

Figure 5.2 presents an experimental verification of the initial box for an ideal and a real velocity profile. The envelope (plotted in thick blue line) bounding the ideal velocity (dash red) are

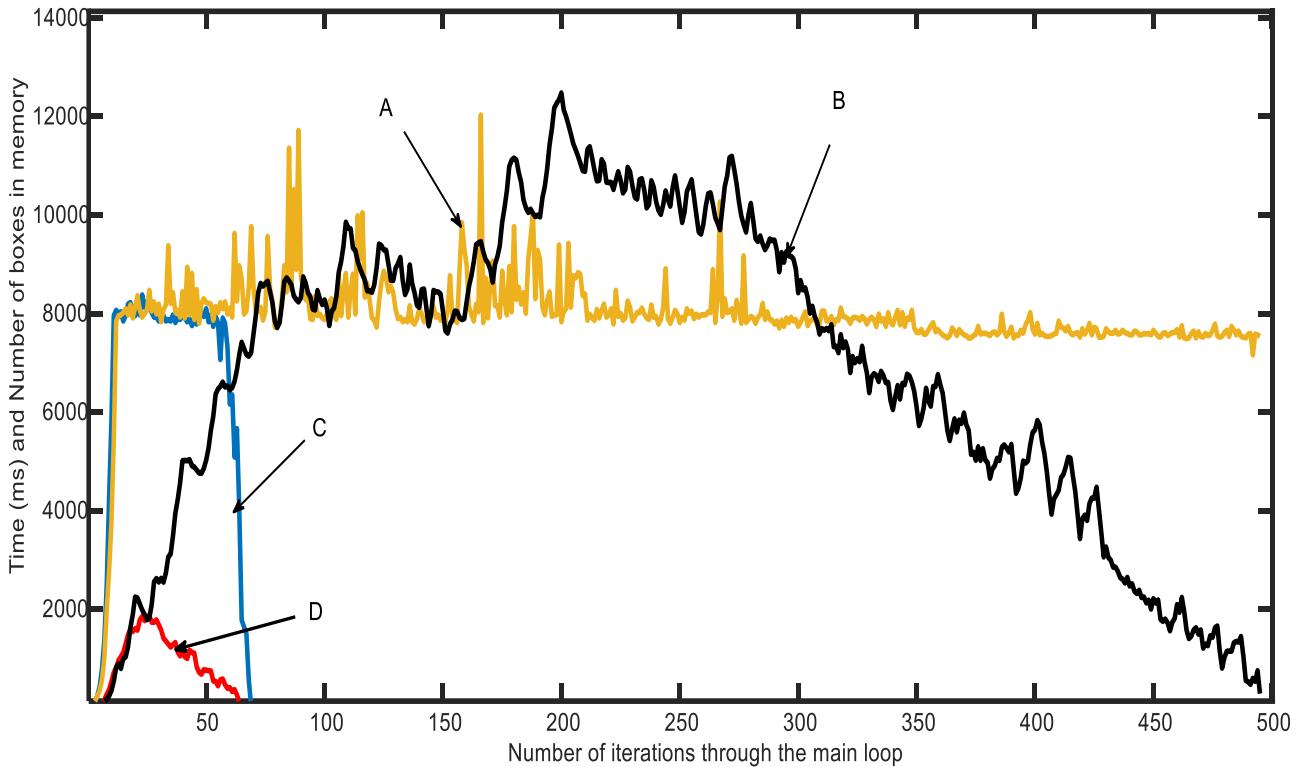


Figure 5.3 Experimental testing of the splitting rule: the execution time and memory space used to extract a handwriting gesture when use: the gradient multiplied by the box dimension (blue (c) and red (D) line), the normalised box dimension (brown (A) and black (B))

presented in Figure 5.2A: The same results are presented in Figure 5. 2B for a real velocity profile. Both velocities are strictly bound by their respective bounding box since no sample comes out of the envelopes. **Selection rule:** select a box \mathbf{P}_i^I such that $\underline{F}(\mathbf{P}_i^I) = \{ \min(F(\mathbf{P}_i^I)), \text{with } \mathbf{P}_i^I \in L_1 \}, i = 1 \dots N \}$, where L_1 is the working list, N the list size, and \mathbf{P}_i^I an interval vector or box stored in the list during the search. In other words, we select in the working list of candidate boxes the one that has the smallest lower bound.

Splitting rule: knowing that the gradient is a crude measure of how much f changes as a variable P_i changes over the box, we bisect the selected box \mathbf{P}^I in the direction i such that $i = \max_{i=1 \dots n} w(P_i)w(F'(P_i))$ [59]. Normalisation is applied to each box since the variables in a box have different units and scale. We normalise boxes by dividing each parameter with its respective absolute middle value. Figure 5. 3 shows the results of two different splitting rules tested with **IAB&BPE**. The experiment compares splitting a box along its largest width (normalized) and the chosen rule ($\max_{i=1 \dots n} w(P_i)w(F'(P_i))$). The results show that, the chosen rule requires less memory (enough to store 2000 boxes) and a processing time of approximately 5 minutes while dividing a box according to the largest normalized width requires a memory space sufficient to store nearly 12000 candidates, as well as an execution time of more than one hour.

Bounding rules: The bounds on f (5.4) are constructed to provide not only the inclusion of the function, but also a close estimate of the real value. In our algorithm, we used the natural interval extension from the fundamental theorem of interval analysis [58] [59] to compute the bounds of f . As shown in (5.6) and provided by the fundamental theorem, the result in the right-hand side of (5.6) is an interval $[\underline{F}(\mathbf{P}^I), \bar{F}(\mathbf{P}^I)]$ that bounds f . To compute that result, we break down the last term $\Delta\Lambda(t; \mathbf{P}^I)$ in the integral of (6) into a unique finite sequence of interval arithmetic operations [57] listed below as steps of Function_ Λ .

Function_ Λ : $([\underline{\mu}_1, \bar{\mu}_1], [\underline{\sigma}_1, \bar{\sigma}_1], [\underline{t}_0, \bar{t}_0], t)$

Set the constant variable const= $\sqrt{2\pi}$;

Compute equation (2) using the following list of expressions (‘;’ are used as separators for brevity).

$$T_1 = [\underline{\sigma}_1, \bar{\sigma}_1]^2 ; \quad T_2 = t - [\underline{t}_0, \bar{t}_0] ;$$

$$T_3 = \ln(T_2) ; T_4 = T_3 - [\underline{\mu}_1, \bar{\mu}_1];$$

$$T_5 = T_4^2 ; T_6 = \frac{T_5}{T_1} ; T_7 = 0,5 \times T_6 ;$$

$$T_8 = -T_7 ; T_9 = \exp(T_8) ;$$

$$T_{10} = \text{const} \times [\underline{\sigma}_1, \bar{\sigma}_1]; T_{11} = T_2 \times T_{10} ;$$

$$T_{12} = \frac{T_9}{T_{11}} ;$$

return T_{12} ;

In this pseudo-code listing, each arithmetic function (\pm, \times, \div , etc.) and unary elementary functions of the type ($\exp, \ln...$) are expressed using intervals arithmetic with only one unique variable in their respective expressions. This ensures that the desirable properties (inclusion isotonicity, etc.) of the Fundamental Theorem are satisfied. Having computed the lambda (Λ) term in (5.6), the bounds are given by the function Compute_objective_bound listed below.

Compute_objective_bound ($\mathbf{P}_i^I, v_t(t), t, N$)

set the initial variables: $\Delta\Lambda[N], F = []; \Lambda1[N]; \Lambda2[N]$

$$1. \quad \mathbf{P}_i^I = \{ [\underline{d}_1, \bar{d}_1], [\underline{\mu}_1, \bar{\mu}_1], [\underline{\sigma}_1, \bar{\sigma}_1], [\underline{d}_2, \bar{d}_2], [\underline{\mu}_2, \bar{\mu}_2], [\underline{\sigma}_2, \bar{\sigma}_2], [\underline{t}_0, \bar{t}_0] \}.$$

2. for $i=0$ to $N-1$

3. if $(t_i - [\underline{t}_0, \bar{t}_0]) > 0$

4. $\Lambda1[i] = \text{Function_}\Lambda([\underline{\mu}_1, \bar{\mu}_1], [\underline{\sigma}_1, \bar{\sigma}_1], [\underline{t}_0, \bar{t}_0], t_i);$

5. $\Lambda2[i] = \text{Function_}\Lambda([\underline{\mu}_2, \bar{\mu}_2], [\underline{\sigma}_2, \bar{\sigma}_2], [\underline{t}_0, \bar{t}_0], t_i);$

6. $\Delta\Lambda[i] = \Lambda1[i] - \Lambda2[i];$

else

$\Lambda1[i] = 0 ; \Lambda2[i] = 0 ; \Delta\Lambda[i] = 0 ;$

7. compute the integral

8. if $(t_i - [\underline{t}_0, \bar{t}_0]) > 0$

$$((v_t[0] - \Delta\Lambda[0])^2 + (v_t[N-1] - \Delta\Lambda[N-1])^2) \times 0.5$$

For i=1 to N-1

$$9. \quad \text{If } (t_i - [\underline{t}_0, \bar{t}_0]) > 0$$

$$F += (v_t[i] - \Delta\Lambda[i])^2;$$

Else

$$10. \quad F += 0;$$

$$11. \text{ return } \frac{F}{\text{sample frequency}};$$

Compute_objective_bound takes as inputs a box \mathbf{P}^I , the ideal velocity profile $v_t(t)$, the sample time t, and the sample size N. From step 3 to 8, Compute_objective_bound computes the Delta-Lognormal function by calling Function_Λ for each lognormal sample and then computes the integral to return the bounds on F. The integral of the nonlinear least square that constitutes our objective function (see 5.4) is calculated numerically (see steps 8-11) as proposed in [73]. The condition tested in steps 4, 9 and 10 guarantees the inclusion of the objective function (when computing the integral) to lie in the whole domain where it is defined, regardless of the box to be evaluated.

Figure 5.4 presents an experimental verification of the bounding rule for the $\Delta\Lambda$ function. The bounding box represented by the thick blue line is computed using the **NIE** for the space defined in Table 5.2. Inside this space, more than 300 $\Delta\Lambda$ curves represented by fine red lines are presented. The parameters of these curves are generated by dividing the bounding box into sub-boxes such

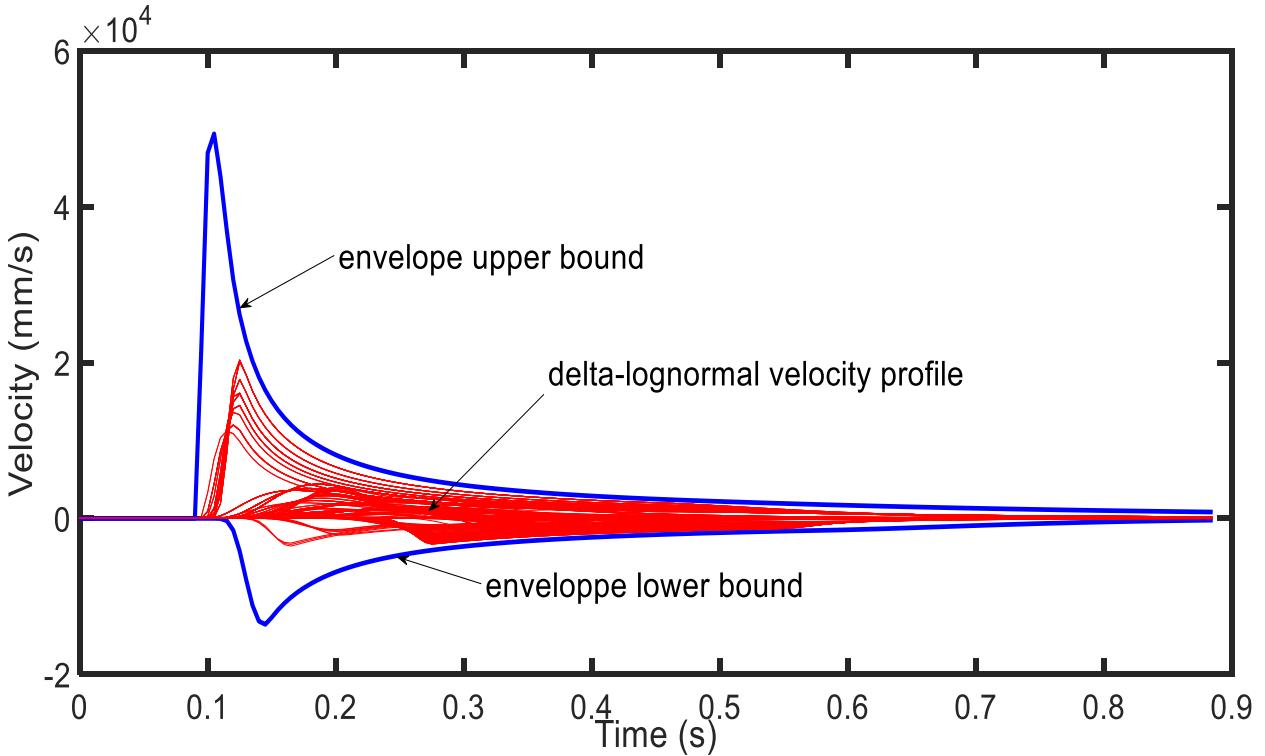


Figure 5.4 Experimental validation of the Delta-Lognormal envelope

that $P_k \subset P$ for $k = 1 \dots n$. As can be seen, the envelope fully bounds all observed $\Delta\Lambda$ in the given space, since no curve extends beyond these limits.

Discarding rule: the following tests are used:

Midpoint test: two tests are applied:

1. Discard any box \mathbf{P}^I which satisfies the condition $\underline{F}(\mathbf{P}^I) > \bar{f}^*$, where \bar{f}^* is the best known upper bound of f^* and $\bar{f}^* = f(m(\mathbf{P}^I))$.
2. Discard a box in which $v_{ti}(t)$ is not included in $\Delta\Lambda(t_i; \mathbf{P}^I)$ for each t_i sample.

As stated, the second test is very restrictive for our optimization problem, it suffices that a single sample of low amplitude of $v_{ti}(t)$ is outside the envelope $\Delta\Lambda(t_i; \mathbf{P}^I)$ to reject the box. In practice

and for some applications, we ensured that at least $N > 1$ $v_{ti}(t)$ samples have their amplitude greater than the $\Delta\Lambda(t_i; \mathbf{P}^I)$ for a reasonable amount fixed by the user.

Cut-off test: discard any box \mathbf{P}_i^I stored in the working list that satisfies $\underline{F}(\mathbf{P}_i^I) > \tilde{f}$.

Monotonicity test: reject any box \mathbf{P}_i^I in which the condition $0 \notin \nabla F(\mathbf{P}_i^I)$ is fulfilled.

Note that all the discarding tests are independent and can be used in parallel for the same box. Let us take a simple example to have a better look on how they are applied.

Considering example 1, the derivative of f with respect to x and y is given respectively by: $f'_x(x, y) = -2 * (1 - x) - 400x(y - x^2)$ and $f'_y(x, y) = 200(y - x^2)$ and the minimum is $(x, y) = (1, 1)$ for $f(1, 1) = 0$ and $f'_x(1, 1), f'_y(1, 1) = 0$; Lets the bound be $X = [2; 3]$ and $Y = [-1; 0]$. For the monotonicity, test applying the NIE to f'_x and f'_y gives: $f'_x(X, Y) = [3204; 12004]$ and $f'_y(X, Y) = [-2000; -800]$ in which case zero does not belong to the results. For the midpoint test, let $f(0.5, 1) = 56.5$ be the best upper bound known then the box $\{X', Y'\}$ cannot be deleted because $F(X', Y') = [0, 904]$ has its lower bound lower than 56.5. Figure 5. 5 shows the efficiency of the different discarding tests for a handwriting pointing gesture extraction. In this experiment, the candidate boxes are evaluated in blocks of 500 per iterations through the main loop. Thus, a potential of 1000 boxes (because we divide the boxes in two) are likely to be future candidates if they are not rejected. As can be seen in Figure 5. 5, the discarding test rejects at least half of them at each iteration, which is very efficient.

Storing rule: we store a sub-box \mathbf{P}_i^I of \mathbf{P}^I in the working list in increasing order with respect to $\underline{F}(\mathbf{P}_i^I)$ as first criterion and in decreasing order with respect to the age of the box \mathbf{P}_i^I already in the list as second criterion. Note in this step that we store a box with its linked information according to (5.8). It avoids calculating them again in the splitting rule.

Termination rule: when a box satisfies the condition $w(P) = \max_{i=1 \dots n} w(\mathbf{P}_i^I) < \varepsilon_x$, $w(F(\mathbf{P}^I)) < \varepsilon_f$ for and $f(m(\mathbf{P}^I)) < \varepsilon_{fmid}$ we store it in the final list L_2 . Parameters ε_x and ε_f are set according to ε_{fmid} which is the absolute mean square error (5.4) or the SNR (5.5).

The tolerance on f ($\varepsilon_{fmid} = \varepsilon_f$) is fixed according to (5.10):

$$\varepsilon_{fmid} = \int (v_{t, \text{full}}(t) - v_{t, \text{round}}(t))^2 dt \quad (5.10)$$

where $(v_{t_full}(t) = v_t(t))$ is the full resolution velocity emitted and $(v_{t_round}(t))$ is the reduced (each sample round off to X decimals) resolution velocity. X is set by the users. The tolerance ε_x

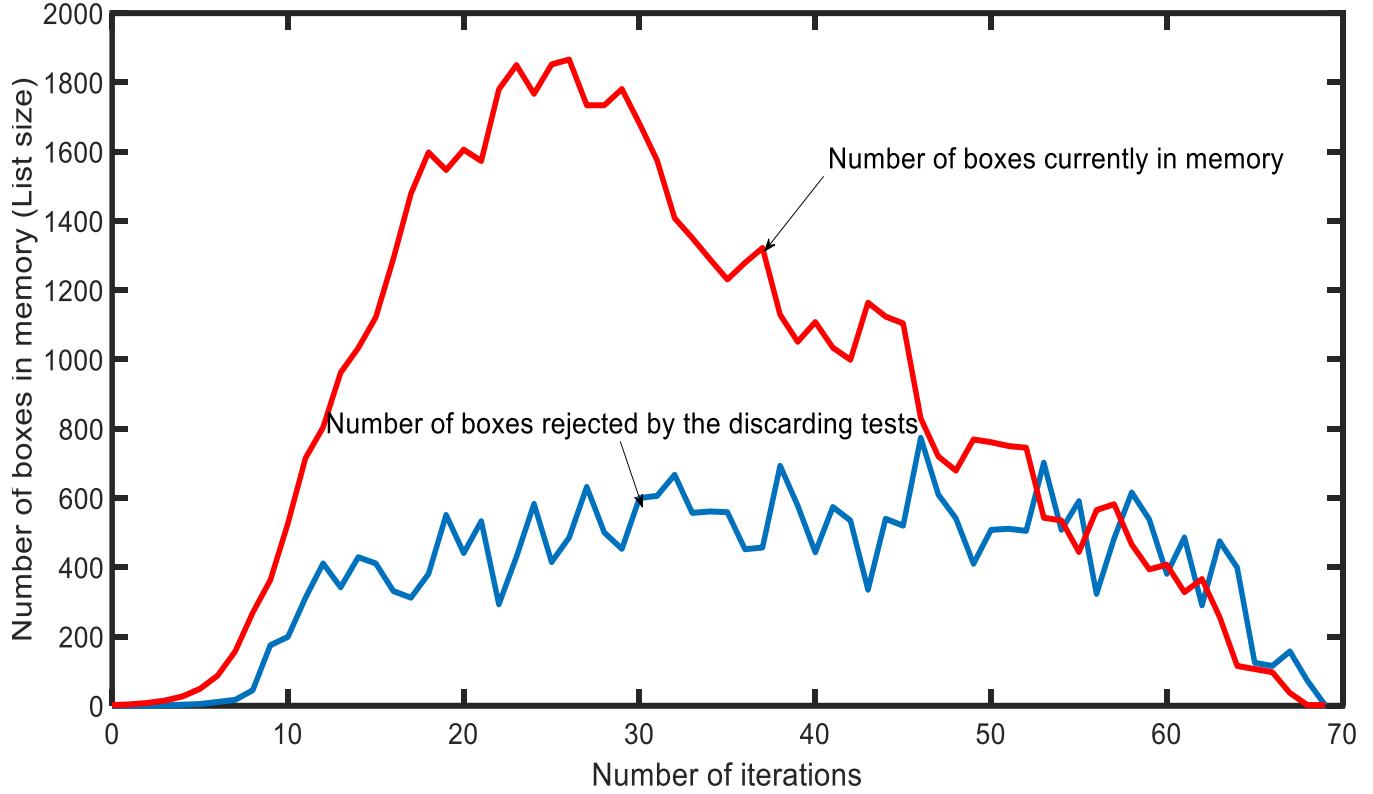


Figure 5.5 Efficiency of the discarding test: the number of boxes rejected by the discarding tests (blue), the number of boxes in the memory (red)

on the boxes is experimentally established using ideal speed profiles. We use (5.11) to find the value of the rounding coefficient (ex) that will correspond to ε_{fmid} .

$$\begin{cases} \varepsilon_x = 10^{-ex} \text{ for } ex = 1 \dots N \\ \varepsilon_{fmid} = \int (v_{t_full}(t) - \Delta \Lambda(t, p_{round}))^2 dt \end{cases} \quad (5.11)$$

where p_{round} is the parameter value rounded to (ex) decimal. To get ε_x we iterate using (5.11) by gradually increasing the rounding coefficient (ex) to get ε_{fmid} . After several tests with ideal speed profiles, we found that ε_{fmid} is recovered up to an error of 10^{-4} when $ex = 4$ which correspond to $\varepsilon_x = 10^{-4}$.

The criteria $W(F(\mathbf{P}^I)) < \varepsilon_f$ comes from the Lemma, at the optimum $W(F(\mathbf{P}_{optimal}^I)) = 0$ and the width of $W(\mathbf{P}_{optimal}^I)$ is also zero. Thus, close to the optimum solution or in a local minimum,

the criteria prevent unnecessary subdivision due to the linearity between the width of a box and the width of the image of the function through that box.

Example 3: Let us say that we want to minimize the non-convex function $f(x, y) = (1 - x)^2 + 100(y - x^2)^2$ that has a minimum at $x = 1, y = 1$ with $f(1,1) = 0$ when we are close to the solution we could arrive to the box $X = [0.95, 1], Y = [1, 1]$. In this case, the NIE of f is $F(X, Y) = (1 - X)^2 + 100(Y - X^2)^2 = [0 ; 0.953]$ and its width is $W(F) = 0.953$. The width of the box is given by $W(X, Y) = 0.05$. We see that $W(F)$ is proportional to the box width $W(X, Y)$ and fixing it can prevent the algorithm for unnecessary subdivision of the box and still preserve the optimum.

5.6 Testing the algorithm with ideal velocity profiles

To test the extractor, we developed a database comprising more than 9000 synthetic $\Delta\Lambda$ curves grouped into nine categories. Each category is defined based on the number of peaks within the $\Delta\Lambda$ velocity profile and the relative positions of the agonist component with respect to the antagonist [22]. To create this database, we start by acquiring real human movements with a *Wacom intuos* digitizer. The *Wacom* tablet collects the Cartesian coordinates of the position of a pen tip movement at a frequency of 200Hz. The recorded coordinates are derived with respect to a 5ms time base to compute the original velocity profile $\overrightarrow{v_t(t)}$. This $v_t(t)$ is then used as input to the **IIX** optimizer to get its $\Delta\Lambda$ parameters.

From the set of parameters produced by **IIX**, we select those that satisfied a SNR of 25 dB with D1 and D2 less than the digitizer size (31cm * 23 cm). With the previously obtained parameters, we construct each ideal velocity test profile ($v_{ideal}(t)$) by evaluating (5.1) at a sample frequency of 200Hz to simulate the data collected with a digitizer. The movement duration is typically between 100 and 500ms. The resulting velocity $v_{ideal}(t)$ is then fed into the **IAB&BPE** to try to retrieve the same values in the p 7-tuple. We used this process mainly to ensure that the generated data are human-like models and that they properly test the algorithm with speed profiles that have known parameter values.

5.6.1 Tests conditions

We run the **IAB&BPE** using a 3.60 GHz Intel core (i7) with 32 GB of RAM. The implementation is done in C++ using rounded interval arithmetic. We implemented our custom interval arithmetic library based on [58] and [74]. In general, the algorithm is tested on its ability to retrieve the global solutions of the nine classes of ideal $\Delta\Delta$ velocity profiles and to extract lognormal parameters from real handwriting movement within a fixed accuracy. These objectives are achieved following three experiments:

1. Test the ability of **IAB&BPE** to find the global bounding box according to a set accuracy.
2. Evaluate its capacity to find the global solution with less processing time than in the base line [50].
3. Test its ability to recover the global solution with low memory usage when compared to the base line [50].

5.6.2 Evaluation of the extractor under ideal conditions

The first experiment deals with ideal $\Delta\Delta$ speed profiles. In this experiment, since we already know the ground truth, a box is considered as a solution and stored in the list of results if the maximum width of the chosen box is less than 10^{-4}mm , or the quadratic error is less than $10^{-6} (\text{mm/s})^2$. This corresponds to a **SNR** of at least 100dB. Table 5.2 presents the search region restricted for simplicity to a finite area, where we know that the solution can be found. Our goal in this experiment is to test **IAB&BPE** in its ability to recover the true parameter values inside a bounding box containing at least the user-supplied target.

The average values of the absolute error for each estimated parameter are shown in Table 5.3. As can be seen for the 9000 tested curves, the estimation errors of t_0 and D_1 are on average 10^{-9}s and 10^{-6}mm respectively. We can also observe that the new algorithm always finds not only the base line target within an accuracy of $\varepsilon = 10^{-6}$, but also a confidence interval (Min and Max in Table 5.3) including the target value. This interval vector reveals that, if we decide to generate a better solution (with a smaller error up to the limit of numerical precision), we just need to continue the search starting from that bounding box.

Table 5.2 Initial box

Parameters	t_0	D_1	μ_1	σ_1	D_2	μ_2	σ_2
Minimum	0.00001	50	-5	0.0001	10	-5	0.001
Maximum	0.9	500	-0.05	1.0	300	0.005	0.9

Table 5.3 Statistics of the absolute error on the parameters estimated by the IAB&BPE on ideal data

Parameters	Mean	Min	Max
t_0	1.18E-09	0.0	2.19E-05
D_1	2.90E-06	0.0	5.13E-04
μ_1	1.08E-08	0.0	1.43E-05
σ_1	4.95E-09	0.0	5.48E-05
D_2	2.01E-06	0.0	3.26E-04
μ_2	1.00E-08	0.0	1.32E-05
σ_2	8.58E-09	0.0	1.40E-05

Table 5.4 Extraction (in %) of the exact value of the parameters for a snr>100db as ending criterion

Classes	Xzero	IIX	[50]	IAB&BPE
Ca0	95.2	99.7	100	100
Ca1	96.6	99.5	100	100
Ca2	100	100	100	100
Cai	100	100	100	100
Cb1	97.9	98.5	100	100
Cbi	98.1	98.1	100	100
Cs2	92	99.6	100	100
Cb2	--	--	--	100
Cs0	--	--	--	100

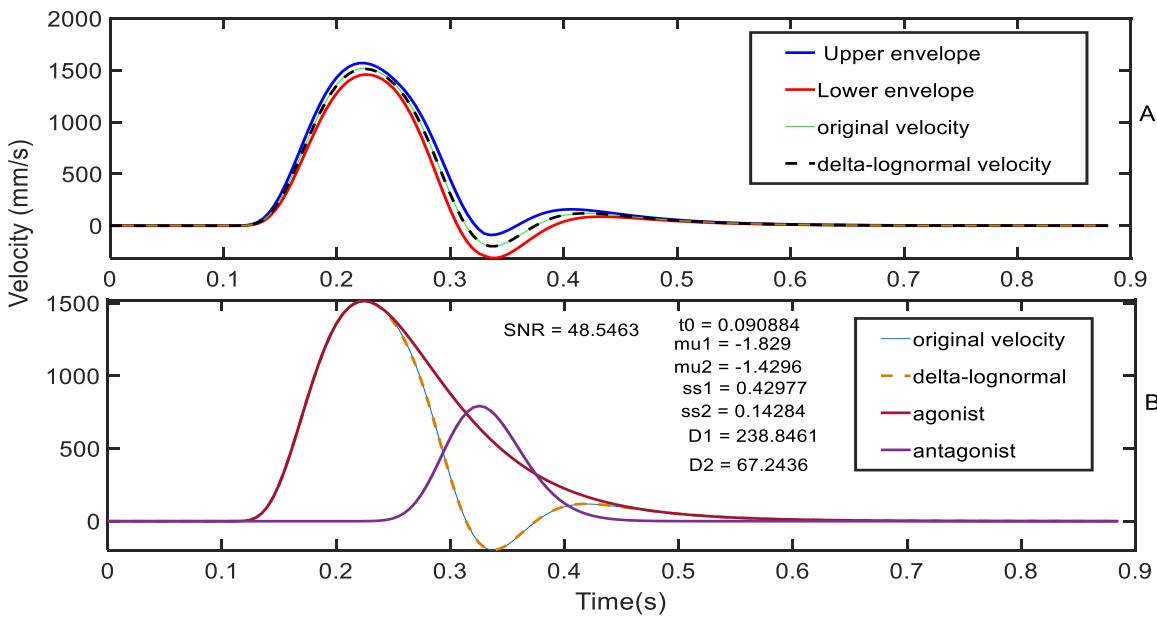


Figure 5.6 Ideal velocity profile parameters extracted by IAB&BPE. A. Upper and lower envelope surrounding the ideal and extracted speed profile. B. original and extracted velocity profile, with the last one being decomposed into its agonist and antagonist neuromuscular

Table 5.4 presents the results of the seven parameters extracted by **IAB&BPE**, **IIX**, and [50] for each class of velocity profile, we compared the extracted parameter with the respective true value. As can be seen in that table, except for the class C_{ai} and C_{a2} , our algorithm always performs better than **Xzero**, **IIX**, and produces the same results as in [50]. Figure 5.6 shows an example of an

extracted velocity profile of the Ca2 class with its resulting bounding box. We can see in the upper part (part A) that the resulting bounding box (upper and lower envelope) clearly frames the original and extracted speed profiles. Note that the bounding box shown in Figure 5. 6 is the result of the refinement of the initial box shown in Figure 5. 4 by **IAB&BPE**. The second part of the graph (part B) is the chosen velocity profile with its corresponding agonist and antagonist components inside that bounding box satisfying at least 51 dB SNR.

5.6.3 Evaluation of the Extractor under Noisy Conditions

The second experiment is to validate the ability of **IAB&BPE** to extract global solutions from ideal velocity profiles to which noise is added. The reason for doing so is that, in real-life applications, digitized data are sometimes distorted from different sources. On the other hand, using noisy ideal data whose parameters and level of noise are known permits to determine the effects of the level of noise on the accuracy of the extracted parameters for real curves. We create noisy data by adding white Gaussian noise to the 9000 ideal curves of the database in such a way that the resulting curves look more like those encountered in real situations. We set each ideal $\Delta\Delta$ velocity profile of the database with various controlled levels of noise (SNR=20, 30, 40, 50 dB). In this experiment, a

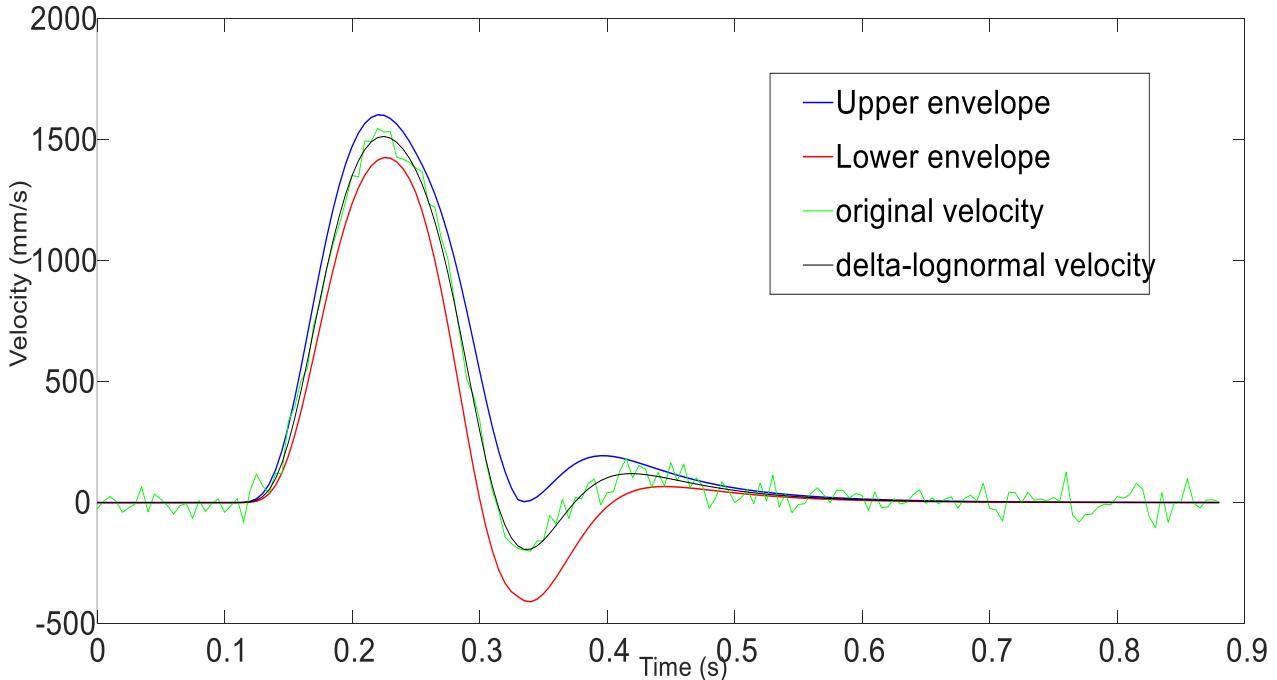


Figure 5.7 Ideal noisy velocity profile parameters extracted by IAB&B. Superior and inferior envelopes surrounding the ideal and extracted speed profiles are plotted

box is considered as a solution if its **SNR** is at least 10 times smaller than the **SNR** of its noisy curve, i.e., for an ideal speed profile for which we imposed a 20 dB **SNR** noise, the stopping criteria is fixed to 10 dB **SNR**. In doing so, we provide the algorithm with the necessary margin to avoid overfitting.

We compared the accuracy performance between the different extractors for all the different noisy cases. We found that the proposed algorithm performed better than **Xzero**, **IIX** and the base line [50]. For all the 9000 noisy curves tested, **IAB&BPE** converges for all, whereas **IIX** and [50] converge on average in 98% of cases and **Xzero** in 95%. Figure 5. 7 shows an example of an ideal noisy velocity profile extracted by **IAB&BPE** for the extreme case of 20 dB **SNR** noise. The envelopes (SUP and INF) that correspond to the minimum and maximum velocity profiles produced by the bounding box returned by **IAB&BPE**, perfectly frame the noisy data for an accuracy chosen at 10 dB **SNR**. In this example, we adjusted the second midpoint test (see Discarding rule) of **IAB&BPE** to only keep boxes that enclose points of the noisy velocity profile where there is a significant change in amplitude between two successive samples. This constraint enabled the **IAB&BPE** to extract the parameters at higher **SNR** levels, without losing the operational optimal solution, despite that some parts of the noisy velocity profile are outside the bounding box found. This important result shows that, the algorithm manages to produce a confidence interval guaranteeing optimality despite the fluctuations caused by the noise in the curves.

5.7 Performance Evaluation with Real Human Movements

After training the **IAB&BPE** with ideal and ideal noisy handwriting gestures, an experiment has been conducted using data collected from human movements. In this experiment, 500 online handwritten movements were selected randomly from the Scribens Laboratory (Polytechnique Montréal) Database. The data were acquired with a Wacom Intuos2 digitizer, sampled at 200 Hz. The movements were produced by 25 humans aged between 18 and 45, except for the parkinsonian subjects who were older. The 25 participants for data collection were distributed as follows: 18 professional college football athletes, 2 persons with concussions, 2 persons with attention deficit disorders (**ADD**), 3 persons with Parkinson's disease. Each participant was asked to draw 30 simple unidirectional lines of at least 130 mm with his left and right hands. We took randomly 20 handwriting pointing gestures from each participant for our test.

The search space for each real movement is built using the expanded $\Delta\Delta$ parameters obtained by the **IIX** extractor. Indeed, each solution returned by **IIX** is considered as the midpoint of a bounding box in which each parameter value on either side of each interval is deduced as presented in the initial rule. Very conservative values were chosen to allow the parameters to take aberrant values in order to validate the ability of the algorithm to reject these solutions. For this test, the only stopping criterion for a box to be considered as a solution is the **SNR** that we fix to be more than 25 dB because it is the regular threshold used in many experiments, according to **IIX** and [50]. Moreover, we preferred this measure to plain error for these movements because it allows comparison of the fitting accuracy of the movement with different signal energies.

For a reconstruction criterion of at least 25 dB **SNR**, the proposed algorithm converges for all cases studied, whereas **IIX** and [50] converge at most 95% and 60% of the time respectively. We also calculated the average error of all $\Delta\Delta$ envelopes enclosing the entire human velocity profile tested for each algorithm and again for this metric, **IAB&BPE** performed better than all the others.

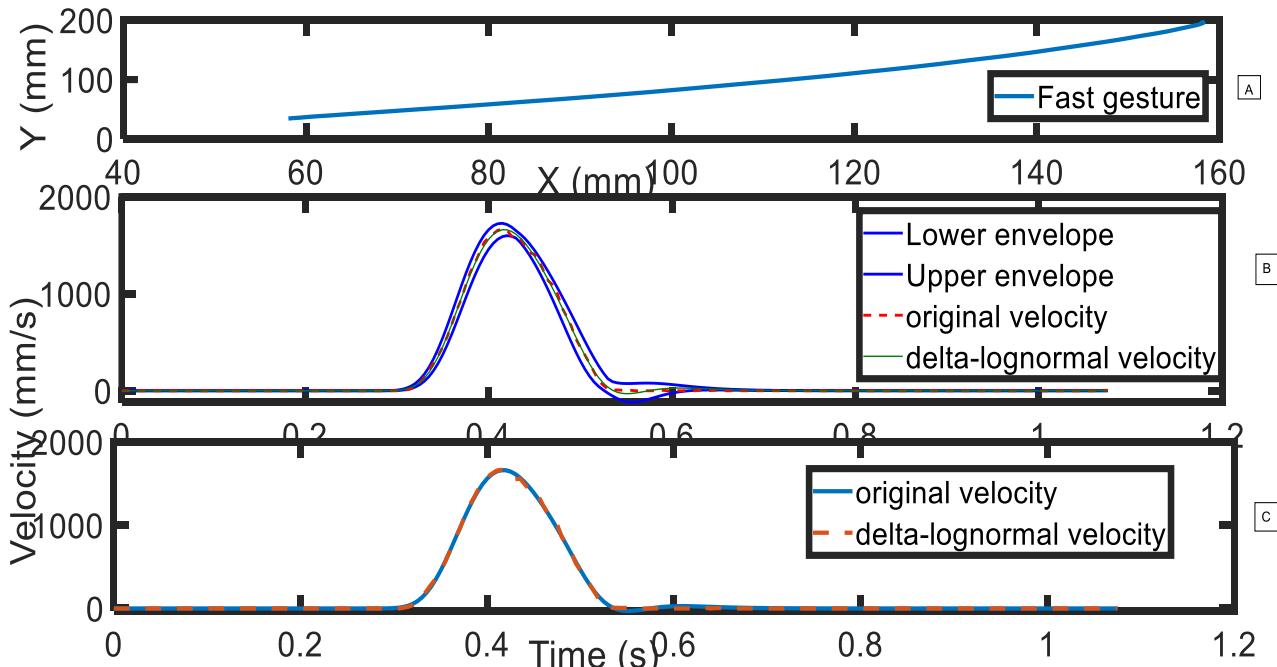


Figure 5.8 Example of human Handwriting strokes extracted by **IAB&BPE**: A. the (x, y) position of the pen tip movement of a writer, B. the real and reconstructed velocity profile with a 31dB **SNR** enclosing their envelopes, C. the reconstructed velocity profile

Figure 5. 8 shows an example of a human pen tip movement and its corresponding original and reconstructed velocity profiles. Both the original and its chosen reconstructed are found in the bounding box returned by the algorithm (Figure 5. 8 B.). Table 5.5 presents the parameters extracted by the **IAB&BPE** for this example, the middle values are the final parameters of the velocity of the handwriting movement of Figure 5.8A. The maximum and minimum values correspond to the envelope (Figure 5. 8.B) that bounds the original and $\Delta\Delta$ velocity profile (Figure 5.8C) of the pointing gesture. The absolute error between the maximum and minimum values in Table 5.5 for each quantity shows small changes which result in good confidence intervals for each parameter.

Table 5.5 Parameter value of the speed profile shown in Figure 5.8.C for the gesture shown in Figure 5.8.A the extracted parameters are for a **SNR**=31dB

Parameters	t_0	D_1	μ_1	σ_1	D_2	μ_2	σ_2
Maximum	0.218	220.43	-1.15	0.259	25	-1.12	0.101
middle	0.217	219.08	-1.54	0.255	25.03	-1.15	0.1
Minimum	0.2175	217	-1.56	0.251	24.4	-1.19	0.098

A performance comparison in terms of processing time and number of boxes explored between the proposed algorithm and [50] has been investigated. In [50], the algorithm was executed in parallel, on a 12-CPU node grid. This latter algorithm requires a run time of about 50s to perform 11 iterations of its main loop, while the algorithm proposed in this paper requires only 702ms in a 3.60 GHz Intel core (i7) computer. On the other hand, the number of boxes needed to find a solution of the desired quality could reach one million with [50], while ours requires at most 70 thousand boxes to extract such a solution.

Figure 5.9 shows on the y axis, the execution time in milliseconds as well as the memory space required expressed in number of boxes (a same scale was used for both), as functions of the number of iterations performed through the main loop for one extraction of the handwritten gesture of Figure 5. 8. The processing time curve presents a rapid increase and decrease at the beginning and at the end of the extraction process. These changes describe the small number of candidates boxes to be explored. At this stage (0th to 26th iteration), the algorithm is expanding the initial bounding

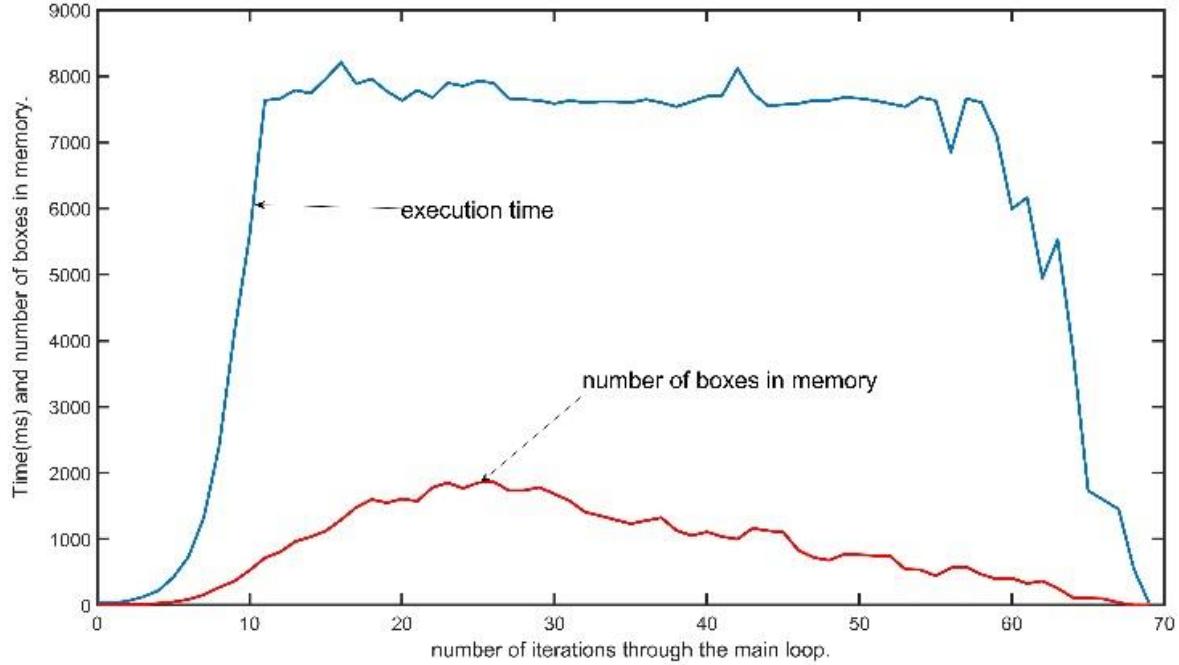


Figure 5.9 Execution time and memory space during the extraction of a real handwriting pointing gesture. Note that a same axis is used both for the time in ms and for the memory space expressed in boxes

box into smaller boxes with few discards. The number of boxes in memory is growing faster until it reached a maximum of 1912 boxes. Between the 10th and 60th iterations, the algorithm is stable (execution time around 8000ms) and works at its full potential. The number of blocks (500) of candidates to be evaluated is the same at each iteration and the discarding tests reject the candidates that do not guarantee a better SNR according to the best currently known. The execution time in this part is almost the same because the memory space has enough (more than 500) storage box for evaluation.

The second example shown in Figure 5. 10 is the speed profile of a Parkinsonian subject (green curve) and its approximated $\Delta\Lambda$ curve (red curve) obtained by the **IAB&BPE**. In this example, as for other handwritten movements from unhealthy (ADD, cerebral concussion, Parkinson's) people, the second midpoint test ($v_{ti}(t) \notin \Delta\Lambda(ti; \mathbf{P}^I)$) is adjusted to consider only the portion of the speed signal that has an amplitude greater than 5% of the maximum amplitude of the original gesture speed profile. This adjustment excludes the portions of the signal that contained small distortions at the beginning ($0.1 < t < 0.4$ ms) and at the end ($1.1 < t < 1.1$ ms) of the gesture speed profile, to be considered by the second midpoint test. it helps the **IAB&BPE** to extract parameters with a higher

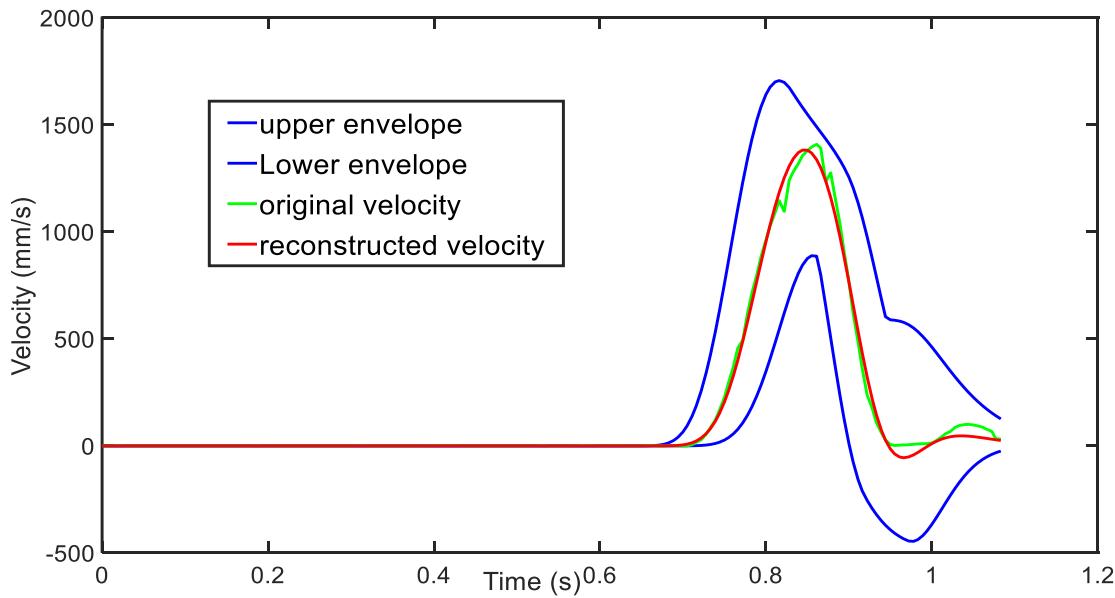


Figure 5.10 Example of parkinsonian handwriting pointing gesture extracted by IAB&BPE for a 26dB **SNR**: the envelope enclosing the original (green) and the $\Delta\Lambda$ (red) velocity profile

SNR without modifying the ability of the algorithm to recover the operational optimal solution since it only frame parts of the signal that have a large amplitude so that the restrictive midpoint test can evaluate the rest of the speed profile.

5.8 Conclusion

This paper proposed a globally optimal $\Delta\Lambda$ parameters extractor based on the Branch and bound procedure. The main concepts proposed are based on original techniques never yet explored in previously published work. The different innovative ideas reported in the present paper are:

- The derivation of a $\Delta\Lambda$ envelope that guarantees to include the signal of the pen tip movement for parameters bounded by box constraints as a way of obtaining a bound on the value of the uncertainty associated with experimental data fitting.
- A technique for splitting an interval vector and calculate its corresponding $\Delta\Lambda$ envelope using interval arithmetic and its fundamental theorem.
- A branch and bound algorithm for the study of complex and multidimensional objective functions that are not easily studied by standard analysis tools. Indeed, since this algorithm produces a finite space of operational optimal solutions, different rules based on the variations

of the objective function have been established to allow the safe pruning of solutions that are inferior to the best current known and to accelerate the convergence of the extractor.

- A set of discarding tests specific to the $\Delta\Lambda$ function, all-independent from each other guides the algorithm in finding the operational optimal solution.

These innovative achievements led to an algorithm able to extract $\Delta\Lambda$ parameters on all types of rapid human pointing gestures. The algorithm automatically provides a confidence interval on each parameter in which the set of possible solutions of a given handwriting pointing gesture velocity profile is found. The confidence interval on each parameter provided by the algorithm permits to construct the $\Delta\Lambda$ envelope that strictly bounds the human speed profile that has been extracted.

Besides recovering solutions in a confidence interval as primary target, the effectiveness of this method as well as performance comparisons in terms of accuracy and processing time with different algorithms on real and synthetic human handwriting pointing gesture has been reported. The results are quite remarkable. The number of evaluations needed was reduced by a 15-fold factor with respect to [50] while obtaining results of a same target quality. Although the proposed algorithm is promising in its ability to recover a global solution in a limited processing time much smaller than any previously reported method, more work remains to be done. For example, to allow a greater use of this tool, methods of acceleration exploiting the natural parallelism as well as the independence of the rules of the **IAB&BPE** will be needed. Tests in this direction will probably leverage graphics processors. The proposed algorithms can be used also with any other mathematically differentiable function.

Acknowledgments

This work was supported in part by NSERC-Canada Grant RGPIN-2015-06409 to R. Plamondon, by RGPIN-2018-06574 to Y. Savaria and by Polytechnique Montréal. We invite anyone interested in using the proposed algorithm to contact the authors for a shared under collaborative non-commercial agreement with Polytechnique Montréal.



S.P.Boyogueno Bidias received the B.Ing and is currently pursuing his Ph.D degree at Polytechnique Montreal in electrical engineering. He was working as a software engineer in the avionics and video industry before joining the Scribens Lab where he is currently working on algorithm design and acceleration. His research interests are diverse including artificial intelligence, theoretical modelling of human movements, machine learning, signal processing, optimization problems, online handwriting analysis and synthesis, biometrics.



J.P. David received the Ph.D. degree from Université Catholique de Louvain, Louvain-la-Neuve, Belgium, in 2002. He has been an Assistant Professor with Université de Montréal, QC, Canada, for three years and moved to Polytechnique Montréal, QC, Canada, in 2006. He is associate professor since 2013. His research interests include digital system design, reconfigurable computing, high-level synthesis, neural networks and their applications.



Réjean Plamondon is Professor at Polytechnique Montréal, P.Q., Canada. Over the years, he has been involved in many pattern recognitions projects, particularly in the field of on-line and off-line handwriting analysis and processing. His main contribution has been the development of the Kinematic Theory of rapid human movements. The theory has been found successful in describing the basic kinematic properties of velocity profiles as observed in fingers, wrist, hand, arm, trunk, head and eye movements as well as in speech. Professor Plamondon has studied and analyzed these bio-signals extensively in order to develop creative and powerful methods and systems in various domains of engineering. Fellow of NIAS (1989), of IAPR (1994) and of IEEE (2000), Professor Plamondon has been involved in the planning and organization of numerous international conferences and workshops and is working with scientists from many countries. He received the IAPR/ICDAR 2013 outstanding achievement award for “theoretical contributions to the understanding of human movement and its applications to signature verification, handwriting recognition, instruction, and health assessment, and for promoting on-line document processing in numerous multidisciplinary fields.



Yvon Savaria FIEEE (S' 77, M' 86, SM' 97, F'08) received the B.Eng. and M.Sc.A in electrical engineering from Polytechnique Montreal in 1980 and 1982 respectively. He also received the Ph.D. in electrical engineering in 1985 from McGill University. Since 1985, he has been with Polytechnique Montreal, where he is currently professor in the department of electrical engineering. He is also affiliated with Hangzhou Innovation institute of Beihang University

He has carried work in several areas related to microelectronic circuits and microsystems. He holds 16 patents, has published 165 journal papers and 450 conference papers, and he was the thesis advisor of 160 graduate students who completed their studies.

He is general Chair of NEWCAS'2020 and was program co-chairman of NEWCAS'2018. He is a member of the Regroupement Stratégique en Microélectronique du Québec (RESMIQ), of the Ordre des Ingénieurs du Québec (OIQ), and of CMC Microsystems Advisory Committee. He was awarded in 2001 a Tier 1 Canada Research Chair (www.chairs.gc.ca) on design and architectures of advanced microelectronic systems that he held until June 2015. He also received in 2006 a Synergy Award of the Natural Sciences and Engineering Research Council of Canada

CHAPITRE 6 DÉVELOPPEMENT INÉDITS ET COMPLÉMENTAIRES À L'ALGORITHME DE SÉPARATION ET ÉVALUATION

Ce chapitre inédit traite des développements non publiés et complémentaires aux contributions techniques discutés aux chapitres précédents. Deux grands sujets indépendants y sont présentés:

1. L'élaboration des mouvements rapides synthétiques simples.
2. Les Résultats détaillés des délais d'extraction des paramètres Delta-Lognormaux par l'algorithme **S&E**.

6.1 Élaboration des mouvements synthétiques simples

Dans cette section, nous discutons de la contribution de la Théorie Cinématique du Mouvement Humain Rapide à la compréhension et à la synthèse des fluctuations observées dans la production d'un trait d'écriture simple. Cette discussion permettra d'élaborer un ensemble de contraintes mathématiques sur la fonction Delta-Lognormale qui serviront à construire de manières automatiques des formes de profils de vitesse synthétiques, semblables aux mouvements de pointages manuscrits. Les profils synthétiques construits permettront de tester la capacité de l'algorithme **S&E** à retrouver avec précision les paramètres lognormaux qui les constituent. La

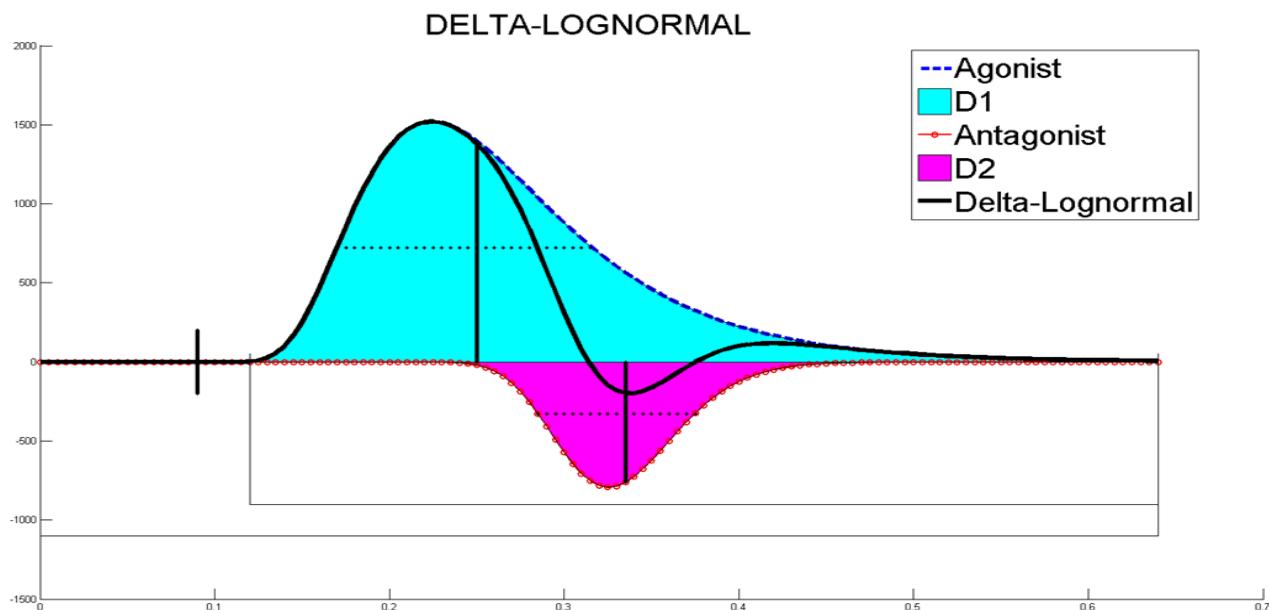


Figure 6.1 Lognormale avec lobe inférieur à droite du lobe principal

réalisation des mouvements synthétiques à partir des changements remarqués au niveau des traits et leurs profils de vitesses manuscrits associés (Figure 6.1) se répartit sur deux tâches principales.

- L'identification et la classification des variabilités observées dans le temps sur des traits et leur profil de vitesse Delta-Lognormal associé.
- L'établissement des contraintes mathématiques sur la fonction Delta-Lognormale afin de générer automatiquement les profils de vitesses synthétiques.

6.1.1 Variabilité dans le temps des mouvements manuscrits simples et classifications des profils de vitesses.

La réalisation des mouvements synthétiques simples représentatifs des traits manuscrits repose sur les observations des caractéristiques des trajectoires engendrées par le mouvement de pointage du stylo sur une tablette numérisée. En s'appuyant sur les travaux de [75] [76]., [77], [78], et nos propres observations, nous avons regroupés plusieurs caractéristiques observables dans les traits et leur profil de vitesse associés après l'émission d'un mouvement manuscrit simple. Ces derniers sont les suivants :

- Le profil de vitesse d'un mouvement pourrait posséder jusqu'à trois lobes.
- Le profil d'accélération pourrait avoir jusqu'à quatre lobes.
- Les trajectoires ne sont pas complètement rectilignes, une légère courbure est observée à mi-parcours au début et à la fin du trait.
- On dénombre au moins deux renversements dans la direction du mouvement.

La présence de lobes à faibles amplitudes par rapport au lobe principal des profils de vitesse (voir Figure 6.1), induit des passages par zéro de la vitesse de pointage. Ces lobes qui sont causés par l'interaction agoniste antagoniste dans l'accomplissement d'une tâche de pointage [13] servent de critères dans l'élaboration des différentes classes des profils de vitesses idéaux à concevoir. Dans ce chapitre, deux nouvelles classes sont identifiées et ajoutées à la collection des sept classes déjà identifiées dans Djouia et Plamondon 2008 dans le cadre des études sur les mouvements à partir du modèle Delta-Lognormal. L'encodage C_{kl} adopté par les auteurs est repris ici pour la clarté. Dans le sigle C_{kl} , la lettre C est l'identificateur de la classe (C pour classe), l'indice k représente la position significative de la composante antagoniste durant le mouvement rapide, et le l désigne le

nombre de passages par zéro du profil de vitesse. À partir de cette nomenclature, neuf classes de profils de vitesse idéaux susceptibles de représenter des profils de vitesses de mouvements rapides de pointage sont établis. Le tableau 6.1 résume la nomenclature associée à l'ensemble des classes des profils de vitesses idéaux observables.

Table 6.1 Classes de profils de vitesses lognormaux idéaux

	Activités significatives de la composante antagoniste au début du mouvement	Activités significatives des composantes antagoniste et agoniste simultanément	Activités significatives de la composante antagoniste à la fin du mouvement
Pas de passage par zéro		C_{s0}	C_{f0}
Passage par zéro imaginaire	C_{di}		C_{fi}
Un passage par zéro réel	C_{d1}		C_f
Deux passages par zéro réels	C_{d2}	C_{s2}	C_{f2}

6.1.2 Construction des différentes classes de profils de vitesses Delta-Lognormales

La conception profils de vitesses synthétiques observées au tableau 6.1 commence par l'obtention des paramètres lognormaux pour chaque classe de profil de vitesse identifié. Cette étape est réalisée par l'acquisition de différents mouvements manuscrits de pointage avec le lognomètre conçu (annexe A). Les mouvements recueillis sont par la suite transmis à l'extracteur **IIX** [53] pour y extraire des paramètres Delta-Lognormaux moyennant un seuil d'ajustement fixé à un rapport signal sur bruit de 25 dB. Les paramètres lognormaux recueillis de **IIX** sont des paramètres

généraux de bases que nous allons conditionner pour générer des profils de vitesses synthétiques semblables aux vitesses réelles. Les valeurs des nouveaux paramètres obtenus par conditionnement de la fonction Delta-Lognormale servent de table de vérité aux profils synthétiques obtenus. Ce procédé est mis en place afin que les profils synthétiques représentent avec une grande fidélité les profils réels (effectués par les humains) et que l'algorithme **S&E** puisse être entraîné avec des mouvements dont les profils de vitesses sont représentatifs de la réalité d'un mouvement rapide de pointage. De plus la précision pourra être facilement quantifiée, car nous avons des valeurs de courbes de vitesses connues. Dans cette section, nous allons montrer comment reproduire les différentes classes de profils de vitesses identifiées au Tableau 6.1 en utilisant des relations analytiques basées sur la fonction Delta-Lognormale et ses paramètres. L'idée de base consiste à établir différentes contraintes sur les paramètres lognormaux en relation avec les morphologies observées. Pour ce faire, nous allons établir les intervalles de dispersion des sept paramètres de la fonction Delta-Lognormale de façon à obtenir les courbes de profils de vitesse correspondantes aux différentes classes du tableau 6.1.

6.1.2.1 Contraintes sur la fonction Delta-Lognormale pour l'obtention des valeurs de t_0 , D_1 et D_2 .

Les contraintes sur les valeurs de D_1 et D_2 s'obtiennent de la manière suivante : Soit D_{\max} la distance maximale possible pouvant être parcourue par un sujet pendant l'émission d'un mouvement de pointage simple sur une tablette numérique de dimension (x^*y) , D_{\max} correspond à la diagonale formée par les dimensions x^*y du dispositif et est donnée par la relation suivante :

$$D_{\max} = \sqrt{x^2 + y^2} \quad (6.1)$$

D'autre part, l'aire en dessous de la courbe Delta-Lognormale (Figure 6-2) correspond à la différence entre les deux amplitudes des composantes neuromusculaires et est donné par la relation :

$$D = D_1 - D_2 \quad (6.2)$$

À partir de la relation 6.2 les intervalles de confiances où résident les valeurs minimales et maximales des amplitudes sont donnés par la relation suivante :

$$\begin{cases} D_{1max} = D_{max} + D_{2max} \\ D_{1min} = D_{min} + D_{2min} \\ D_{2max} = D_{1max} - D_{max} \\ D_{2min} = D_{1min} - D_{min} \end{cases} \quad (6.3)$$

En fixant de manière empirique la valeur de D_1 à 5% de la valeur de D_2 , on est capable de générer des intervalles de confiances pour les valeurs des amplitudes neuromusculaires pour l'obtention des profils de vitesses synthétiques selon la position relative dominante d'une composante vis-à-vis de l'autre. Notons que le choix de 5% dépend de l'expérience neuromusculaire envisagée et la capacité de la tablette numérique à fournir le plus petit mouvement avec effet mesurable. Dans la réalité nous avons établi une plage allant de 5% à 10% afin d'avoir l'ensemble des profils de vitesses synthétiques.

La valeur de t_0 est déterminée à partir de la définition des paramètres lognormaux selon leur interprétation physiologique et des études tirées de la littérature sur les temps de réaction du système nerveux central à un stimulus [10]. La contrainte mathématique du problème qui veut que t_0 soit strictement positif a aussi été prise en compte. Pour nos expériences, nous avons fixé les temps de réaction du système nerveux à un stimulus à $t_{0min} = 50\text{ms}$ pour l'ensemble des expériences. t_{0max} est fixé à la valeur de t pour laquelle le profil de vitesse obtenu avec la tablette présente des valeurs significatives de temps supérieures à 50ms avec une amplitude au moins égale à 10% de l'amplitude maximale après t_{0min} .

En résumé, les équations 6.1 à 6.3 et les hypothèses faites sur D_{max} et D_{min} Permettent d'obtenir les intervalles de confiances des valeurs minimales et maximales des amplitudes des composantes neuromusculaires pour les mouvements synthétiques. La déduction des valeurs de $[D_{2min} \ D_{2max}]$ s'effectue en fixant les valeurs de l'activité antagoniste à 10% de la valeur maximale de D_{1max} . Ces valeurs peuvent être ajustées dépendamment de la classe du profil de vitesse à émettre.

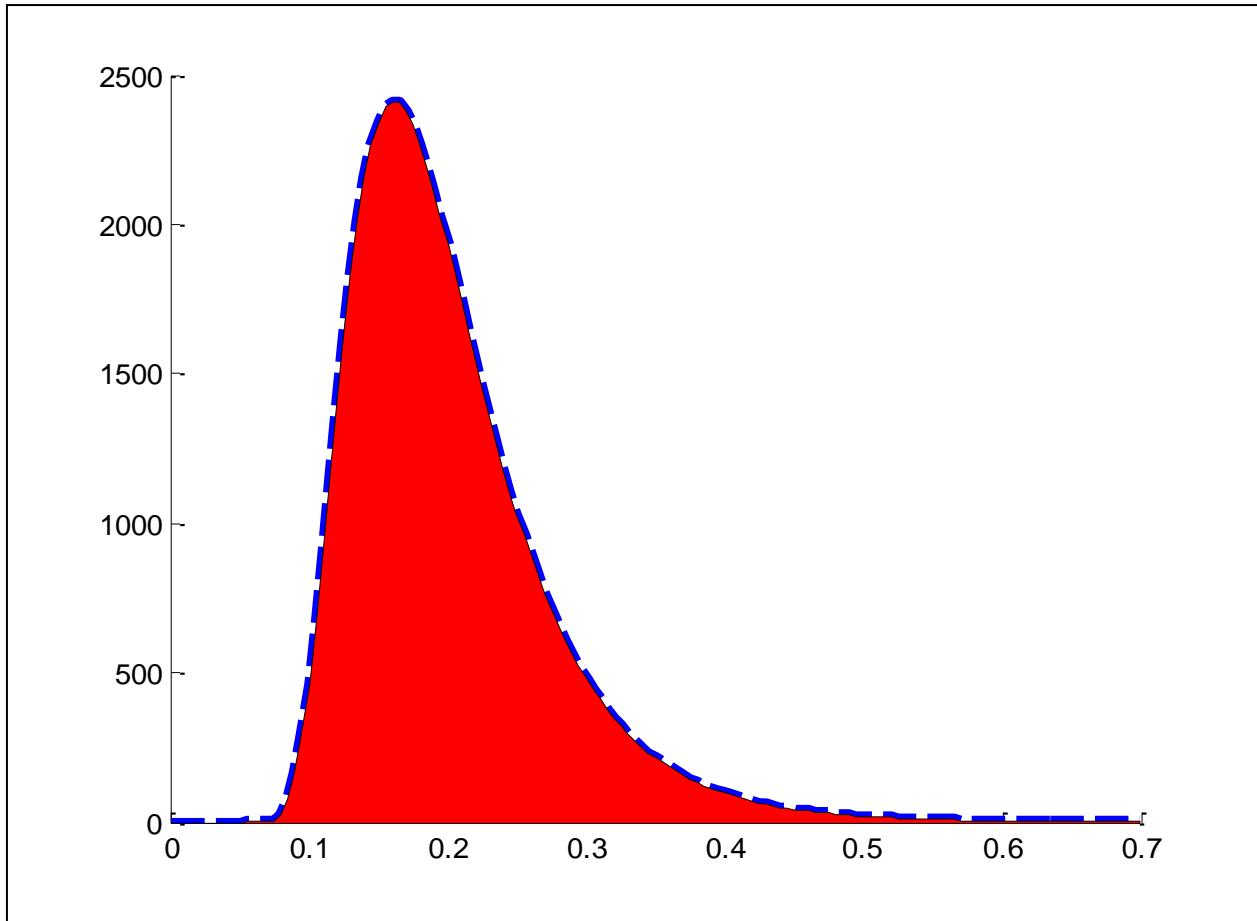


Figure 6.2 Distribution lognormale avec sa surface peinte en rouge

6.1.2.2 Contraintes sur la fonction Delta-Lognormale pour l'obtention des valeurs de $(\mu_1, \sigma_1, \mu_2, \sigma_2)$.

Les paramètres temporels $(\mu_1, \sigma_1, \mu_2, \sigma_2)$, nécessaires à la génération des profils de vitesses synthétiques, sont obtenus en posant des contraintes sur le mode de chacune des lognormales représentatives des sous-systèmes agoniste-antagoniste qui composent un profil de vitesse de pointage manuscrit.

Supposons le profil de vitesse lognormal de la Figure 6.2; 99,97% de la surface couverte par le signal sur cette courbe est comprise entre les temps :

$$t \in [e^{\mu-3\sigma}, e^{\mu+3\sigma}] \quad (6.4)$$

Ceci entraîne une durée du mouvement pour la composante neuro musculaire égale à :

$$\Delta t = e^{\mu+3\sigma} - e^{\mu-3\sigma} \quad (6.5)$$

D'autre part, les équations du mode et de la médiane d'une lognormale sont données par :

$$Mode = M = e^{\mu-\sigma^2} \quad (6.6)$$

$$Mediane = m = e^{\mu} = \frac{\Delta t}{2 \sinh 3\sigma} \quad (6.7)$$

En considérant le mode de la fonction lognormale et la durée du signal Δt comme point d'intérêt on obtient le système suivant :

$$\begin{cases} e^{\mu_{min}-\sigma_{max}^2} = M_{min} \\ e^{\mu_{max}} \sinh 3\sigma_{min} = 0.5\Delta t \\ e^{\mu_{min}} \sinh 3\sigma_{max} = 0.5\Delta t \\ e^{\mu_{max}-\sigma_{min}^2} = M_{max} \end{cases} \quad (6.8)$$

avec M_{min} et M_{max} les temps d'occurrence maximal et minimal respectifs du mode pour une lognormale. Ces valeurs sont choisies d'une façon empirique en fonction des observations effectuées sur les courbes expérimentales. Un exemple de valeurs utilisées pour M_{min} et M_{max} dans le système d'équations 6.8 pour un type de mouvement de durée $\Delta t = 0.8s$ serait de fixer les modes extrêmes à 10% et 60% de Δt respectivement. Cet exemple nous conduirait au problème suivant :

$$\begin{cases} e^{\mu_{min}-\sigma_{max}^2} = 0.1\Delta t \\ e^{\mu_{max}} \sinh 3\sigma_{min} = 0.6\Delta t \\ e^{\mu_{min}} \sinh 3\sigma_{max} = 0.6\Delta t \\ e^{\mu_{max}-\sigma_{min}^2} = 0.6\Delta t \end{cases}$$

La résolution de ce système donne des valeurs de μ et σ dans les intervalles :

$$\mu \in [\mu_{min}, \mu_{max}]$$

$$\sigma \in [\sigma_{min}, \sigma_{max}]$$

Rappelons que les intervalles de μ et σ ci dessus sont établies pour une lognormale. Étendons maintenant le concept à une Delta-Lognormale afin d'obtenir les intervalles de confiances sur les paramètres temporelles ($\mu_1, \sigma_1, \mu_2, \sigma_2$) représentatifs de l'interaction agoniste-antagoniste mis en jeux lors de l'établissement d'un mouvement tel que prédit par la théorie cinématique du mouvement humain rapide. Pour faciliter la lecture, rappelons ici l'équation de la Delta-Lognormale :

$$\Delta\Lambda(t; p) = D_1\Lambda(t; t_0, \mu_1, \sigma_1) - D_2\Lambda(t; t_0, \mu_2, \sigma_2) \quad (6.9)$$

L'agoniste et l'antagoniste dans cette équation sont représentés par les indices 1 et 2 respectivement.

Dans le cas d'une activité du système antagoniste au début du mouvement, nous estimons que sa réponse devient non significative après le maximum du profil de vitesse. La relation entre le mode et l'aire totale significative sous la courbe lognormale se traduit par :

$$e^{\mu_2+3\sigma_2} \leq e^{\mu_1-\sigma_1^2} \quad (6.10)$$

Pareillement, pour une activité du système antagoniste à la fin du mouvement, sa réponse commence à devenir prononcée après le maximum du profil de vitesse. Ceci se traduit par la relation 6.11 entre le mode et la valeur minimale de l'aire totale sous la courbe d'un profil Delta-Lognormale:

$$e^{\mu_2-3\sigma_2} \geq e^{\mu_1-\sigma_1^2} \quad (6.11)$$

Les contraintes analytiques sur la fonction lognormale et Delta-Lognormale, ainsi que les différentes observations faites sur les profils de vitesses recueillis avec les tablettes numériques, nous ont permis d'élaborer l'algorithme 6.1 qui permet de générer des courbes Delta-Lognormales avec des paramètres ajustés selon que l'activité du système antagoniste soit significative au début ou à la fin du mouvement.

Algorithme 6.1 : Génération des courbes Delta-Lognormales :

1. Fixer la durée Δt en secondes.
2. Fixer les valeurs de $[t_{0min}, t_{0max}]$
3. Déterminer et fixer les valeurs de Dmin et Dmax
4. Utiliser les équations 6.8 pour déterminer $[\mu_{min}, \mu_{max}]$, $[\sigma_{min}, \sigma_{max}]$.
5. Choisir des valeurs aléatoires pour $[\mu_1, \sigma_1]$ dans les intervalles de l'étape 4.
6. Déterminer les valeurs de $[\mu_2, \sigma_2]$ dans les intervalles de l'étape 4. Selon les conditions fixées par les équations 6.10 ou 6.11
7. Faire une boucle pour le nombre de profils désirés qui regroupe les étapes 2, à,6.
8. Calculer pour chaque paramètre la valeur du profil de vitesse en utilisant l'équation 6.9.

En résumé, l'algorithme 6.1 prend en entrée un intervalle de temps $[t_{0min}, t_{0max}]$ fixé à $t_{0min} = 50ms$ et t_{0max} fixé à la valeur du temps pour lequel l'amplitude du signal se trouve à 10% de sa valeur maximale (D_{max}) après t_{0min} sur le profil de vitesse. Les valeurs de D_{min} et D_{max} et les intervalles des valeurs de $[\mu_{min}, \mu_{max}]$, $[\sigma_{min}, \sigma_{max}]$ sont calculées en utilisant les équations 6.3 et 6.8 respectivement. L'étape 5 génère aléatoirement des valeurs pour les composantes temporelles agonistes en utilisant une distribution uniforme sous la contrainte des valeurs des intervalles $[\mu_{min}, \mu_{max}]$, $[\sigma_{min}, \sigma_{max}]$. L'étape 6 effectue la mutation afin d'obtenir des profils avec activités significatives de l'antagoniste en début ou en fin selon les conditions des équations 6.10 et 6.11. Les étapes 4, 5 et 6 de l'algorithme 6.1 sont conçues avec une flexibilité afin de permettre des ajustements à la convenance des programmeurs. Un cas de changement consiste à calculer les valeurs minimales et maximales d'intervalles pour chaque composante neuromusculaire en gardant le même t_0 afin de générer les profils de vitesse selon la position désirée des différents modes des composantes neuromusculaires. Dans ce cas particulier, l'algorithme 6.1 devient l'algorithme 6.2.

Algorithme 6.2 : Génération des courbes Delta-Lognormales :

1. Fixer la durée du temps Δt en secondes.
2. Fixer les valeurs de $[t_{0min}, t_{0max}]$
3. Déterminer et fixer les valeurs de D_{min} et D_{max}
4. Utiliser les équations 6. 8 pour déterminer $[\mu_{1min}, \mu_{1max}]$, $[\sigma_{1min}, \sigma_{1max}]$.
5. Utiliser les équations 6.8 pour déterminer $[\mu_{2min}, \mu_{2max}]$, $[\sigma_{2min}, \sigma_{2max}]$. Selon les conditions fixées par les équations 6.10 et 6.11
6. Boucler selon le nombre de profils désirés.
7. Choisir aléatoirement des valeurs pour $[\mu_1, \sigma_1]$ dans les intervalles déterminés à l'étape 4.
8. Choisir aléatoirement des valeurs pour $[\mu_2, \sigma_2]$ dans les intervalles déterminés à l'étape 5.
9. Choisir aléatoirement la valeur de t_0
10. Choisir aléatoirement la valeur de D_1 et D_2
11. Calculer pour chaque paramètre la valeur du profil de vitesse en utilisant l'équation 6.9

Les algorithmes 6.1 et 6.2 ont permis la construction d'une banque de plus de neuf mille courbes artificielles selon les différentes classes établis à la Table 6.1. Une des questions qui pourraient interroger le lecteur serait de savoir la correspondance de ces profils de vitesses du temps vers l'espace afin de visualiser les trajectoires rectilignes associées. Un tel exercice a été initié dans [76] où il présente une variabilité spatiale des profils de vitesses de mouvements simples en utilisant la théorie cinématique du mouvement rapide et son modèle sigma-lognormal. Dans cette thèse nous nous sommes concentrés sur l'aspect temporel du profil de vitesse, car c'est le point d'entrée de l'algorithme **S&E**. De plus, la variation des positions angulaires reliée à chaque mouvement dans l'espace ne se prête pas facilement à une modélisation Delta-Lognormale.

6.1.3 Résultats du processus de génération de profils de vitesses synthétiques.

Tel que décrit ci-dessus, un profil de vitesse peut avoir jusqu'à deux passages par zéro et conduire à un maximum de trois pics. La figure 6.3 présente toutes les classes de profils de vitesse suivant

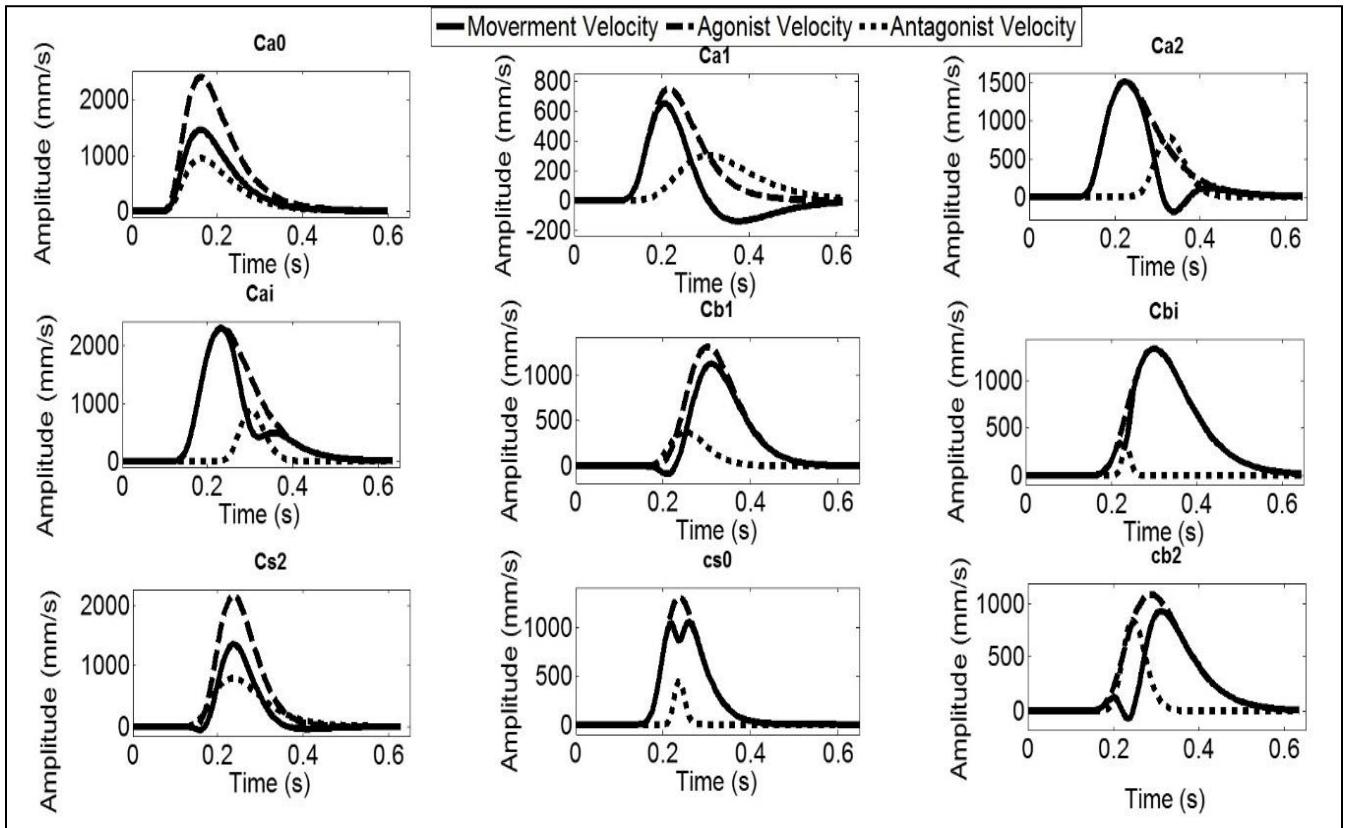


Figure 6.3 Toutes les classes des courbes delta-lognormales : en traits continus les courbes Delta-Lognormales obtenues par soustraction des deux courbes en pointillés

cette terminologie. Elle représente l'ensemble des neuf classes de profils de vitesse des traits simples construits. Pour les classes Ca, l'antagoniste (lignes pointillées) est dominant après le pic d'activité de l'agoniste (ligne pointillée). Le contraire est observé pour les classes Cb. Les classes Cs2, Ca0 et Cs0 représentent des cas particuliers dans lesquels les composantes agonistes et antagonistes sont parfaitement synchrones. De plus, chacune d'elle a ses propres caractéristiques. Par exemple dans la classe Cs0, les pics maximaux de l'antagoniste et de l'agoniste sont parfaitement synchrones, et dans le (Cs2), l'antagoniste est dominant à chaque extrémité du profil de vitesse par rapport à l'agoniste.

6.1.4 Synthèse des travaux sur la génération de profils de vitesse synthétiques

Cette section propose différents profils de vitesse manuscrite artificielle identifiés et caractérisés à partir des morphologies de traits manuscrits et leurs profils de vitesse associés. La réalisation de ces courbes artificielles s'est effectuée en utilisant le modèle Delta-Lognormal et les algorithmes 6.1 et 6.2. Une banque de neuf mille (9000) profils de vitesses Delta-Lognormaux artificielles a été construite automatiquement à partir de ces algorithmes. La banque se compose de 1000 courbes pour chacune des classes identifiées et présentées à la Table 6.1. Ces courbes ont permis d'entraîner l'algorithme de **S&E** en validant la précision des résultats des extractions des courbes synthétiques. On retrouve dans la Table 6.1 la nomenclature associée aux différentes classes de profils identifiés et présentés à la Figure 6.3. Les trois grandes classes représentatives de l'interaction agoniste antagoniste nécessaire à l'établissement du mouvement sont bien représentées dans les mouvements conçus.

Différentes relations entre les propriétés observées et l'équation du Delta-Lognormale sont illustrées dans les équations 6.2 à 6.11. Ces équations permettent de conditionner la fonction Delta-Lognormale afin de générer automatiquement (à l'aide des algorithmes 6.1 et 6.2) les courbes de la figure 6.3 représentatives des trois grandes classes de l'interaction agoniste antagoniste. L'ensemble de la banque de données artificielle permet d'entraîner l'algorithme **S&E** (chapitre 5) dans sa capacité à retrouver avec précisions les extractions les valeurs des paramètres ayant conçus les courbes artificielles

NB : la nomenclature Ca, Cb et Cs équivaut à celle de la table 6.1 avec a pour *after*, b pour *before*, et s pour *simultaneous*.

6.2 Résultats détaillés des délais d'extractions des paramètres Delta-Lognormaux par l'algorithme S&E

Cette section présente les résultats complémentaires à l'algorithme présenté au chapitre 5 de cette thèse. Les résultats portent sur les temps de calculs requis à l'algorithme **S&E**, pour recouvrir les paramètres Delta-Lognormaux des profils de vitesses synthétiques et réelles. L'algorithme a été implémenté sur un processeur Intel core (i7) 3.60 GHz de fréquence et 32 Go de Ram. Les tests ont été effectués avec différents mouvements de pointages synthétiques et réels. Les mouvements synthétiques tels que décrits à la section 6.1 sont des profils de vitesses manuscrits ayant leurs valeurs de paramètres Delta-Lognormaux connus. Les profils de vitesses réelles sont des mouvements de pointages manuscrits simples obtenus à partir des tracés rapides sur des tablettes numériques [65]. Les objectifs visés par les tests selon les types de données sont les suivants : Dans le cas des données synthétiques, les tests ont pour objectif d'évaluer la précision avec laquelle l'algorithme **S&E** extrait les paramètres des profils de vitesses. Soit la capacité de l'algorithme à retrouver avec précision les bonnes valeurs des paramètres Delta-Lognormaux ayant conçus les différents profils de vitesses synthétiques. Dans le cas des données réelles, les tests ont pour but ;

1. D'évaluer la capacité de l'algorithme **S&E** à les extraire dans un intervalle de confiance suivant un rapport signal sur bruit défini à 25 dB.
2. D'évaluer les délais moyen requis pour une extraction.

L'évaluation de l'extracteur **S&E** dans le cas de la précision des valeurs extraites pour des vitesses synthétiques et réelles sont présentes dans [57] [72] et au chapitre #5 de cette thèse. Cette section présente les résultats détaillés des temps moyens d'exécutions nécessaires à l'algorithme **S&E** pour recourir à une extraction d'un profil de vitesse synthétique et réelle.

6.2.1 Échantillons de tests pour l'évaluation des délais d'extractions des profils de vitesses synthétiques par l'algorithme S&E conçus

L'échantillon retenu pour l'évaluation des délais dans le cas des données synthétiques est de cent cinquante (150) profils de vitesses par groupes de classe de profils Delta-Lognormaux identifiés (voir section 6.1). Rappelons que dans le cas des profils de vitesses synthétiques, une classe est la forme expérimentale que peut avoir un profil de vitesse synthétique identifié et caractérisé à partir

des morphologies d'un trait manuscrit et son profil de vitesse associé. Pour cette expérience et pour des raisons de simplicités, les neuf classes identifiées ont été réparties en trois grands groupes selon l'activité des composantes neuromusculaires en jeux dans l'accomplissement du mouvement :

1. Activités significatives de la composante antagoniste au début du mouvement (**ASAD**).
2. Activités significatives des composantes antagoniste et agoniste simultanément. (**ASAS**)
3. Activités significatives de la composante antagoniste à la fin du mouvement. (**ASAF**)

Nous avons considéré que tous les mouvements dans un groupe sont indépendants et nécessitent les mêmes efforts musculaires.

Le tableau 6.2 résume l'ensemble de l'échantillon pour des données synthétiques.

Table 6.2 Échantillons de synthétiques pour l'évaluation des temps d'exécution de l'algorithme S&E

Noms des groupes des classes	Nombre total de traits
Activités significatives de la composante antagoniste au début du mouvement.	150
Activités significatives des composantes antagoniste et agoniste simultanément	150
Activités significatives de la composante antagoniste à la fin du mouvement	150

6.2.2 Échantillons de tests pour l'évaluation des délais d'extraction des profils de vitesses synthétiques par l'algorithme S&E conçus

L'échantillon des données réelles est constitué de 40 profils de vitesses par groupe d'individus ayant participé aux différentes collectes. Les différents groupes de participants sont les suivants : athlètes de football universitaire, personnes atteintes de troubles d'attention (**TDAH**), personnes atteintes de la maladie de parkinson et personnes ayant manifesté des symptômes d'une commotion cérébrale. La sélection a été effectuée de manière aléatoire dans la banque de 500 profils de vitesses rassemblés pour entraîner l'algorithme **S&E** détaillé au chapitre 5. Le choix de 40 profils par groupe se justifie par la disponibilité limitée des données pour les participants manifestant une commotion cérébrale et la nécessité d'avoir un échantillon de taille égale pour l'évaluation des

délais d'exécution intragroupes et intergroupes. Dans ce test, nous avons considéré que tous les mouvements sont indépendants et ont subi le même protocole de prise de donnée.

Le tableau 6.3 résume l'ensemble de l'échantillon de test avec les données réelles.

Table 6.3 Échantillon de données réelles pour l'évaluation des temps d'exécutions de l'algorithme S&E

Types de participants	Nombre total des traits simples
Athlètes de football (FP)	40
Personnes atteintes de déficit de l'attention (TDAH)	40
Personnes atteintes de la maladie de parkinson (PK)	40
Personnes ayant une commotion cérébrale (CC)	40

6.2.3 Résultats des tests sur les données synthétiques

Le tableau 6.4 présente le rapport détaillé des délais moyens nécessaires à l'algorithme **S&E** pour recouvrir les valeurs optimales des profils de vitesses synthétiques des trois groupes de l'échantillon présenté au tableau 6.2. Dans cette expérience, une solution est optimale si l'erreur quadratique moyenne entre le profil synthétique et l'approximation Delta-Lognormale est supérieure à 10^{-5} (mm/s)². Ce seuil correspond à un recouvrement des valeurs des paramètres de l'ordre du millième 10^{-3} . Les résultats recueillis montrent un temps moyen d'extraction de 4,65mn, 5,74mn et 5,21mn minutes respectivement pour un profil (**ASAD**) (**ASAS** (**ASAF**)). Ces délais montrent que l'algorithme **S&E** met plus de temps à recouvrir les paramètres des profils de vitesse dont la composante antagoniste est noyer dans l'agoniste. En effet, l'extraction des paramètres des profils de vitesses synthétiques donc l'antagoniste est complètement noyée dans l'agoniste peuvent présenter des linéarités dans les valeurs des paramètres des deux composantes neuromusculaires et être approximées par une lognormale. Cette particularité force le passage de l'algorithme **S&E** dans une longue phase d'extension des solutions probables et rend difficile l'extraction de ces profils. Les variances respectives de 5,35, 6,2 et 8,46. des profils (**ASAD**) (**ASAS**) (**ASAF**) présentés au tableau 6.3 sont relativement élevés et traduisent la forte dispersion des délais dans les différents groupes. Cette variabilité est cohérente avec la réalité, car pour un groupe on peut y avoir une infinité de traits différents qui ne traduisent pas forcément une un écart

significatif dans les délais de leur extraction. Dans tous les groupes, le cas où la composante antagoniste est complètement noyée dans l'agoniste allonge les délais.

Le tableau 6.5 présente les résultats détaillés de l'ANOVA à un facteur sur les délais entre les différents groupes **(ASAD)**, **(ASAS)**, **(ASAF)**. On peut constater sans surprise que F critique $> F$ et $P=0,001 < 5\%$ donc il existe des différences statistiquement significatives entre les délais d'extraction des trois groupes de profils synthétiques. L'ANOVA nous permet de conclure de l'existence des différences entre les groupes, mais ne nous donne aucune indication sur lequel des groupes présentent des différences significatives par rapport à un autre.

Le tableau 6.6 présente les résultats des tests de Student (**T test**) avec une correction de Bonferroni appliquée au seuil de probabilité sur les trois groupes. La correction de Bonferroni est due à la disparité que présentent les variances observées au tableau 6.3. Les résultats du **T test** montre une valeur de $P = 0,000735766 < 0,16$ dont il existe une différence significative entre les délais moyens d'extraction observés entre le groupe **(ASAD)** (moyenne=4,64+-2,3 mn) et **ASAS** (moyenne=5,74 +- 2,9). Ce dernier implique que, le temps d'exécution des profils de vitesses dont l'activité de la composante antagoniste est significative au début du mouvement est moins élevé que les temps d'exécutions des profils de vitesses dont l'activité de la composante antagoniste est synchrone avec l'agoniste.

Table 6.4 Résultats détaillés des délais moyens d'extraction des paramètres Delta-Lognormaux des profils de vitesses synthétiques par l'algorithme S&E

Groupes	Nombre	Somme	Moyenne	Variance
(ASAD)	150	697,09	4,64	5,35
(ASAF)	150	782,5	5,22	6,19
(ASAS)	150	861,25	5,74	8,45

Table 6.5 Résultats de l'ANOVA sur les délais moyens d'extraction des paramètres Delta-Lognormaux des profils de vitesses synthétiques par l'algorithme S&E

Source des variations	Somme des carrés	Degré de liberté	Moyenne des carrés	F	Probabilité	Valeur critique pour F
Entre Groupes	89,87	2	44,94	6,74	0,0013	3,01
A l'intérieur des groupes	2980,32	447	6,66			
Total	3070,205832	449				

Table 6.6 Résultats des tests de Student sur les délais d'extractions des données synthétiques par l'algorithme S&E

Groupes	Valeur P (T test)	Signifiant?	Alpha ANOVA	Alpha correction de Bonferroni
(ASAF) Vs (ASAD)	0,15	non	0,05	0,016
(ASAD) Vs (ASAS)	0,00073	oui	0,05	0,016
(ASAS) Vs (ASAF)	0,056	non	0,05	0,016

6.2.4 Résultats des tests sur les données réelles

Le tableau 6.7 présente le rapport détaillé des délais moyens nécessaires à l'algorithme **S&E** pour effectuer une extraction des paramètres Delta-Lognormaux sur des profils de vitesses manuscrites des quatre groupes (**FP**, **TDAH**, **PK**, **CC**) de notre échantillon présenté au tableau 6.3. Dans cette expérience la solution optimale est l'unique boite ou vecteur d'intervalles ayant approximé le profil de vitesse manuscrit avec un seuil (rapport signal sur bruit) minimal de 25dB. On observe dans le tableau 6.7, des temps moyens d'extraction respectifs de 8,97min, 10,7min, 10,7min, et 9,27min minutes pour des profils de vitesse (**FP**) (**TDAH**) (**PK**) et(**CC**). Ce résultat montre que l'algorithme **S&E** met plus de temps à recouvrir les paramètres des profils de vitesse des participants **PK** et **TDAH**. Pareillement, on observe dans le tableau 6.7 des variances respectives de 1,84, 1,45, 3, et 1.89 pour les profils (**FP**) (**TDAH**) (**PK**) (**CC**). Ces variances relativement élevées traduisent la forte dispersion des temps d'extraction des différents groupes.

Le tableau 6.8 présente les rapports détaillés de l'ANOVA à un facteur sur les délais entre les différents groupes (**FP**) (**TDAH**) (**PK**) (**CC**). Ici encore, $F_{critique} > F$ et $P = 3,9786E-10 < 5\%$ donc il existe des différences statistiquement significatives entre les délais d'extraction des quatre groupes de profils de vitesses réelles. Les variabilités observées dans les différentes variances (tableau 6.7) ne permettent pas d'identifier lequel des groupes présente une moyenne de temps élevé par rapport à un autre. Le test de Student présenté au tableau 6.9 permet d'y apporter des réponses. Le rapport détaillé du test de Student (tableau 6.9) avec une correction de Bonferroni appliquée au seuil de probabilité sur les quatre groupes (**FP**) (**TDAH**) (**PK**) (**CC**) montre que : le temps moyen d'extraction des profils de vitesses des footballeurs professionnelles (**FP**) est significativement inférieur à celui des personnes atteintes des troubles de l'attention (**TDAH**) ($P = 5,39473E-08 < 0.001$; moyenne=10.7 min +-1,2) et des personnes atteintes de la maladie de parkinson **PK** ($P = 3,81874E-06 < 0.001$; moyenne=10.7 min +-1,73). Toujours dans le même sens,

il ressort du tableau 6.9 que le temps moyen d'extraction des paramètres des profils de vitesses des personnes présentant des symptômes d'une commotion cérébrale est inférieur de manière significative à celui des personnes atteintes de la maladie de parkinson PK ($P=3,81874E-06<0.001$; moyenne=10.7 min +-1,73) et des individus atteints des troubles de l'attention **TDAH** ($P=5,39473E-08<0.001$; moyenne=10.7 min +-1,2).

Table 6.7 Résultats détaillés des délais moyens d'extraction des paramètres Delta-Lognormaux des profils de vitesses synthétiques par l'algorithme S&E

Source des variations	Somme des carrés	Degré de liberté	Moyenne des carrés	F	Probabilité	Valeur critique pour F
Entre Groupes	101,57	3	33,85	18,09	3,97E-10	2,66
A l'intérieur des groupes	291,93	156	1,87			
Total	393,51	159				

Table 6.8 Résultats de l'ANOVA sur les délais moyens d'extraction des paramètres Delta-lognormaux des profils de vitesses des données réelles par l'algorithme S&E

Groupes	Nombre de personnes	Somme min	Moyenne min	Variance
Athlète de football (FP)	40	358,98	8,97	1,84
Individus atteints de troubles de l'attention (TDAH)	40	428,05	10,70	1,45
Individus atteints de la maladie de parkinson (PK)	40	428,22	10,70	3
Individu présentant une commotion cérébrale (CC)	40	370,93	9,27	1,18

Table 6.9 Résultats des tests de Student sur les délais d'extraction des données réelles par l'algorithme S&E

Groupes	Valeur P (T test)	Signifiant ?	Alpha ANOVA	Alpha avec correction de Bonferroni
(FP) Vs (TDAH)	5,39E-08	oui	0,05	0,0125
(FP) Vs (PK)	3,81E-06	oui	0,05	0,0125
(FP) Vs (CC)	0,280	non	0,05	0,0125
(TDAH) Vs (PK)	0,98	non	0,05	0,0125
(TDAH) Vs (CC)	3,67E-07	oui	0,05	0,0125
(PK) Vs (CC)	3,10E-05	oui	0,05	0,0125

6.2.5 Synthèse des résultats sur les temps d'extractions des différents profils.

Cette section présente les résultats détaillés des délais sur l'extraction des profils de vitesses synthétiques et réelles avec l'algorithme **S&E**. Dans le cas des données synthétiques, l'échantillon de test est constitué des profils de vitesses Delta-Lognormaux des neuf classes identifiées et présentées à la section 6.1. Les classes ont été regroupées en trois principaux groupes (**ASAD**) (**ASAS**) (**ASAF**) afin de simplifier l'analyse statistique. L'algorithme **S&E** est capable d'extraire tous les profils de vitesse de l'échantillon dans un délai moyen de 4,7min 5,21 min et 5,75 min pour chacun des profils (**ASAD**) (**ASAS**) (**ASAF**) respectivement. L'ANOVA suivie d'un test T a permis de conclure que le temps d'exécution des profils de vitesses dont l'activité de la composante antagoniste est significative au début du mouvement est moins élevé que les temps d'exécutions des profils de vitesses dont l'activité de la composante antagoniste est synchrone avec l'agoniste.

L'échantillon des données réelles contient des athlètes professionnelles, des personnes atteintes de maladies neurologiques (parkinson), des individus présentant des symptômes de commotions cérébrales et des individus présentant des troubles de l'attention. L'extraction des paramètres lognormaux par l'algorithme **S&E** de ces différents groupes s'effectue dans un délai moyen compris entre 8 à 10 min. L'ANOVA suivie d'un test T a mené à deux conclusions :

1. Le temps moyen d'extraction des profils de vitesses des footballeurs professionnelles (**FP**) est significativement inférieure à celui des personnes atteintes des troubles de l'attention **TDAH** et des personnes atteintes de la maladie de parkinson **PK**.
2. Le temps moyen d'extraction des paramètres des profils de vitesses des personnes présentant des symptômes d'une commotion cérébrale est inférieur de manière significative à celui des personnes atteintes de la maladie de parkinson **PK** et des individus atteints des troubles de l'attention **TDAH**.

Bien que les résultats présentés ci-dessus semblent encore élevés ils constituent néanmoins une avancée considérable comparativement à l'algorithme proposé dans [50] pour qui les extractions mêmes nécessitent un délai s'étalant sur plusieurs jours.

6.3 Synthèse des développements inédits

Ce chapitre rapporte l'ensemble des développements non publiés associés à la réalisation de l'algorithme **S&E** développé dans cette thèse. On y retrouve:

- Les techniques permettant la création et la génération automatiques des profils de vitesses de mouvements de pointage synthétiques à partir des modèles lognormaux et des observations faites sur les profils de vitesses de mouvements manuscrits.
- Les résultats détaillés des temps moyens d'exécutions nécessaires à l'algorithme **S&E** pour recourir à une extraction d'un profil de vitesse synthétique et réelle.

Ces développements non publiés regorgent des techniques utiles tant à l'algorithme **S&E** développé dans cette thèse qu'à toutes autres applications utilisant des mouvements de pointages manuscrits. Les statistiques autour des délais moyens d'extraction révèlent des temps d'exécution en moyenne de 5 minutes pour des données synthétiques et 10 minutes pour des données réelles. Ces résultats, bien qu'élevés, constituent une avancée considérable dans l'état de l'art sur l'extraction des paramètres Delta-Lognormaux en utilisant un algorithme de type **S&E**.

CHAPITRE 7 DISCUSSION GÉNÉRALE

Ce chapitre débute par un rappel des fondements théoriques de la méthode de séparation et évaluations pour des problèmes d'optimisation. Par la suite, une discussion générale autour des techniques fondamentales de l'algorithme **S&E** conçus est ouverte. Le chapitre se ferme avec une discussion entre le modèle Delta-Lognormale de la théorie cinématique du mouvement rapide et l'extracteur **S&E**.

Cette thèse fait état d'un outil d'analyse des mouvements manuscrits de pointages utilisant un algorithme à caractère global de type **S&E**. La méthode de séparation et évaluation est une procédure qui consiste à effectuer la recherche d'une solution optimale dans un ensemble fini de solutions candidates. Elle est caractérisée par deux procédures fondamentales qui forment son nom :

1. La séparation qui consiste à diviser un ensemble en sous-ensembles.
2. L'évaluation qui consiste à évaluer les sous-ensembles entre eux de manière optimiste.

La séparation est caractérisée par la création d'un ensemble φ dans lequel on isolera des sous-ensembles de φ de plus en plus réduits. Elle nécessite des procédures de réduction de l'ensemble Φ sans perte de la solution optimale. L'évaluation est caractérisée par des règles d'examen et de sélection qui, en fonction du ou des critères, sert à éliminer des sous-ensembles de Φ ne garantissant pas l'optimalité et à sélectionner parmi ceux qui restent, celui auquel le principe de séparation doit à nouveau être appliqué.

7.1 Ensemble Φ de la procédure de séparation

L'algorithme **S&E** conçus permet d'extraire les paramètres Delta-Lognormaux des profils de vitesses de mouvements de pointages manuscrits. Dans cet algorithme, l'ensemble Φ des solutions potentielles est un vecteur d'intervalles (boîte) de sept paramètres Delta-Lognormaux dont l'enveloppe Delta-Lognormal renferme la totalité du profil de vitesse à optimiser. La boîte est conçue en utilisant un optimisateur local (**IIX**) qui produit sept valeurs de paramètres initiaux, qui sont par la suite agrandis (proportionnellement) de manière itérative de sorte que l'enveloppe Delta-Lognormal résultante encadre complètement le profil de vitesse manuscrit à approximer. La boîte initiale de recherches obtenues (ensemble φ) est un vecteur d'intervalles garantissant une

borne minimale et maximale globale, assez large et inclusive de la solution optimale à recouvrir. L'enveloppe est l'évaluation de la fonction Delta-Lognormale par le vecteur d'intervalles (boite) en utilisant l'arithmétique par intervalles et son théorème fondamental. La conception de la boite initiale est une étape cruciale dans un algorithme **S&E** car elle permet de circonscrire de manière géométrique l'espace de recherche en garantissant l'inclusion complète du signal à l'étude dans l'enveloppe formée par la boite initiale. Cette approche géométrique est largement plus inclusive et complémentaire en comparaison avec celles proposées dans [53] [23] où seulement quelques points spécifiques du profil de vitesse à approximer sont garantis de résider dans l'enveloppe de départ. Les figures 7.1 à 7.4 montrent l'ensemble des résultats obtenus avec **S&E** pour différentes contraintes de relaxation sur la solution désirée. Les courbes en bleu et rouge forment l'enveloppe Delta-Lognormale pour le profil de vitesse (courbe en verte) dont les paramètres lognormaux sont à recouvrir. Dans ces graphiques, on s'aperçoit que l'algorithme **S&E** rassemble dans une boite englobante (Figure 7.1) l'ensemble des cas possibles pour une courbe donnée de sorte que différentes solutions peuvent en être extraites selon les rapports signal sur bruit possible et les applications visées (Figure 7.2 à 7.4). Ce type de résultat avec intervalle de confiance construit automatiquement et sans recours à une quelconque statistique lui confère une place de choix pour des applications où une marge d'erreur dans le diagnostic n'est pas tolérable.

7.2 Sous-ensemble réduit de Φ de la procédure de séparation

Les sous-ensembles de la boite englobante (Φ) sont obtenus en divisant une boite en N sous boites, N étant le nombre de dimensions de la boite. Ce qui engendrerait un maximum de 2^N boites possibles. Pour des problèmes à plusieurs dimensions, une division sur toutes les dimensions peut rapidement faire exploser le nombre de boites et entraîner des coûts de calcul très élevés. Il est donc important de trouver le nombre de divisions nécessaire à faire progresser l'algorithme vers l'obtention d'une solution meilleure. Rappelons que le processus de division a pour but de réduire l'espace de recherche tout en forçant l'algorithme à progresser vers un minimum local meilleur ou atteindre le minimum global. Le problème considéré à cette étape de l'algorithme est la capacité d'une boite candidate à produire des sous-boites en quantité raisonnable et capable non seulement de réduire l'espace, mais aussi de produire au moment de leurs évaluations des solutions meilleures que celles connues avant la division. La littérature propose plusieurs heuristiques plus ou moins efficaces pour résoudre ce problème [50] [56], dans l'algorithme **S&E** nous avons choisis la

descente du gradient associé à la largeur de la dimension considérée dans une boîte pour nous guider dans le choix de la variable de coupure. L'algorithme **S&E** divise la boîte candidate en deux dans la direction où la largeur du produit entre le gradient et la dimension considérée est maximale (voir Splitting rule Chapitre 5). Ce choix judicieux combine l'efficacité de la descente du gradient, mais aussi tient compte de la largeur de la dimension. La publication « **A Global Optimization Algorithm for the Estimation of Delta-Lognormal Parameters Characterizing Pointing Gestures** » qui a été soumise dans la revue « **Pattern Analysis and Machine Intelligence** » (**PAMI**) présente des tests de différentes stratégies de division. Il en ressort clairement que la division combinant gradient et dimension est la meilleure, pour une extraction d'un profil de vitesse. Le temps d'exécution passe d'au moins 1h avec les autres procédures à environs 15 mn lorsque la coupure se fait en utilisant le gradient.

De manière plus générale, la procédure de séparation de l'algorithme **S&E** est la combinaison de la construction de l'espace de recherche et la division de cet espace en sous-espaces. Dans l'extracteur **S&E** construit, cette procédure requiert à peine quelques millisecondes pour une exécution complète sur un ordinateur conventionnel de ses fonctions principales (fonctions de constructions des bornes supérieures et inférieures). Ce module a par ailleurs été implémenté dans l'algorithme de [21] (voir Annexe B) et nous avons observé une diminution du délai d'exécution (702ms contrairement à 50s) pour 11 itérations dans la boucle principale du programme pour une extraction des paramètres des profils de vitesses synthétiques. Les résultats des tests présentés dans la contribution « **On the Use of Interval Arithmetic for the Branch and Bound Delta-Lognormal Parameter Extraction of Rapid Human Movements** » (Annexe B) montrent de façon générale la polyvalence de la procédure de séparation de l'algorithme **S&E**; ce dernier rend l'extracteur de [50] exploitable dans le cas des données synthétiques.

En définitive, la procédure de séparation de l'algorithme **S&E** enlève l'ambiguïté liée au filtrage et à l'attribution de distorsions à certaines parties du signal de base. Il évite les doutes sur la qualité des résultats obtenus, car la totalité du signal est prise en compte.

7.3 Discussion sur la procédure d'évaluation

Le deuxième point abordé se rapporte à la procédure d'évaluation proposée dans de l'algorithme **S&E** conçus. Cette procédure fonctionne par énumération de toutes les solutions de l'ensemble de

bases qui respectent tous les critères d'inclusions fixés par la procédure de séparation. Une compétition s'en suit alors entre chaque solution potentielle afin de retenir la meilleure courante. La compétition consiste à attribuer un cout à chaque candidate potentielle et ensuite les comparer tout en retenant à chaque étape le meilleur coût garantissant la minimisation (maximisation) de la fonction de coût à optimiser. En plus de la compétition, chaque candidate se voit subir différents tests permettant de conserver ou rejeter une solution candidate. Cette étape permet d'élaguer certaines solutions n'offrant aucune garantie d'optimalité. Dans l'algorithme **S&E** proposé, l'énumération des candidates est régie par cinq règles conçues à partir des propriétés de la fonction de coût (voir chapitre 5) et du théorème fondamental de l'arithmétique par intervalles. Des heuristiques sont parfois utilisées pour insérer des solutions jugées prometteuses dans la liste de travail, mais n'entraînent aucune conséquence indésirable sur la garantie d'obtention de l'optimum, car la phase d'évaluation de **S&E** force l'énumération de toutes les solutions existantes au moins une fois. De manière générale, la procédure d'évaluation d'un algorithme **S&E** requiert des temps de calcul élevés, car aucune approximation n'est possible, l'obtention de l'optimum dépend uniquement de l'efficacité des tests de rejets et de la capacité de la boîte en cours d'évaluation à élaguer les autres boîtes potentiellement prometteuses.

Dans le cas de l'algorithme **S&E** conçus, le module d'évaluation garantit toujours la solution optimale bien sûr au prix d'un temps relativement faible en comparaison avec le précédent algorithme [50]. L'exécution d'une itération complète dans la boucle principale oscille entre 60 à 70 ms et l'obtention de la solution optimale pour un profil de vitesse varie entre 9 à 15 minutes. Ce résultat est amélioré d'un facteur de 300 à 400 fois en comparaison avec [50]. En général, concernant la problématique, ce module produit des solutions globales dans un délai court, mais pas assez pour des utilisations en temps réel. On pourrait relativiser sur notre interprétation de temps réel, mais nous estimons que des délais de 15 minutes en moyenne permettent déjà certaines évaluations en clinique.

Sur le plan architectural, le présent algorithme **S&E** tels que construit repose sur des fondements solides sur lesquels toutes les architectures de niveau supérieur pourraient aisément se greffer. Chaque partie est conçue de manière indépendante des autres et est gouvernée par une règle mathématique. Cette modularité laisse beaucoup de flexibilité à toute amélioration ou changement dépendamment de l'application et de l'utilisation si nécessaire. Malgré les progrès observés sur les temps d'exécutions pour l'obtention des solutions, la lenteur dans l'énumération des solutions reste

un problème. Le parallélisme inhérent de l'algorithme ainsi qu'une bonne organisation des différentes fonctions du problème sont des exemples de solutions proposées qui pourraient être exploitées. Il serait intéressant de tester les performances du présent extracteur **S&E** en y ajoutant un optimiseur local pour chaque boîte constituant un sous-espace probable. On combinerait alors, le succès des propositions de la littérature (optimiseur local) [22] avec la garantie de succès du présent extracteur.

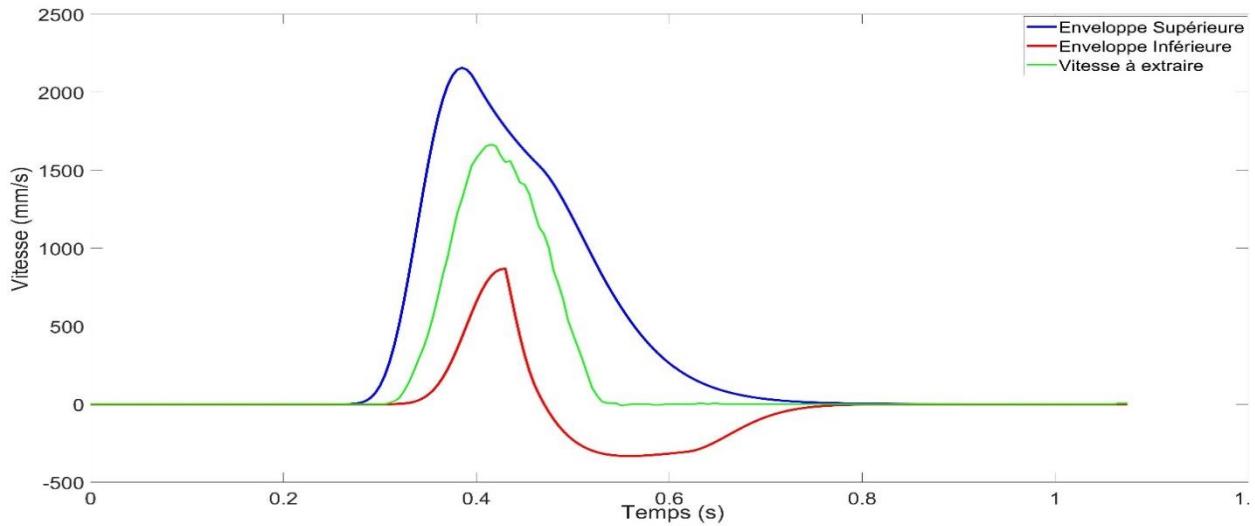


Figure 7.1 Boîte englobante obtenue avec l'algorithme S&E pour SNR fixé à 20dB

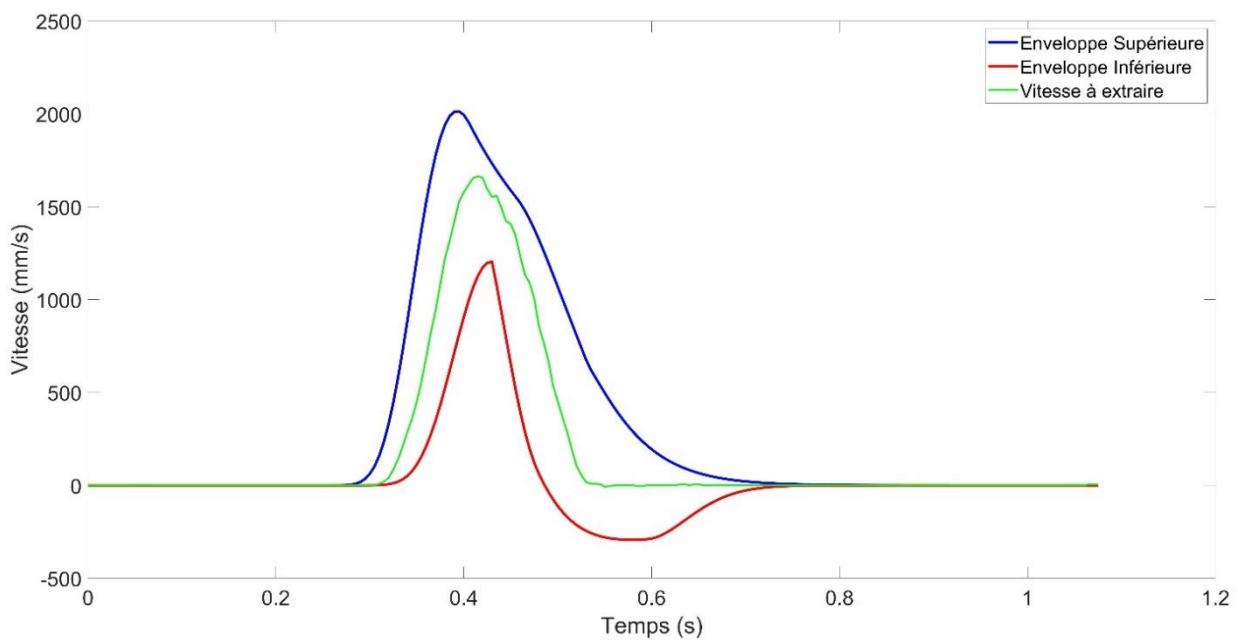


Figure 7.2 Boîte englobante obtenue avec l'algorithme S&E pour SNR fixé à 23dB

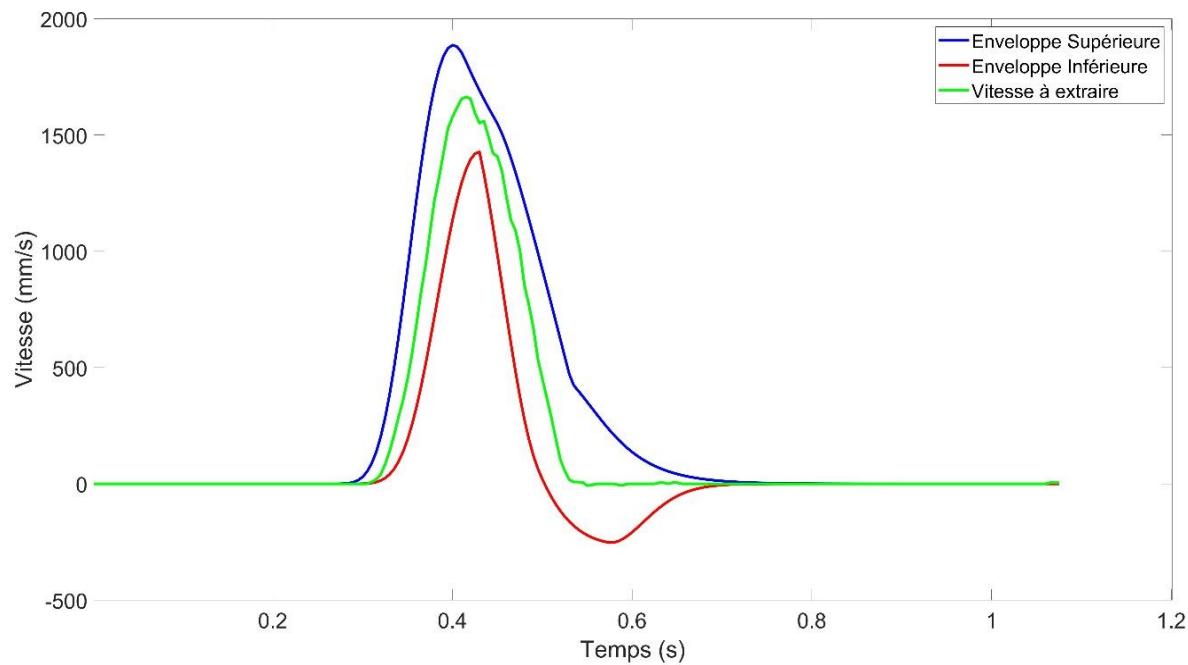


Figure 7.4 Boite englobante obtenue avec l'algorithme S&E pour SNR fixé à 25dB

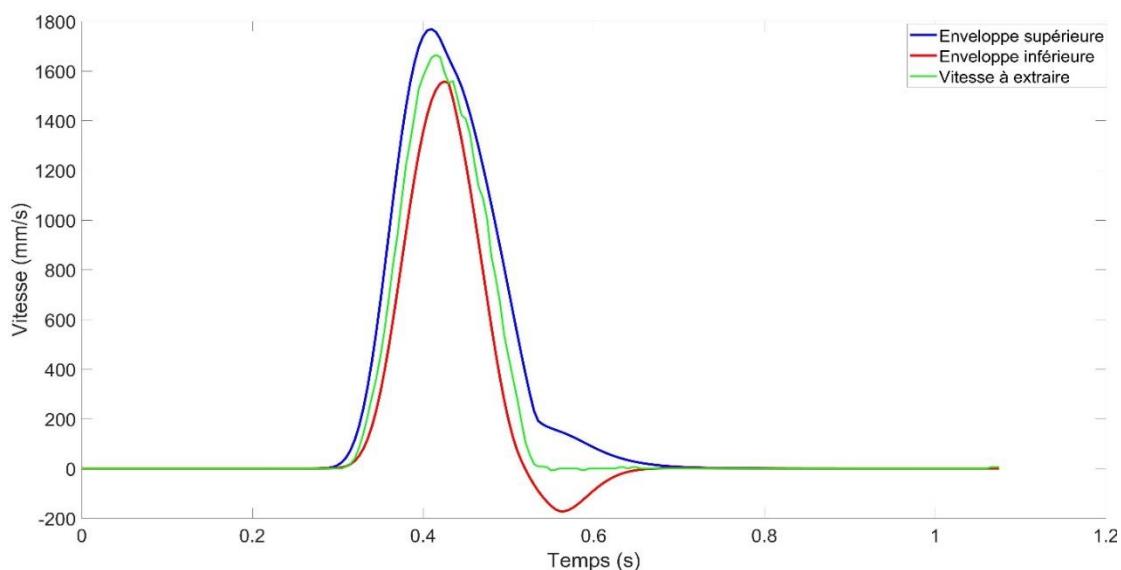


Figure 7.3 Boite englobante obtenue avec l'algorithme S&E pour SNR fixé 28dB

7.4 Discussion relative au modèle Delta-Lognormal de la théorie cinématique du mouvement rapide et l'extracteur S&E

Concernant l'analyse du mouvement avec le modèle Delta-Lognormale, l'algorithme **S&E** apporte des éléments de réponses au mauvais conditionnement de la fonction Delta-Lognormale. Il enlève toute ambiguïté sur la valeur de la solution finale en fournissant le plus petit vecteur d'intervalles dans lequel la solution existe. La tâche finale revient à l'utilisateur d'en faire l'interprétation et l'usage adéquats. D'autre part, on pourrait qualifier l'algorithme d'effectuer du surajustement si on prend pour acquis que les signaux des tablettes sont distortionnés. Ce n'est pas le cas, car l'algorithme ne résout pas le mauvais conditionnement des paramètres du modèle Delta-Lognormale, mais il fournit un intervalle réduit pour chaque paramètre dans lequel des interprétations au regard de l'application peuvent être faites. L'amélioration majeure au regard des objectifs est le court délai d'exécution sur ordinateur conventionnel requis pour avoir une solution avec n'importe quel profil de vitesse de mouvement manuscrit simple. Cette amélioration a été obtenue en moyen d'assertions reposant sur les propriétés mathématiques dégagées par la fonction Delta-Lognormale.

CHAPITRE 8 CONCLUSION ET RECOMMANDATIONS

Cette thèse fait état du problème d'ajustement des courbes de vitesses de mouvements manuscrits rapides selon le modèle Delta-Lognormal de la théorie cinématique du mouvement rapide. Le problème posé est la nécessité d'un algorithme capable de produire des solutions globales pour l'ajustement lognormal des profils de vitesses de mouvements manuscrits simples pour des applications biomédicales. La solution proposée est un algorithme de recherche de solutions exactes qui utilise la méthode de séparation et évaluation (**S&E**). La réalisation de cet algorithme s'est repartie sur cinq objectifs principaux :

1. Collecter des mouvements humains rapides chez des individus sains et malades pour entraîner l'extracteur.
2. Créer des mouvements rapides de pointages manuscrits synthétiques avec table de vérité. Ces mouvements devraient être semblables aux mouvements de pointages manuscrits collectés sur des personnes saines et malades.
3. Proposer un estimateur global des paramètres Delta-lognormaux en utilisant l'algorithme de séparation et évaluation (**S&E**) communément appelé (Branch and Bound).
4. Établir des contraintes à cet extracteur global pour obtenir des délais d'extraction meilleurs que ceux obtenus dans [50] pour le problème considéré.
5. Développer un banc de test pour validation.

Chacun de ces objectifs a fait l'objet de développements novateurs qui enrichissent l'état de l'art sur l'extraction des paramètres lognormaux.

Concernant le premier objectif, un nouvel outil peu coûteux permettant de recueillir des mouvements rapides manuscrits a été élaboré. Le dispositif intègre dans des tablettes numérisées un protocole de test neuromoteur qui est utilisé dans la capture de mouvements de pointages rapides. Les mouvements enregistrés sont automatiquement sauvegardés dans une base de données pour étude. Le dispositif de collecte conçu (Lognomètre) est en exploitation dans divers lieux (école, centre sportif, clinique) pour permettre la capture de mouvements rapides et effectuer des recherches sur tout aspect physiologique lié aux mouvements de pointages manuscrits. Le lognomètre conçu fait l'objet de deux contributions novatrices.

La première contribution est l'ensemble des réalisations suivantes :

- L'intégration de l'ensemble de tests neuromoteurs nécessaires pour recueillir des mouvements de pointage humain dans un système de capture de mouvement utilisant une tablette et un stylo numérique.
- L'élaboration et le respect des contraintes d'acquisition des échantillons de mouvements pour un test neuromoteur à une fréquence de 200HZ et une résolution de 100 points par millimètre.
- L'insertion d'une base de données conçue en langage C++ pour conserver les mouvements de pointage captés par la tablette selon différentes contraintes associées aux types de mouvement et au nom du projet pour lequel sont recueillies les données.

Dont la publication s'intitule : « **A New Experimental Set-up to Run Neuromuscular Tests** » publiée dans la conférence : « International Conference on pattern Recognition and artificial intelligence: Workshop on the Lognormality Principles and its Applications » (**ICPRAI**) Montréal 2018, met en relief l'implémentation de la batterie de tests neuromoteurs proposée par le personnel en clinique et les ingénieurs.

La deuxième contribution est l'intégration complète de la batterie des neuf tests neuromoteurs dans la tablette numérique, la transformation en logiciel de l'outil matériel externe permettant de réaliser certains tests neuromoteurs nécessitant un stimulus visuel, l'intégration de la version logicielle du stimulus visuel dans la tablette, et la mise à niveau de l'ensemble du système pour une utilisation dans un système d'exploitation moderne (Window 10).

Cette contribution a conduit à l'article: « **An Interactive Tablet-based System to Run Neuromuscular Tests** » publié dans le livre « The Lognormality Principle and its applications in E-Security, E-learning and E-Health ». Cet article poursuit dans la même veine d'idées que la publication de conférence ci-dessus en mettant de l'avant un système d'acquisition complet et autonome. Une copie conforme à l'originale pour chacune des publications est présentée en annexe D et E de cette thèse respectivement. Au-delà de l'acquisition des mouvements manuscrits avec le lognomètre conçu, une base de données permettant l'entreposage de ces mouvements a été construite. Cet objectif est pleinement atteint au regard de l'utilisation du dispositif dans divers milieux d'études.

Dans le cas du deuxième objectif, deux algorithmes permettant la génération automatique de mouvements synthétiques pour les différentes classes établies à partir des observations faites sur les profils manuscrits des personnes saine et malade et les propriétés de la fonction Delta-Lognormale de la théorie cinématique du mouvement rapide ont été produits. Ces Mouvements synthétiques permettent d'entrainer tout algorithme utilisant un modèle paramétrique pour approximer des mouvements manuscrits. Le grand avantage de ces mouvements synthétiques est qu'ils possèdent des valeurs de paramètres connus pour chaque mouvement et permettent ainsi de valider la précision sur valeur de la solution obtenue. Une banque de plus de dix mille courbes Delta-Lognormales synthétiques a été conçue pour entraîner l'extracteur **S&E** élaboré. L'objectif de créer des données synthétiques avec pour but d'entrainer l'algorithme **S&E** est accomplis.

Le troisième objectif visant à proposer un estimateur global des paramètres Delta-lognormaux en utilisant l'algorithme de séparation et évaluation (**S&E**) a été lui aussi complété avec succès. En effet, un algorithme original de type **S&E** permettant l'extraction des paramètres Delta-Lognormaux de profils de vitesses de mouvement manuscrit a été conçu et implémenté en langage de haut niveau sur un ordinateur conventionnel. L'algorithme produit une boîte solution unique garantissant l'inclusion de la solution optimale pour tous mouvements simples manuscrits. La réalisation de cet objectif fait l'objet de **quatre contributions principales novatrices dans l'état sur l'extraction des paramètres Delta-Lognormaux des mouvements manuscrits.**

La première contribution est la conception de la borne minimale et de la borne maximale pour la fonction Delta-Lognormale en utilisant l'arithmétique par intervalles. Cette contribution nécessaire dans un algorithme de type **S&E** fait l'objet de la publication scientifique: « **On the use of Interval Arithmetic to Bound Delta- Lognormal Rapid Human Movements Models dans International Conference on pattern Recognition and artificial intelligence: Workshop on the Lognormality Principles and its Applications (ICPRAI) Montréal 2018.** » détaillée en annexe A de cette thèse. Elle met en lumière le concept d'enveloppe et sa boîte englobante associée en utilisant l'arithmétique par intervalles et son théorème fondamental. La procédure permettant de concevoir une enveloppe Delta-Lognormale et sa boîte englobante est proposée. La détermination de la borne minimale et maximale de l'enveloppe Delta-Lognormale avec un coût en calcul moindre (en comparaison avec [50]) et une garantie d'inclusion du profil de vitesses du mouvement à étudier dans les bornes est détaillée. La procédure utilise une approche de calcul par intervalle novatrice jamais utilisée dans l'état de l'art. Dans de nombreux problèmes résolus avec la méthode

de **S&E**, cette étape de conception de l'enveloppe, garantissant les bornes minimales et maximales inclusives du problème à résoudre est considérée comme problème non complet du fait de sa complexité. Des tests et résultats montrant la validité du procédé sur des profils de vitesses de mouvements manuscrits synthétiques sont présentés et illustrés dans la publication.

La deuxième contribution est la conception de l'enveloppe Delta-Lognormale pour quatre classes de profils de vitesses synthétiques et son utilisation dans un algorithme de type **S&E**. Cette contribution fait l'objet de la publication intitulée « **On the Use of Interval Arithmetic for the Branch and Bound Delta-Lognormal Parameter Extraction of Rapid Human Movements** » dans un chapitre du livre « **The Lognormality Principle and its applications in E-Security, E-learning and E-Health** ». Elle discute des techniques de boites englobantes et montre une utilisation dans un algorithme général de type **S&E** sur une gamme de six classes de profils de vitesses de mouvements manuscrits synthétiques. Elle est détaillée en annexe B de cette thèse.

La troisième contribution est l'utilisation du gradient de la fonction de coût (erreur quadratique moyenne entre données observées et fonction Delta-Lognormale) pour choisir la dimension de coupure d'une enveloppe englobante en sous-enveloppes et permettre une extraction de paramètres Delta-Lognormaux sur neuf classes différentes de profils de vitesses synthétiques dans un algorithme de type **S&E**. Cette contribution fait l'objet de la publication « **Combining Interval Arithmetic with the Branch and Bound Algorithm for Delta-lognormal Parameter Extraction.** » dans International Conference on pattern Recognition and artificial intelligence: Workshop on the Lognormality; Cancún Mexico 2019 ». Elle poursuit dans la même veine que la deuxième contribution en enrichissant les procédures de conceptions et tests de l'enveloppe Delta-Lognormale et sa boite englobante associée. Les tests se font sur des profils de vitesses synthétiques et réels. Elle illustre l'application de l'enveloppe construite dans un algorithme de **S&E** simple avec une procédure de division de boites.

La Quatrième contribution est l'ensemble des développements suivant :

- ✓ La définition de l'optimum global pour le problème d'extraction des paramètres Delta-Lognormaux.
- ✓ L'élaboration et validation de 5 différentes règles permettant l'évaluation des solutions candidates.

- ✓ L'élaboration automatique (algorithme) des données synthétiques pour toutes les classes (9) de profils de vitesses de mouvements rapides simples connus à ce jour.
- ✓ Des tests avec résultat concluant dans un algorithme de types **S&E** sur l'ensemble des données d'écritures synthétiques de toutes les classes de courbes de vitesses synthétiques.
- ✓ Les résultats d'un ensemble de tests de mouvements de pointages humains imposés et extrait avec succès, dans des délais raisonnables par l'algorithme **S&E**.

Cette contribution fait l'objet de la publication « **A Global Optimization Algorithm for the Estimation of Delta-Lognormal Parameters Characterizing Pointing Gestures** ». Elle est soumise dans la revue « **Pattern Analysis and Machine Intelligence** » (PAMI) et est présentement en cours d'évaluation par l'éditeur. On y propose un algorithme de **S&E** conçu de toute pièce. Bien qu'il reste encore plusieurs étapes avant d'arriver à un prototype utilisable en milieu clinique, l'élaboration de l'algorithme **S&E** constitue un avancement majeur vers l'atteinte de cet objectif finale. Les autres contributions substantielles dans ce sens à savoir : l'utilisation de la cinématique du mouvement humain dans différentes applications, l'accroissement du temps d'exécution de l'algorithme, conduiront à coup sûr vers l'objectif final.

Le quatrième objectif consiste à apporter des solutions novatrices aux problèmes de délais d'exécution que nécessite ce type d'algorithme par énumération combinatoire. Quatre critères de rejets des solutions non prometteuses (voir chapitre 7, discards tests) tous indépendants les uns des autres et utilisant des concepts mathématiques nouveaux dans l'état de l'art ont été proposés et implémentés. Ils favorisent l'obtention des solutions optimales dans un délai de l'ordre de dizaines de minutes sur ordinateur conventionnel, ce qui est une avancée très prometteuse pour l'algorithme **S&E** vers du temps réel. Ce passage d'un algorithme s'exécutant pendant des jours voir des semaines sur des serveurs parallèles tels que proposé par [50], à un algorithme s'exécutant sur ordinateur conventionnel (Intel core i7) en moins de 15 mn pour une extraction d'un profil de vitesse, classe l'algorithme **S&E** comme objectif accompli.

Finalement en réponse au cinquième objectif, un algorithme intégrant la lecture automatique de mouvement manuscrit d'une base de données fournie est opérationnel et permet de tester l'extracteur **S&E** conçu pour tout type de mouvement de pointages manuscrits simples. Le système est fonctionnel uniquement sur le système d'exploitation Windows, mais peut être adapté

moyennant quelques modifications à d'autres systèmes. Tout ordinateur conventionnel peut utiliser le système. Cet objectif peut être aussi considéré comme accompli.

Les objectifs fixés et accomplis avaient donné lieu à la formulation de deux hypothèses qui nécessitent d'être validé.

La première hypothèse énoncée : « *Il est possible de créer des profils de vitesse de mouvements synthétiques simples, semblables aux profils de vitesse de mouvement de pointages manuscrits collecter à partir des tablettes numériques et d'en extraire leurs paramètres Delta-Lognormaux à l'aide d'un algorithme de type S&E.* » est relative à la conception des profils de vitesses de mouvements synthétiques à partir des mouvements manuscrits simples et leur extraction par la fonction Delta-Lognormale dans un algorithme de type **S&E**. Pour cette hypothèse, deux algorithmes permettant la génération automatique de mouvements synthétiques pour les différentes classes établis à partir des observations faites sur les profils manuscrits des personnes saine et malade et les propriétés de la fonction Delta-Lognormale de la théorie cinématique du mouvement rapide ont été produits. Neuf mille profils de vitesses en raison de mille par classes ont été entreposés dans notre base de données (voir chapitre 6). Ces mouvements synthétiques ont été extraits par l'extracteur **S&E** conçus tels que présenté dans les publications « **Combining Interval Arithmetic with the Branch and Bound Algorithm for Delta-Lognormal Parameter Extraction.** » et « **A Global Optimization Algorithm for the Estimation of Delta-Lognormal Parameters Characterizing Pointing Gestures** ». Cette dernière est soumise dans la revue « **Pattern Analysis and Machine Intelligence** » détaillée en annexe C et au chapitre 5 respectivement. Cette hypothèse est vérifiée et validée expérimentalement à travers les tests et résultats présentés dans les deux publications ci-dessus et les développements proposés au chapitre 6.

La deuxième hypothèse énoncée « il est possible d'obtenir automatiquement un intervalle de confiance sur la combinaison des paramètres approximant un profil de vitesse de mouvement simple de pointage manuscrit avec l'algorithme **S&E**. Les intervalles de confiances sur chaque paramètre ou vecteur d'intervalles (boîte) produisent une enveloppe Delta-Lognormale unique qui contient la totalité du profil de vitesse de la courbe à approximer » fait état du caractère unique du vecteur d'intervalles des paramètres Delta-Lognormaux représentatifs d'un mouvement manuscrit simple quelconque. Cette hypothèse novatrice est validée par l'ensemble des contributions de la

publication « **A Global Optimization Algorithm for the Estimation of Delta-Lognormal Parameters Characterizing Pointing Gestures** ». Elle est soumise dans la revue « **Pattern Analysis and Machine Intelligence** » détaillée au chapitre 5. En effet, l'algorithme **S&E** retourne toujours une et une seule boite comme solution finale. Les valeurs pouvant être aberrantes sont exclues dès le départ dans l'initialisation de l'algorithme. La règle initiale (initiale rule) dans **S&E** conçoit une enveloppe renfermant la totalité de du profil à approximer selon les valeurs limites de la fonction Delta-Lognormale et du matérielle d'acquisitions des mouvements.

Notons que les contributions proposées aux annexes D et E relatives à la conception des logiciels développés pour l'acquisition de mouvements manuscrits à extraire les paramètres Delta-Lognormaux par l'algorithme **S&E** sont considérées comme secondaires dans cette thèse, car certains développements techniques ne sont pas rattachés directement aux objectifs de la thèse.

Le nouvel algorithme de type **S&E** conçus, en plus de rendre possible des études pour des applications où une solution locale n'est plus suffisante, ouvre la voie à plusieurs opportunités. Il est maintenant possible pour n'importe quel mouvement rapide de pointage d'obtenir les sept paramètres Delta-Lognormaux dans un intervalle de confiance garantissant l'inclusion de la solution optimale. Cet avantage est d'autant plus significatif pour des modélisations utilisant la notion de sous mouvement [79] avec chaque pic de vitesse dans un mouvement considéré comme un sous mouvement. Une telle modélisation est d'ailleurs étudiée dans [17] et il ressort plusieurs ambiguïtés dans la comparaison entre les paramètres des différents mouvements. La modélisation en **S&E** de la fonction Delta-Lognormale permet de lever ces ambiguïtés en regroupant l'essentiel de l'information du mouvement dans un vecteur d'intervalles de sept paramètres globalement optimaux.

Observons cependant que les éléments de méthodologies développés dans ce chapitre s'appliquent à d'autres problèmes d'extractions de paramètres par régression non linéaire. Bien que les développements présentés au chapitre 5 ainsi que les annexes A B C D et E de cette thèse montrent des résultats satisfaisants, on note néanmoins des temps d'exécution moyens d'environ 10 à 12 minutes par profil d'essai. Cependant, l'algorithme présente certaines caractéristiques très intéressantes telles que:

- Un parallélisme inhérent à plusieurs niveaux, notamment dans le calcul des caractéristiques (coût, gradient, coût réel, etc.) Reliée à une boite candidate, l'application des propriétés de

rejet qui sont complètement indépendantes pourrait être valorisée dans un processeur numérique adapté (**FPGA, GPU**).

- L'emboîtement des différentes opérations principales peut être factorisé et regroupé en un seul bloc évitant ainsi la duplication d'opérations.
- L'utilisation d'optimiseur locaux ou d'algorithme plus sophistiqué avec un coût en calcul modéré dans certains passages de l'algorithme notamment lors de la division d'une boîte en sous-espace. À cette étape, on pourrait tenter une réduction de la largeur de la boîte avant d'effectuer la division si la nécessité y est encore de mise.

Ces caractéristiques bien outillées et exploitées adéquatement ouvrent la porte au développement futur d'un meilleur l'algorithme.

Perspectives

Le travail sur la conception de l'extracteur S&E ouvrent la portent à plusieurs perspectives. En effet, les mouvements de pointages synthétiques peuvent maintenant être caractérisés pour chacune des formes qu'elles représentent, car l'algorithme S&E conçu, fournit en sortie un intervalle de confiance qui regroupe l'ensemble des solutions possibles (Delta-Lognormaux) pour un mouvement. Cette avancée majeure va contribuer à construire de nouveaux mouvements synthétiques plus spécialisés (chaque classe appartient à un intervalle de confiance connu) pour des systèmes d'extractions plus robustes. Sur le plan algorithmique, l'algorithme S&E conçu peut être adapté pour extraire les paramètres des mouvements de pointages complexes tels des signatures en utilisant le modèle vectoriel Sigma-Lognormale de la théorie cinématique du mouvement rapide. Les modifications dans ce sens nécessitent la détermination des intervalles de chaque position angulaire de tous les sous-mouvements contenus dans un mouvement complexe. La bibliothèque en arithmétique par intervalles construite dans le cadre de ce projet, regroupe déjà les implémentations par intervalles des opérations sur les fonctions trigonométriques de bases contenues dans l'équation caractéristique du modèle Sigma-Lognormale. Il ne reste plus qu'à les adapter adéquatement aux équations caractéristiques du modèle vectoriel selon le mouvement complexe à extraire.

Un autre axe de recherche intéressant est l'accélération de l'exécution de l'algorithme conçu en tirant profit des corrélations importantes entre les valeurs des paramètres Delta-Lognormaux. Par

exemple, il est possible de réduire le nombre de dimensions du problème de 7 à 6 en fixant t_0 au temps de réaction minimale d'un individue. Davantage d'investigations sur la fonction Delta-Lognormale de la théorie cinématique du mouvement rapide sont nécessaires pour mieux saisir ses caractéristiques et procurer plus de contraintes à l'algorithme proposé. Notons que pour le travail présenté dans cette thèse, le parallélisme naturel de l'algorithme a été partiellement exploité dans un processeur graphique afin de tirer de la puissance de calcul de cet outil à exécution parallèle. Des études approfondies dans ce sens conduiront à coût sûr vers un algorithme d'extraction dans des délais meilleurs et permettront une plus grande utilisation de l'extracteur.

Finalement, davantage de recherches sur l'utilisation de l'algorithme S&E seront nécessaires pour mieux comprendre le problème de mauvais conditionnement de l'extraction des paramètres Delta-Lognormaux sur des mouvements rapides. Cet axe de recherche important permettrait d'identifier un ensemble de règles de rejet de mouvements ne nécessitant pas l'utilisation de l'extracteur S&E il sera alors important de mesurer l'impact du rejet de cette classe de mouvements sur les phénomènes étudiés.

RÉFÉRENCES

- [1] R. Plamondon, G. Pirlo, E. Anquetil et al, «Personal digital bodyguards for e-security, e-learning and e-health: A prospective survey,» *Pattern Recognition*, vol. 81, pp. 633-659, 2018.
- [2] S. Rosenblum et L.-Z. Miri, «Handwriting process and product characteristics of children diagnosed with developmental coordination disorder,» *Human movement science*, vol. 27, n° 12, pp. 200-214, 2008.
- [3] S.-H. Chang et Y. Nan-Ying, «Evaluation and classification of types of Chinese handwriting deficits in elementary schoolchildren,» *Perceptual and Motor Skills* , vol. 101, n° 12, pp. 631-647, 2005.
- [4] T. Schenk et al, «Does training improve writer's cramp? An evaluation of a behavioral treatment approach using kinematic analysis,» *Journal of hand therapy*, vol. 17, n° 13, pp. 349-363, 2004.
- [5] A. Schroter, R. Mergl, K. Burger, H. Hampel, H.-J. Moller and U. Hergerl, "Kinematic Analysis of Handwriting Movements in Patients with Alzheimer's Disease, Mild Cognitive Impairment, Depression and Healthy Subjects," *Dementia and Geriatric Cognitive Disorders*, vol. 15, no. 3, pp. 132-142, 2003.
- [6] H.-L. Teulings et S. George E., «Control of stroke size, peak acceleration, and stroke duration in Parkinsonian handwriting,» *Human Movement Science*, vol. 10, n° 12-3, pp. 315-334, 1991.
- [7] M. G. Longstaff, R. A. et Heath, «Spiral drawing performance as an indicator of fine motor function in people with multiple sclerosis,» *Human movement science*, vol. 25, n° 14-5, pp. 474-491, 2006.

- [8] A. Alimi et R. Plamondon, «A comparative study of speed/accuracy tradeoff formulations: the case of spatially constrained movements where both distance and spatial precision are specified,» chez Handwriting and Drawing Research, IOS Press, Burke, 1996, pp. 127-142.
- [9] F. Lacquaniti, C. Terzuolo et P. Viviani, «The law relating the kinematic and figural aspects of drawing movements. *Acta psychologica*,» vol. 54, n° 11-3, pp. 115-130, 1983.
- [10] P. Morasso, «Spatial control of arm movements. *Experimental brain research*,» vol. 42, n° 12, pp. 223-227, 1981.
- [11] J. M. Hollerbach, «Computers, brains and the control of movement,» *Trends in Neurosciences*, vol. 5, pp. 189-192, 1982.
- [12] R. Plamondon, "A kinematic theory of rapid human movements: Part I: Movement representation and generation," *Biological cybernetics*, vol. 72, pp. 295-307, 1995.
- [13] R. Plamondon, «A kinematic theory of rapid human movements: Part III. Kinetic outcomes,» *Biological Cybernetics*, vol. 78, n° 12, pp. 133-145, 1998.
- [14] R. Plamondon, C. Feng and A. Woch, "A kinematic theory of rapid human movement. Part IV: a formal mathematical proof and new insights," *Biological Cybernetics*, vol. 89, no. 2, pp. 126-138, 2003.
- [15] H.-L. Teulings, J. L. Contreras-Vidal, G. E. Stelmach and A. C.H, "Parkinsonism reduces coordination of fingers, wrist, and arm in fine motor control," *Experimental neurology*, vol. 146, no. 1, pp. 159-170, 1997.
- [16] A. W. Van Gemmert, H.-L. Teulings et G. E. Stelmach, «The influence of mental and motor load on handwriting movements in Parkinsonian patients,» *Acta psychologica*, vol. 100, n° 11-2, pp. 161-175, 1998.

- [17] C. O'Reilly, «Développement d'outils d'analyse de la motricité fine pour l'investigation de troubles neuromusculaires: Théorie, prototype et mise en application dans le contexte des accidents vasculaire cérébraux,» Polytechnique Montréal, Montréal, 2012.
- [18] P. Laniel, N. Faci, R. Plamondon, M. H. Beauchamp et B. Gauthier, «Kinematic analysis of fast pen strokes in children with ADHD using the Sigma-lognormal model,» International Conference on Pattern Recognition and Artificial Intelligence (ICPRAI), 2018.
- [19] R. Plamondon, «A kinematic theory of rapid human movements Part II. Movement time and control,» *Biological Cybernetics*, vol. 74, n° 14, pp. 309-320, 1995.
- [20] R. Plamondon, "A kinematic theory of rapid human movements: Part III. Kinetic outcomes," *Biological Cybernetics*, vol. 78, no. 2, pp. 133-145, 1998.
- [21] C. O'Reilly and R. Plamondon, "Development of a Sigma–Lognormal representation for on-line signatures," *Pattern Recognition*, vol. 42, no. 12, pp. 3324-3337, 2009.
- [22] M. Djouia et R. Plamondon, «A new algorithm and system for the characterization of handwriting strokes with delta-lognormal parameters,» *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, n° 111, pp. 2060-2072, 2009.
- [23] W. Guerfali et R. Plamondon, «A new method for the analysis of simple and complex planar rapid movements,» *Journal of neuroscience methods*, vol. 82, n° 11, pp. 35-45, 1998.
- [24] W. H. Press, umerical recipes 3rd edition: The art of scientific computing, Cambridge university press, 2007.
- [25] A. Alimi et R. Plamondon, «Analysis of the parameter dependence of handwriting generation models on movement characteristics,» chez Advances in Handwriting and Drawing: a multidisciplinary approach, 1994, pp. 363-378.
- [26] R. Plamondon, A. Alimi, P. Yergeau et al, «Modelling velocity profiles of rapid movements: a comparative study,» *Biological cybernetics*, vol. 69, n° %12, pp. 119-128, 1993.

- [27] A. J. Vander, J. H. Sherman, D. Luciano et al, *Human Physiology*, vol. 2, Boston: McGraw-Hill, 1989.
- [28] I. S. MacKenzie, «Fitts' law as a research and design tool in human-computer interaction,» *Human-computer interaction*, vol. 7, n° 11, pp. 91-139, 1992a.
- [29] R. Plamondon et A. Alimi, «Speed/Accuracy tradeoffs in target-directed,» *Behavioral and Brain Sciences*, vol. 20, n° 12, pp. 279-349, 1997.
- [30] T. Kvalseth, «An alternative to Fitts law,» vol. 16, n° 15, pp. 371-373, 1980.
- [31] S. A. Jax, D. A. Rosenbaum et J. Vaughan, «Extending Fitts Law to manual obstacle avoidance,» *Experimental brain research*, vol. 180, n° 14, pp. 775-779, 2007.
- [32] J. Vaughan, D. A. Barany, A. W. Sali et al, «Extending Fitts' Law to three-dimensional obstacle-avoidance movements: support for the posture-based motion planning model,» *Experimental brain research*, vol. 207, n° 11-2, pp. 133-138, 2010.
- [33] J. Wu, J. Yang et T. Honda, «Fitts' law holds for pointing movements under conditions of restricted visual feedback,» *Human movement science*, vol. 29, n° 16, pp. 882-892, 2010.
- [34] M. Bohan, M. G. Longstaff, A. W. Van Gemmert et al, «Differential effects of target height and width on 2D pointing movement duration and kinematics,» *Motor Control*, vol. 7, n° 13, pp. 278-289, 2003.
- [35] M. G. Longstaff et M. Isaac, «The Ponzo Illusion affects movement characteristics in memory guided target aiming movements,» 2011.
- [36] J. Mendoza, S. Hansen, C. M. Glazebrook et al, «Visual illusions affect both movement planning and on-line control: a multiple cue position on bias and goal-directed action,» *Human movement science*, vol. 24, n° 15-6, pp. 760-773, 2005.

- [37] P. Van Donkelaar, «Pointing movements are affected by size-contrast illusions,» *Experimental Brain Research*, vol. 125, n° 14, pp. 517-520, 1999.
- [38] T.-T. Chiu, K.-Y. Young, S.-H. Hsu et al., «A study of Fitts' law on goal-directed aiming task with moving targets,» *Perceptual and motor skills*, vol. 113, n° 11, pp. 339-352, 2011.
- [39] E. Brenner et J. B. Smeets, «Quickly 'learning' to move optimally,» *Experimental brain research*, vol. 213, n° 11, pp. 153-161, 2011.
- [40] J. S. Augustyn et D. A. Rosenbaum, «Metacognitive control of action: Preparation for aiming reflects knowledge of Fitts's law,» *Psychonomic bulletin & review*, vol. 12, n° 15, pp. 911-916, 2005.
- [41] C. O'Reilly et R. Plamondon, «Impact of the principal stroke risk factors on human movements,» *Human movement science*, vol. 30, n° 14, pp. 792-806, 2011a.
- [42] B. Rohrer, S. Fasoli, H. I. Krebs et al, «Submovements grow larger, fewer, and more blended during stroke recovery,» *Motor control*, vol. 8, n° 14, pp. 472-483, 2004.
- [43] C. O'Reilly, R. Plamondon, L.-H. Lebreun, B. Clément et P. Mathieu, «Sigma-lognormal analysis of a complex movements neuromuscular test,» chez 14th conference of the international graphonomics society, Dijon, France, 2009.
- [44] D. W. Marquardt, «An algorithm for least-squares estimation of nonlinear parameters,» *Journal of the society for Industrial and Applied Mathematics*, vol. 11, n° 12, pp. 431-441, 1963.
- [45] A. C. Cohen et W. Betty Jones, «Estimation in the three-parameter lognormal distribution,» *Journal of the American Statistical Association*, vol. 75, n° 1370, pp. 399-404, 1980.
- [46] K. Shimizu, *Point estimation*, chap. 2 in *lognormal distributions: Theory and applications*, el crow and k. shimizu eds, 1988.

- [47] R. Plamondon, X. Li et M. Djoua, «Extraction of delta-lognormal parameters from handwriting strokes,» *Frontiers of Computer Science in China*, vol. 1, n° 11, pp. 106-113, 2007.
- [48] W. Guerfali et R. Plamondon, «Signal Processing for the Parameter Extraction of the Delta Lognormal Model ($\Delta\Lambda$),» *Research in computer and robot vision*, pp. 217-232, 1995.
- [49] M. Djoua, R. Plamondon, A. Delle Cioppa and al., "Deterministic and evolutionary extraction of delta-lognormal parameters: performance comparison.," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 21, no. 0, 2007.
- [50] C. O'Reilly and R. Plamondon, "A globally optimal estimator for the Delta-Lognormal modeling of fast reaching movements," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 42, no. 5, pp. 1428-1442, 2012.
- [51] M. E. Wise, "The geometry of log-normal and related distributions and an application to tracer-dilution curves," *Statistica Neerlandica*, vol. 20, no. 1, pp. 119-142, 1966.
- [52] B. R. Rohrer et N. Hogan, «Avoiding spurious submovement decompositions: a globally optimal algorithm,» *Biological cybernetics*, vol. 89, n° 13, pp. 190-199, 2003.
- [53] M. Djoua et R. Plamondon, «A new algorithm and system for the characterization of handwriting strokes with delta-lognormal parameters,» *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, n° 111, pp. 2060-2072, 2008.
- [54] R. A. Schmidt, T. D. Lee, C. Winstein et al, *Motor control and learning: A behavioral emphasis*, 2018.
- [55] F. Nadir, S. P. Boyogueno Bidias, R. Plamondon et N. Bergeron, ««A new experimental Set-up to Run neuromuscular tests»,» *chez Proceedings of ICPRAI : First International Worksop on the lognormality Principle and its applications*, Montreal, mai 2018.

- [56] R. E. Moore, *Methods and applications of interval analysis*, Society for Industrial and Applied Mathematics, 1979.
- [57] S. P. Boyogueno Bidias, J.-P. David, Y. Savaria and R. Plamondon, "On the use of Interval Arithmetic to Bound Delta-Lognormal Rapid Human Movements," in International Conference on pattern Recognition and artificial intelligence: Workshop on the Lognormality Principles and its Applications, Montreal, 2018.
- [58] R. E. Moore, *Interval analysis*, Englewood Cliffs: Prentice-Hall, 1966.
- [59] E. Hansen et G. W. Walster, *Global optimization using interval analysis: revised and expanded*, CRC Press, 2003.
- [60] C. E. Wright and D. E. Meyer, "Conditions for a linear speed-accuracy trade-off in aimed movements," *The Quarterly Journal of Experimental Psychology*, vol. 35, no. 2, pp. 279-296, 1983.
- [61] R. A. Abrams, D. E. Meyer and S. Kornblum, "Speed and accuracy of saccadic eye movements: characteristics of impulse variability in the oculomotor system," *Journal of Experimental Psychology: Human Perception and Performance*, vol. 15, no. 3, p. 529, 1989.
- [62] P. M. Fitts and J. R. Peterson, "Information capacity of discrete motor responses," *Journal of experimental psychology*, vol. 67, no. 2, p. 103, 1964.
- [63] H. N. Zelaznik, S. Mone, G. P. Mccabe and al, "Role of temporal and spatial precision in determining the nature of the speed-accuracy trade-off in aimed-hand movements," *Journal of Experimental Psychology: Human Perception and Performance*, vol. 14, no. 2, p. 221, 1988.
- [64] C. O'Reilly and R. Plamondon, "Agonistic and antagonistic interaction in speed/accuracy tradeoff: A Delta-Lognormal perspective.,," *Human movement science*, vol. 32, no. 5, pp. 1040-1055, 2013.

- [65] N. Faci, S. P. Boyogueno Bidias, R. Plamondon and N. Bergeron, "A new experimental set-up to run neuromuscular tests," in In Proceedings of the International Conference on Pattern Recognition and Artificial Intelligence, Montreal, 2018.
- [66] C. O'Reilly, R. Plamondon and L.-H. Lebrun, "Linking brain stroke risk factors to human movement features for the development of preventive tools," *Frontiers in Aging Neuroscience*, vol. 6, no. 3, p. 150, 2014.
- [67] S. Djeziri, W. Guerfali, R. Plamondon and al, "Learning handwriting with pen-based systems: computational issues," *Pattern Recognition*, vol. 35, no. 5, pp. 1049-1057, 2002.
- [68] A. Laurent, R. Plamondon and M. Begon, "Central and Peripheral Shoulder Fatigue Pre-screening Using the Sigma-Lognormal Model: A Proof of Concept," *Frontiers in human neuroscience*, vol. 14, p. 171, 2020.
- [69] M. P. Broderick, A. Van Gemmert, H. A. Shill and al, "Hypometria and bradykinesia during drawing movements in individuals with Parkinson's disease," *Experimental brain research*, vol. 197, no. 3, pp. 223-233, 2009.
- [70] G. Pirlo, M. Diaz, M. A. a. Ferrer and al, "Early diagnosis of neurodegenerative diseases by handwritten signature analysis," In : *International Conference on Image Analysis and Processing*. Springer, Cham, pp. 290-297, 2015.
- [71] M. A. Ferrer, M. Diaz, C. Carmona-Duarte and al, "iDeLog: Iterative dual spatial and kinematic extraction of sigma-lognormal parameters," *IEEE transactions on pattern analysis and machine intelligence*, vol. 42, no. 1, pp. 114-125, 2018.
- [72] S. P. Boyogueno Bidias, J.-P. David, Y. Savaria and R. Plamondon, "Combining Interval Arithmetic with the Branch and Bound Algorithm for Delta-lognormal Parameter Extraction.,," in in *International Conference on pattern Recognition and artificial intelligence: Workshop on the Lognormality*, cancun, mexico, 2019.

- [73] A. Fortin, Analyse numérique pour ingénieurs, Presses inter Polytechnique, 2001.
- [74] O. Knüppel, "PROFIL/BIAS—a fast interval library," *Computing*, vol. 53, no. 3, pp. 277-287, 1994.
- [75] A. Woch et R. Plamondon, «Using the framework of the kinematic theory for the definition of a movement primitive.,» *Motor Control*, vol. 8, n° 14, pp. 547-557, 2004.
- [76] R. Plamondon and M. Djouia, "A multi-level representation paradigm for handwriting stroke generation," *Human movement science*, vol. 25, no. 4, pp. 586-607, 2006.
- [77] R. Plamondon et F. J. Maarse, «An evaluation of motor models of handwriting,» *IEEE Transactions on systems, man, and cybernetics*, vol. 19, n° 15, pp. 1060-1072, 1989.
- [78] D. Bullock, S. Grossberg et C. Mannes, «A neural network model for cursive script production,» *Biological Cybernetics*, vol. 70, n° 11, pp. 15-28, 1993.
- [79] L. Fradet, G. Lee et N. et Dounskoia, «Origins of submovements during pointing movements,» *Acta psychologica*, vol. 129, n° 11, pp. 91-100, 2008.
- [80] A. I. Rusu et V. Govindaraju, «On the challenges that handwritten text images pose to computers and new practical applications,» 2005. [En ligne]. Available: <https://spiedigitallibrary.org/conference-proceedings-of-spie/5676/1/on-the-challenges-that-handwritten-text-images-pose-to-computers/10.1117/12.586350.full>. [Accès le 12 11 2017].
- [81] L. Dipietro, H. I. Krebs, S. Fasoli, B. T. Volpe et N. Hogan, «Submovement changes characterize generalization of motor recovery after stroke,» *Cortex*, vol. 45, n° 13, pp. 318-324, 2009.
- [82] R. Plamondon, C. O'Reilly et O.-P. Claudéric, «Strokes against stroke—strokes for strides.,» *Pattern Recognition*, vol. 47, n° 13, pp. 929-944, 2014.

- [83] A. De santis, V. Caggiano, B. Siciliano et e. al., «Anthropic inverse kinematics of robot manipulators in handwriting tasks.,» chez In : 12th Conference of the International Graphonomics Society., 2005.
- [84] M. Simmer, «Advance in forensic document examination,» *J. Forensic Document Examination*, vol. 13, pp. 1-109, 2000.
- [85] M. Fairhurst, M. Erbilek et C. Li, «Enhancing the forensic value of handwriting using emotion prediction,» chez Biometrics and Forensics (IWBF), International Workshop on. IEEE, 2014. .
- [86] V. W. Henderson, «Alexia and agraphia contrasting perspectives of J.-M. charcot and J. hughlings jackson,» *Neurology*, vol. 70, n° 15, pp. 391-400, 2008.
- [87] M. Lorch, «Written language production disorders: historical and recent perspectives,» *Current neurology and neuroscience reports*, vol. 13, n° 18, p. 369, 2013.
- [88] P. Drotár, J. Mekyska, I. Rektorová et al, «Evaluation of handwriting kinematics and pressure for differential diagnosis of Parkinson's disease.,» *Artificial intelligence in Medicine*, vol. 67, pp. 39-46., 2016.
- [89] M. P. CALIGIURI, H.-L. TEULINGS et C. E. e. a. DEAN, «Handwriting movement kinematics for quantifying extrapyramidal side effects in patients treated with atypical antipsychotics.,» *Psychiatry Research*,, vol. 177, n° 11-2, pp. 77-83, 2010.
- [90] O. Tucha, L. Tucha, G. Kaumann et al., «Training of attention functions in children with attention deficit hyperactivity disorder,» *ADHD Attention Deficit and Hyperactivity Disorders*, vol. 3, n° 13, pp. 271-283, 2011.
- [91] R. Plamondon, «A kinematic theory of rapid human movements,» *Biological Cybernetics*, vol. 72, n° 1 4, pp. 295-307, 1995.

- [92] R. Plamondon, L.-d. Yu, G. E. Stelmach et B. Clément, «On the automatic extraction of biomechanical information from handwriting signals,» *IEEE transactions on systems, man, and cybernetics*, vol. 21, n° 11, pp. 90-101, 1991.
- [93] R. Plamondon, L.-d. YU, G. E. Stelmach et B. Clément, «On the automatic extraction of biomechanical information from handwriting signals,» *IEEE transactions on systems, man, and cybernetics*, vol. 21, n° 11, pp. 90-101, 1991.
- [94] A. Jain, R. Bansal and A. Kumar, "A comparative study of visual and auditory reaction times on the basis of gender and physical activity levels of medical first year students," *International Journal of Applied and Basic Medical Research*, vol. 5, no. 2, p. 124, 2015.
- [95] J. Shelton and G. P. Kumar, "Comparison between auditory and visual simple reaction times," *euroscience & Medicine*, vol. 1, no. 1, p. 30, 2012.
- [96] A. W. NG and A. H. Chan, "Finger response times to visual, auditory and tactile modality stimuli," in *Proceedings of the International MultiConference of Engineers and Computer Scientists*, Hong Kong, 2012.
- [97] T. Duval, C. Rémi, L. Prevost, A. Dorville et R. Plamondon, «Etude de la faisabilité de l'évaluation de l'efficacité des mouvements de tracé du jeune apprenti-scripteur,» chez IHM'14, 26e conférence francophone sur l'Interaction Homme-Machine, Lile, France, 2014.
- [98] U. Bhattacharya, R. Plamondon, S. D. Chowdhury, P. Goyal et S. K. Parui, «A sigma-lognormal model-based approach to generating large synthetic online handwriting sample databases,» *International Journal on Document Analysis and Recognition (IJDAR)*, pp. 1-17, 2017.
- [99] C. O'Reilly, R. Plamondon, M. K. Landou et B. Stemmer, «Using kinematic analysis of movement to predict the time occurrence of an evoked potential associated with a motor command,» *European Journal of Neuroscience*, vol. 32, n° 12, pp. 173-180, 2013.

- [100] R. Plamondon, C. O'Reilly, C. Rémy and T. Duval, "The lognormal handwriter: learning, performing, and declining," *Frontiers in psychology*, vol. 4, 2013.
- [101] D. Impedovo et G. Pirlo, «Dynamic handwriting analysis for the assessment of neurodegenerative diseases: a pattern recognition perspective,» *IEEE Reviews in Biomedical Engineering*, 2018.
- [102] F. Costa, S. Marino et A. Accardo, «Kinematic Analysis of Handwriting in Parkinson Disease,» In : IGS2017: 18th Conference of the International Graphonomics Society, pp. 135-138, 2017.
- [103] T. E. Eichhorn, T. Gasser, N. Mai, C. Marquardt, G. Arnold, J. Schwarz et W. H. Oertel, «Computational analysis of open loop handwriting movements in Parkinson's disease: a rapid method to detect dopaminergic effects,» *Movement disorders*, vol. 11, n° 13, p. 289, 1996.
- [104] J. G. Phillips, G. E. Stelmach et N. Teasdale, «What can indices of handwriting quality tell us about Parkinsonian handwriting?,» *Human Movement Science*, vol. 10, n° 12-3, pp. 301-314, 1991.
- [105] M. L. Kaiser, M. M. Schoemaker, J. M. Albaret et R. H. Geuze, «What is the evidence of impaired motor skills and motor control among children with attention deficit hyperactivity disorder (ADHD)? Systematic review of the literature,» *Research in developmental disabilities*, vol. 25, pp. 338-357, 2015.
- [106] S. Graham, E. J. Fishman, R. Reid et M. Hebert, «Writing Characteristics of Students with Attention Deficit Hyperactive Disorder: A Meta-Analysis,» *Learning Disabilities Research & Practice*, vol. 31, n° 12, pp. 75-89, 2016.
- [107] J. B. Goulardins, J. C. Marques et J. A. De Oliveira, «Attention deficit hyperactivity disorder and motor impairment: a critical review,» *Perceptual and motor skills*, vol. 124, n° 12, pp. 425-440, 2017.

- [108] P. Laniel, N. Faci, R. Plamondon, M. H. Beauchamp et B. Gauthier, «Kinematic analysis of fast pen strokes in children with ADHD,» *Applied Neuropsychology: Child*, pp. 1-16, 2019.
- [109] R. Plamondon, "A kinematic theory of rapid human movements: Part II. Movement time and control," *Biological Cybernetics*, vol. 72, pp. 309-320, 1995.
- [110] M. Djioua et R. Plamondon, «A new algorithm and system for the extraction of delta-lognormal parameters,» 2008.
- [111] C. O'Reilly, R. Plamondon, L.-H. Lebrun, B. Clément et P. Mathieu, «Sigma-lognormal analysis of a complex movements neuromuscular test,» chez 14th conference of the international graphonomics society, Dijon, France, 2009.
- [112] R. Plamondon, C. O'Reilly, C. Rémi and T. Duval, "The lognormal handwriter: learning, performing, and declining," *Frontiers in psychology*, vol. 4, 2013.
- [113] R. Plamondon and M. Djioua, "A multi-level representation paradigm for handwriting stroke generation," *Human movement science*, vol. 25, no. 4, pp. 586-607, 2006.
- [114] N. Hogan, «An organizing principle for a class of voluntary movements,» *Journal of Neuroscience*, vol. 4, n° 111, pp. 2745-2754, 1984.

ANNEXES

ANNEXE A ARTICLE 1: ON THE USED OF INTERVAL ARITHMETIC TO BOUND DELTA-LOGNORMAL RAPID HUMAN MOVEMENTS MODELS

Simon Pierre Boyogueno Bidias <i>Electrical engineering</i> <i>Polytechnique Montréal</i> Montréal, Canada simon- pierre.boyogueno- bidias@polymtl.ca	Jean-Pierre David <i>Electrical engineering</i> <i>Polytechnique Montréal</i> Montréal, Canada JPDavid@polymtl.ca	Yvon Savaria <i>Electrical engineering</i> <i>Polytechnique Montréal</i> Montréal, Canada yvon.savaria@polymtl.ca	Réjean Plamondon <i>Electrical engineering</i> <i>Polytechnique Montréal</i> Montréal, Canada rejean.plamondon@polymtl.ca
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------

This paper is published by (CENPARMI) on May 13 2018, for the International Conference on Pattern Recognition and Artificial Intelligence: (ICPRAI) Montréal 2018.

The paper has been submitted on February 16, 2018.

Abstract. This paper proposes an algorithm based on interval arithmetic to construct envelope models that contain all possible parameter values of a Delta-lognormal function that can match an observed human movement. In the context of the Kinematic Theory of rapid human movements, these parameters represent the basic timing properties of a neuromuscular system, as well as the control strategies used to produce a rapid human movement. The basic idea behind the proposed algorithm is to exploit the isotonicity and monotonicity over an interval to sharply bound the range of the Delta-lognormal function, within a finite space. The new algorithm is described, tested (under various handwritten strokes) and compared with another algorithm previously developed for the same purpose. The numerical results show that the proposed algorithm gives better results in terms of speed and accuracy and provides new tools for the basic study of a single stroke used as a primitive for understanding rapid human movements, according to the Kinematic Theory.

Keywords— *bounding model, monotonicity, isotonicity, interval arithmetic, Delta-lognormal, kinematic theory.*

A. INTRODUCTION

Human movements can be very complex, and their study in terms of motor control has been intensively used in many fields of research. In neurosciences, handwritten strokes constitute a specific class of rapid human movements, notably used to study neurodegenerative processes such

as those involved in Parkinson and Alzheimer's diseases [5], [15]. They have also been used in cybersecurity applications [80], as well as for the rehabilitation of patients having suffered from cerebrovascular accidents [81], [16], [82]. In this context, the need to study the elementary properties of a single stroke for a realistic analysis becomes a key step to understand how the motor control system accomplishes complex movements. The Kinematic Theory is commonly used to model rapid human movements [12], [14], [20]. It presents handwritten strokes as primitives that contain information on both the motor control and the neuromuscular system. The study of such strokes relies on the extraction of parameters that characterize their velocity profile [19], [14].

To meet this need, three deterministic [47], [22], [48] algorithms have been developed in our laboratory in order to extract the lognormal parameters derived from handwriting velocity profiles. These algorithms have successfully led, for example, to applications for recognition of signatures and handwriting. Although the parameter extractor developed in [47], [22], [48] show good data fitting, none of these prior works can guarantee the optimality of the solution obtained. Such a requirement is not always critical for some applications, as excellent curve fitting is sufficient for many patterns recognition tasks. However, globally optimal solutions are important in biomedical applications where a confidence interval on the value of an extracted parameter is necessary in developing a benchmark, or a diagnostic. To address this need, a first branch and bound algorithm was developed in [50].

This algorithm is based on building a tree where each node represents a possible solution of the problem considered. The nodes are Delta-lognormal envelopes, built by studying the variations of the Delta-lognormal function, according to its seven parameters. Each subset or child node is created by subdividing the parent node into three children. The dimension according to which a parent node is subdivided is selected depending on whether the variables that minimize the area of the parent Delta-lognormal envelope. The tree thus constructed is traversed in a way that we prune non-promising solutions based on a monotonicity test. In summary, the branch and bound algorithm proposed in [50] is based on a Delta-lognormal envelope that bounds all the possible Delta-lognormal parameters within a subspace, and then computes a bound on the error between the signal and the nearest envelope. This method, even though it clearly circumscribed the challenge, suffers for lower bound problems (in some cases the computation of the lower bound of the envelope does not hold) and requires very long processing times.

The goal of this paper is to propose a new algorithm to relieve the bounding problem of the extractor proposed in [50]. The proposed method is used to bound the Delta-lognormal parameters representing velocity profile data. Its effectiveness is demonstrated with ideal and experimental velocity profiles.

The rest of the paper is organized as follows. Section 1 reviews the Delta-lognormal equation and the modelling tools that will be used to bound the objective function. Section 2 introduces some theoretical aspects of interval arithmetic and proposes an algorithm for computing enclosures of the bound of the objective function. Section 3 presents numerical results and performance comparisons with [50]. Section 4 discusses the importance of the proposed algorithm, and finally Section 5 proposes possible future research and summarizes the main contributions of this paper.

B. THE DELTA- LOGNORMAL MODEL

The Kinematic Theory of rapid human movements, from which the Delta-lognormal model is derived, considers the trajectory of a pen tip as a single stroke having a delta- lognormal velocity profile.

$$\Delta\Lambda(t; p) = D_1\Lambda(t; t_0, \mu_1, \sigma_1) - D_2\Lambda(t; t_0, \mu_2, \sigma_2) \quad (\text{A.1})$$

Where

$$\Lambda(t; t_0, \mu, \sigma) = \begin{cases} \frac{1}{\sigma\sqrt{2\pi}(t-t_0)} \exp^{-\frac{(\ln(t-t_0)-\mu)^2}{2\sigma^2}}, & \text{for } t_0 < t, \\ 0, & \text{elsewhere.} \end{cases} \quad (\text{A.2})$$

With

$$\sigma > 0; \mu, t_0 \in \mathbb{R}$$

is the lognormal probability density function. In this theory, a stroke is produced by a synergy of two neuromuscular systems: the agonist, acting in the direction of the movement and the antagonist, acting in the opposite direction. The agonist and antagonist are lognormal convolved with their corresponding input commands, represented by the subscript 1 and 2 in equation (A.1) respectively. The subtraction of the two systems responses produces the velocity profile $\Delta\Lambda(t; p)$ described by the seven parameters p shown in equation (A.3).

$$p = [t_0, D_1, \mu_1, \sigma_1, D_2, \mu_2, \sigma_2] \quad (\text{A.3})$$

In (A.3), t_0 represents the system activation time, $\mu_1, \sigma_1, \mu_2, \sigma_2$ correspond to the timing properties (on a logarithmic scale) of the two neuromuscular commands, in reaction to their input amplitudes $D_1 > 0$ and $D_2 > 0$ respectively. To bound the model parameters from velocity data, we need to find the bounding box P that contains all the possible seven parameters p , which fit the original velocity movement (collected from synthetic human movements) most accurately. This problem can be achieved by building an envelope to the nonlinear least square function f (A.4) that maximizes the signal to noise ratio SNR (A.5).

$$f = \int (v_t(t) - \Delta \Lambda(t; p))^2 dt \quad (\text{A.4})$$

$$SNR = 10 \log \left(\frac{\int v_t^2(t) dt}{f} \right) \quad (\text{A.5})$$

C. INTERVAL ARITHMETIC

Before formulating the proposed algorithm, some definitions and notations must be established. Consider the following problem:

Bound $f(x)$ subject to $x \in \mathbf{X}^I$, where f is an n dimensional continuously differentiable function $\mathbb{R}^N \rightarrow \mathbb{R}$ and $\mathbf{X}^I \subseteq \mathbb{R}^N$ is an n dimensional interval vector.

We denote by $I = [a, b] = \{x \in \mathbb{R} : a \leq x \leq b\}$ the set of real numbers and one dimensional closed intervals. In this paper, an interval will be denoted with a variable in upper case and the set of values contained in the interval, in lower case. Thus $X = [\underline{x}, \bar{x}]$ is an interval variable where \underline{x} and \bar{x} represent the left and right end points of X respectively. The width of X is defined and denoted by $w(X) = \bar{x} - \underline{x}$ and the midpoint is given by $m(X) = \frac{1}{2}(\bar{x} + \underline{x})$.

We denote by $\mathbf{X}^I = (X_1, X_2, \dots, X_n)$ the set of n dimensional interval vectors or box such that:

$$\mathbf{X}^I = (X_1, X_2, \dots, X_n) = ([\underline{x}_1, \bar{x}_1], [\underline{x}_2, \bar{x}_2], \dots, [\underline{x}_n, \bar{x}_n]).$$

The width and the midpoint of an interval vector are defined as:

$$w(\mathbf{X}^I) = \max_i w(X_i^I)$$

and

$$m(\mathbf{X}^I) = (m(X_1), m(X_2), \dots, m(X_n)).$$

We call F an inclusion function of $f: \mathbb{R}^N \rightarrow \mathbb{R}$, the interval extension of f , such that for $x \in X \Rightarrow f(x) \subseteq F(X)$. $f(X)$ is the real range of f on X and $f(X) \subseteq F(X)$. Now that we have these definitions and notations, it is possible to use interval arithmetic [58], [59] to bound a function defined by a mathematical expression such as the Delta-lognormal function.

1. INTERVAL ARITHMETIC OPERATIONS

In this section, we summarize some principles of interval arithmetic needed to derive the bounding operation of the objective function stated in (A.4). For more details, the reader can consult [58], [59].

Definition1: the *natural interval extension* of a given function $f(x_1, x_2, \dots, x_n)$ of n variables is given by the interval function $F(X_1, X_2, \dots, X_n)$, which is obtained by replacing the real variable x with the corresponding interval variable X .

Example1: The natural interval extension of

$$f(x, y) = (1 - x)^2 + 100(y - x^2)^2 \quad (\text{A.6})$$

$$g(w, z) = \exp\left(\frac{w+z}{z^2+w}\right) \quad (\text{A.7})$$

are:

$$F(X, Y) = (1 - X)^2 + 100(Y - X^2)^2$$

$$G(W, Z) = \exp\left(\frac{W+Z}{Z^2+W}\right)$$

Where X, Y, W and Z are intervals. If we assign for example $X = [-1, 2], Y = [-1, 3], W = [0, 2], Z = [-1, 1]$ we have:

$$F([-1, 2], [-1, 3]) = (1 - [-1, 2])^2 + 100([-1, 3] - [-1, 2]^2) = [0, 2504] \quad (\text{A.8})$$

$$\begin{aligned} G([0, 2], [-1, 1]) &= \exp\left(\frac{[-1, 3]}{[0, 1]+[1, 1]}\right) \\ &= \exp([-1, 3]) = [\exp(-1), \exp(3)] \end{aligned} \quad (\text{A.9})$$

It is of interest that, in these examples:

In (A.8), if we take any interval, say $X' = [-1, 1] \subseteq X$, the value of $F(X', Y) = [0, 904]$ is contained in $F(X, Y)$. This important property is called inclusion isotonicity.

In the right-hand side of equation (A.9), we use the monotonicity of the exponential function to get the result. This property is also very important for bounding operations and is called inclusion monotonicity.

To this end, the following fundamental theorem of interval arithmetic can be stated:

Fundamental Theorem. Let $F(X_1, X_2, \dots, X_n)$ be the natural interval extension of $f(x_1, x_2, \dots, x_n)$ then

$$f(X_1, X_2, \dots, X_n) \subseteq F(X_1, X_2, \dots, X_n).$$

And for all intervals,

$$Y_k \subset X_k \text{ for } k = 1, \dots, n, f(Y_1, Y_2, \dots, Y_n) \subseteq F(Y_1, Y_2, \dots, Y_n).$$

where

$$f(Y_1, Y_2, \dots, Y_n) = \{f(x_1, x_2, \dots, x_n) : x_k \in Y_k \text{ for } k = 1, \dots, n\}.$$

This theorem due to Moore [58], was extended by Hansen [59]. It states that, the evaluation of a function in any point inside a closed interval where both the function and variable are defined, is guaranteed to produce a result contained in the natural interval extension of the function. Its proof can be found in [59]. This fundamental theorem shows how easy it is to bound the range of a function and makes possible the resolution of the bounding problem stated in Section 1. In the next sub-section, we will apply this theorem to the Delta-lognormal function as a specific sequence of interval arithmetic operations to bound the range of equation (A.4) within an interval vector.

2. ALGORITHM FOR COMPUTING THE UPPER AND LOWER BOUND OF THE DELTA-LOGNORMAL FUNCTION.

To compute a bound of equation (A.4), we first define the bounding space of its variables as:

$$\mathbf{P}^I = \{[\underline{D}_1, \bar{D}_1], [\underline{\mu}_1, \bar{\mu}_1], [\underline{\sigma}_1, \bar{\sigma}_1], [\underline{D}_2, \bar{D}_2], [\underline{\mu}_2, \bar{\mu}_2], [\underline{\sigma}_2, \bar{\sigma}_2], [\underline{t}_0, \bar{t}_0]\} \quad (\text{A.10})$$

Then, through the Fundamental Theorem we apply the natural interval extension on f . This yield:

$$\begin{aligned} f(\mathbf{P}^I) \in F(\mathbf{P}^I) &= [\underline{F}(\mathbf{P}^I), \bar{F}(\mathbf{P}^I)] \\ &= \int (v_t(t) - \Delta\Lambda(t; \mathbf{P}^I))^2 dt \end{aligned} \quad (\text{A.11})$$

As provided by the theorem and the natural interval extension, the result of the right-hand side of equation (A.11) is an interval $[\underline{F}(\mathbf{P}^I), \bar{F}(\mathbf{P}^I)]$ that bounds f . To compute that result, we break down the last term $\Delta\Lambda(t; \mathbf{P}^I)$ in the integral of equation (A.11), into a unique finite sequence of interval arithmetic operations presented below as Algorithm 1.

Algorithm 1: Eval_Lognormal($[\underline{\mu}_1, \bar{\mu}_1], [\underline{\sigma}_1, \bar{\sigma}_1], [\underline{t}_0, \bar{t}_0], t$)

Set the constant variable const= $\sqrt{2\pi}$;

Compute equation (A.2) using the following list of expressions (‘;’ are used as separators for brevity).

$$T_1 = [\underline{\sigma}_1, \bar{\sigma}_1]^2; T_2 = t - [\underline{t}_0, \bar{t}_0];$$

$$T_3 = \ln(T_2); T_4 = T_3 - [\underline{\mu}_1, \bar{\mu}_1];$$

$$T_5 = T_3^2; T_6 = \frac{T_5}{T_1}; T_7 = 0,5 \times T_6$$

$$T_8 = -T_7; T_9 = \exp(T_8)$$

$$T_{10} = \text{const} \times [\underline{\sigma}_1, \bar{\sigma}_1]; T_{11} = T_2 \times T_{10};$$

$$T_{12} = \frac{T_9}{T_{11}};$$

return T_{12} ;

This algorithm starts by expressing f as a code list, where each arithmetic function (\pm, \times, \div , etc.) and unary elementary functions of the type ($\exp, \ln...$) are expressed with only one unique variable in their respective expressions. In this way, we ensure that the desirable properties (inclusion isotonicity, etc.) of the Fundamental Theorem are satisfied. Another advantage of using this method is the sharpness of the bound that results from this calculation process. In summary, Algorithm 1 takes as input each lognormal parameter P_i and the sample time t . From steps 1 to 3, the Delta-lognormal function is computed by calling Eval_Lognormal for each lognormal.

D. TESTING THE ALGORITHM UNDER IDEAL CONDITIONS

The algorithm was tested through two experiments:

- The ability of the resulting envelope to maintain, in a defined finite space, all the enclosed spaces of this set.
- Its ability to enhance the performance of the extractor in [50] , when used as a bounding operation. In this test we do not take into account the dependencies between the different parameters. Each parameter space was set according to its domain.

1. TESTS CONDITIONS

The experiments performed to validate Algorithm 1 use the database proposed in [22] and [50]. This database is composed of 7000 Delta-lognormal curves, grouped into categories. These ideal curves were generated automatically using parameter sets derived from human handwriting movements, previously extracted using the **IIX** algorithm presented in [22]. The choice to use synthetic data for the test is justified by the fact that the exact values of the parameters of the speed profiles to be tested are known. Therefore, the extraction algorithm can be validated based on its ability to find the same values as in the truth table. To this end, we chose randomly 300 ideal Delta-lognormal parameters from class Ca0 and Ca2 (see Figure A.1). Each ideal velocity profile, constructed from the ideal Delta-lognormal parameters (by evaluating equation A.1), was sampled

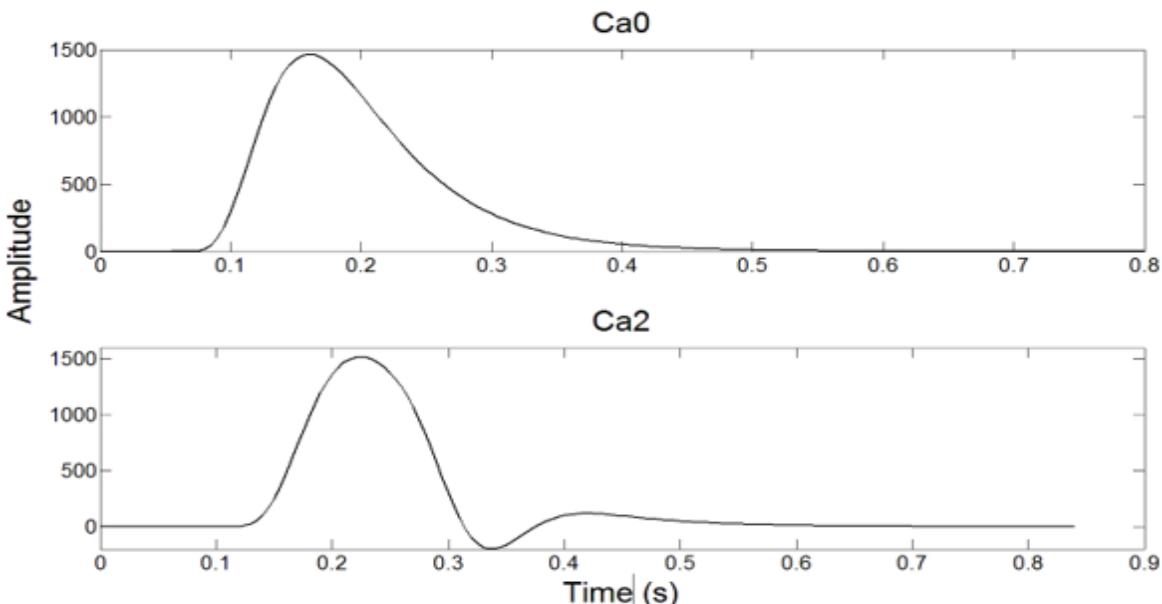


Figure A.1 Delta-lognormal velocity profile example

at 200Hz to simulate the data collected with a digitizer. In the second experiment, we replaced the boundary operations contained in the branch and bound algorithm proposed in [50] by Algorithm 1. The resulting speed profiles were then fed to this reconstructed algorithm to try to recover the true parameters value.

We carried out tests using a 3.60 GHz Intel core (i7) with 32 Go of RAM. The implementation was done in C++ using our custom interval arithmetic library based on [58] and [74]. All computations were performed using rounded interval arithmetic.

2. RESULTS AND DISCUSSION

For the first experiment, Table A.1 presents the search region that is restricted, for simplicity, to a finite box where the solution can be found.

Table A.1 The initial box

Parameters	t_0	D_1	μ_1	σ_1	D_2	μ_2	σ_2
Minimum	0.0001	50	-5	0.001	10	-5	0.001
Maximum	0.9	500	0.005	1.0	300	0.005	0.9

As shown in Figure A.2, thick black lines represent the envelope produced by Algorithm1 with the bounding box values of Table A.1. Inside this envelope, more than 3000 Delta-lognormal curves are shown as thin gray lines. The parameters of these curves have been generated by splitting the bounding box into sub-boxes such that $P_k \subset P$ for $k=1\dots n$. As can be seen, the envelope bounds

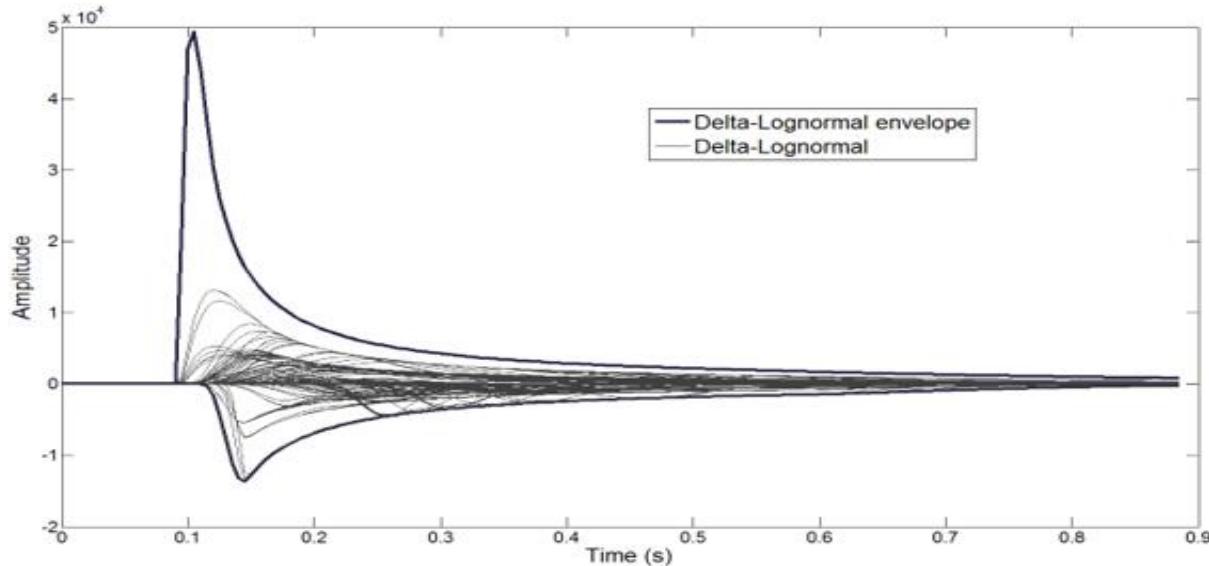


Figure A.2 Experimental testing of the Delta-Lognormal envelope

the possible values of equation (A.2) inside the given space quite well, since no curve extends beyond these borders.

In the second experiment, a box was considered as a solution if the upper bound of its quadratic error, or its maximum width, was less than 10^{-5} . Another criterion was based on equation (A.5) where the signal to noise ratio of the optimal box has to be more than 100dB. To this end, we computed the noise by evaluating equation (A.4) with real arithmetic using the middle of the optimal box as reference.

Table A.2 Statistics of the absolute error on the parameters estimated from ideal Delta-lognormal velocity profile

Parameters	Mean error	STD	Minimum	Maximum
t_0	1,18E-09	1,0E-09	1,6E-10	2,1E-09
D_1	2,90E-06	2,2E-06	6,7E-07	5,1E-06
μ_1	1,08E-08	3,5E-09	7,1E-09	1,4E-08
σ_1	4,95E-09	5,2E-10	4,4E-09	5,4E-09
D_2	2,01E-06	1,2E-06	7,6E-07	3,2E-06
μ_2	1,00E-08	3,1E-09	6,8E-09	1,3E-08
σ_2	8,58E-09	5,4E-09	3,1E-09	1,4E-08

Table A.3 Statistics of the absolute error on the parameters estimated from ideal Delta-lognormal velocity profile (the bounding box)

Bounding box		
Parameters	Minimum	Maximum
t_0	8,5E-06	9,4E-06
D_1	2,5E-08	2,1E-08
μ_1	1,0E-08	1,9E-08
σ_1	4,8E-06	7,2E-06
D_2	2,7E-08	3,3E-08
μ_2	1,9E-08	3,1E-08
σ_2	2,4E-09	1,9E-09

Table A2 and A.3 present the main statistics of the absolute error on each estimated parameter. As can be seen for the 300 tested curves, the errors on the estimates of t_0 and D_1 are on average 10^{-9} and 10^{-6} respectively. We can also observe that the new algorithm always finds not only the true value within an accuracy of $\varepsilon = 10^{-6}$, but a confidence interval around the true value (Table A.3). This interval contains all the possible solutions satisfying at least the ending criteria. Thus, if we decide to generate a solution with a smaller error (up to the limit of numerical precision), we just

need to continue the search from that bounding box. This is an advantage of using interval arithmetic. All the results compiled in this table were obtained for a signal to noise ratio of more than 100dB.

Figure A.3 presents an example in which the Delta-lognormal parameters of a real human generated velocity profile of a pen-tip movement are extracted using Algorithm 1 combined with [50]. As can be seen in the middle trace of Figure A.3, the extracted velocity profile fits quite well (to an SNR=31dB) the human speed profile of the pen-tip movement shown in the top trace of Figure A.3. The bottom trace in Figure A.3 shows the extracted velocity profile decomposed into two neuromuscular systems acting in direct opposition as predicted by the Delta-lognormal model. The agonist and the antagonist are respectively plotted in the figure by dashed and thin lines around the continuous line movement.

Table A.4 shows a performance comparison in terms of processing time and number of boxes explored for the proposed algorithm and the one proposed in [15]. In [15], the algorithm was executed in parallel, on a 12-CPU node grid. As can be seen in Table A.4 below, this algorithm requires a run time of about 50s to perform 11 iterations of its main loop, while the one we propose requires only 702ms. On the other hand, the number of boxes needed to find a solution of the

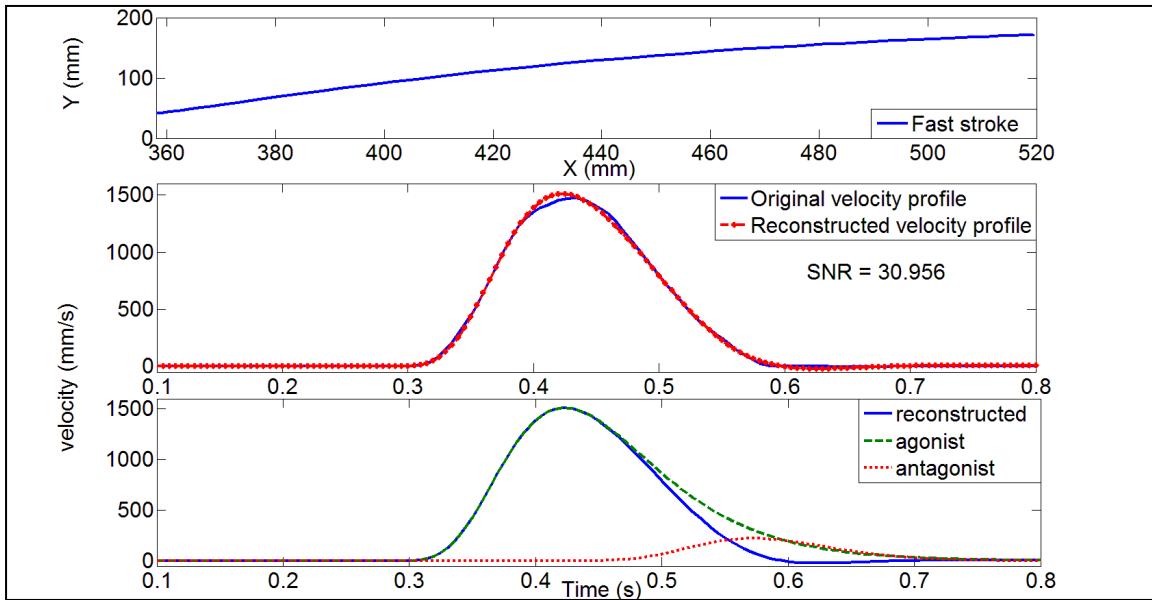


Figure A.3 Example of application of algorithm 1 in extracting real human pen-tip movement

desired quality can reach one million with the previous algorithm, while ours requires at most 70 thousand boxes to extract such a solution.

Table A.4 Comparison of the number of boxes explored to extract one velocity profile and the processing time required to perform 11 iterations in the main loop

Methods		
Performance metric	[50]	Proposed
Number of Boxes	Min 800000	Max 70000
Processing time (s)	50	0.702

D. CONCLUSION

In this paper we have shown that, with a good bounding method applied to the Delta-Lognormal function, when embedded in a branch and bound algorithm, the parameters of Delta-Lognormal velocity profiles can be extracted with less computational costs. The effectiveness of the proposed bounding method is quite remarkable. The number of evaluations was reduced 12-fold to obtain results of a same target quality. This algorithm exploits the natural interval extension and the fundamental theorem of interval arithmetic to compute the bounding operations of the Delta-Lognormal function. Although the proposed algorithm is promising in its ability to recover a global solution in a processing time much smaller than a previously reported method, more work remains to be done. The proposed algorithm should be tested on all classes of velocity profiles that the Delta-Lognormal expression can model, and it should be more thoroughly tested on real human movement data.

**ANNEXE B ARTICLE 2: ON THE USE OF INTERVAL ARITHMETIC
FOR BRANCH AND BOUND DELTA-LOGNORMAL PARAMETER
EXTRACTION OF RAPID HUMAN MOVEMENTS**

Simon Pierre Boyogueno Bidias, Jean-Pierre David, Yvon Savaria, Réjean Plamondon

Département de génie Électrique Polytechnique Montréal.

Montréal, Canada H3C 3A7

simon-pierre.boyogueno-bidias@polymtl.ca, JPDavid@polymtl.ca, yvon.savaria@polymtl.ca,
rejean.plamondon@polymtl.ca

This paper is published in the book « The Lognormality Principle and its applications in E-Security, E-learning and E-Health » on December 10 2020, for the International Conference on Pattern Recognition and Artificial Intelligence: Workshop on the Lognormality Principles and its Applications (ICPRAI) Montréal 2018.

The paper has been submitted on November 07, 2019.

Abstract. This chapter proposes an algorithm based on interval arithmetic to construct envelope models that contain all possible parameter values of a Delta-Lognormal function that can match an observed human movement. In the context of the Kinematic Theory of rapid human movements, these parameters represent the basic timing properties of a neuromuscular system, as well as the control strategies used to produce a rapid movement. The basic idea behind the proposed algorithm is to exploit the isotonicity and monotonicity over an interval to sharply bound the range of the Delta-Lognormal function within a finite space of its parameters. The new algorithm is described, tested (under various handwritten strokes) and compared with another algorithm previously developed for the same purpose. The numerical results show that the proposed algorithm gives better results in terms of speed and accuracy and enables the creation of new tools for the basic study of single strokes used as primitives for understanding rapid human movements, according to the Kinematic Theory.

A. INTRODUCTION

Human movements can be very complex, and their studies in terms of motor control have been intensively used in many fields of research. From a general perspective, the complex levels of manual dexterity that humans exhibit can be attributed and demonstrated in tasks controlled by the nervous system. Specifically, handwriting gestures are part of this type of rapid movements, where we find an intervention of the motor system to achieve them. In neurosciences, handwritten strokes constitute a specific class of rapid human movements, notably used to study neurodegenerative processes such as those involved in Parkinson and Alzheimer's diseases [5], [15]. These simple movements can be seen as the basic primitives [12] that are combined to produce more complex movements like handwriting signatures. They were deeply studied by psychologists, neuroscientists, physicists, and computer scientists over the last century. Handwriting movements are also used in cybersecurity applications [80], [83] as well as for the rehabilitation of patients having suffered from cerebrovascular accidents [16], [81]. In forensic science, handwritten strokes are used to validate authorship of known suspected writers [84], [85].

In this context, the need to study the elementary properties of a single stroke for a realistic analysis becomes a key step to understand how the motor control system accomplishes complex movements. Modelling this is a powerful way to understand the basic principles of human motor control and its disorders [86], [87]. Among the various models of rapid human movements proposed in the literature, the Kinematic Theory is commonly used because of its ability to reproduce faithfully a handwritten stroke, with a set of parameters having a physiological meaning for these movements [19], [20]. This theory presents handwritten strokes as primitives that contain information on both the motor control and the neuromuscular system [75]. The study of such strokes relies on the extraction of parameters that characterize their velocity profile [48], [47].

To meet this need, three deterministic algorithms [22], [47], [48], have been developed in our laboratory in order to extract the lognormal parameters derived from handwriting velocity profiles. These algorithms have successfully led, for example, to applications for verification of signatures and recognition of handwriting [70]. Although the parameter extractors developed in [22], [47], [48], show good data fitting, none of these prior works can guarantee the *optimality* or *uniqueness* of the solution obtained. Note that defining what constitutes an optimal solution is not so simple, but in loose terms, for medical doctor to accept extracted parameters as part of a diagnosis process,

they ask for a level of certainty that there is no other *good enough* solution that departs significantly from the one provided by a parameter extraction system. Such a requirement is not always critical for some applications since excellent curve fitting is sufficient for many patterns recognition tasks. However, globally optimal solutions are important in biomedical applications where a confidence interval on the value of an extracted parameter is necessary in developing a benchmark, or a pre-screening. To address this need, a first branch and bound algorithm was developed in [50].

It is based on building a tree where each node represents a possible solution of the problem considered. The nodes are Delta-Lognormal envelopes that correspond to a bounding box, built by studying the variations of the Delta-Lognormal function, according to its seven parameters. Each subset or child node is created by subdividing the parent node into three children. The dimension according to which a parent node is subdivided is selected depending on the variables that are estimated to minimize the area of the parent Delta-Lognormal envelope. The so-constructed tree is traversed in a way that non-promising solutions are pruned based on a monotonicity test. In summary, the branch and bound algorithms proposed in [50] is based on a Delta-Lognormal envelope that bounds all the possible Delta-Lognormal parameters within a subspace, and then computes a bound on the error between the signal and the nearest envelope. This method, despite the fact that it clearly circumscribed the challenge, suffers from lower bound problems (according to [50], the previously proposed procedure for computing an envelope to the velocity profile did not always hold) and each calculation of the estimated area of the envelope requires long processing time. Thus, the total processing time could be reduced significantly if these calculations were not performed so often. The aim of this chapter is to propose a new algorithm to relieve the bounding problem of the extractor proposed in [50] and show its effectiveness when combined in a branch and bound scheme to extract model parameters from ideal velocity profile data. The proposed method is used to bound the Delta-Lognormal parameters representing synthetic velocity profile data.

The rest of the chapter is organized as follows. Section 2 reviews the Delta-Lognormal equation and the modelling tools that will be used to bound the objective function. Section 3 introduces some theoretical aspects of interval arithmetic and proposes an algorithm for computing enclosures of the bound of the objective function. Section 4 presents numerical results and performance comparisons with [50]. Section 5 discusses the importance of the proposed algorithm, and finally Section 6 proposes possible future research and summarizes the main contributions of this chapter.

B. THE DELTA-LOGNORMAL MODEL

The Kinematic Theory of rapid human movements, from which the Delta-Lognormal model is derived, considers the trajectory of a pen tip as a single stroke having a delta- lognormal velocity profile. La notation utilisée depuis toujours est de mettre sigma carré.

$$\Delta\Lambda(t; p) = D_1\Lambda(t; t_0, \mu_1, \sigma_1^2) - D_2\Lambda(t; t_0, \mu_2, \sigma_2^2) \quad (\text{B.1})$$

where

$$\Lambda(t; t_0, \mu, \sigma^2) = \begin{cases} \frac{1}{\sigma\sqrt{2\pi}(t-t_0)} \exp^{-\frac{(\ln(t-t_0)-\mu)^2}{2\sigma^2}}, & \text{for } t_0 < t, \\ 0, & \text{elsewhere.} \end{cases} \quad (\text{B.2})$$

With $\sigma > 0 ; \mu, t_0 \in \mathbb{R}$

that is the lognormal impulse response function. In this theory, a stroke is produced by a synergy of two neuromuscular systems: the agonist, acting in the direction of the movement and the antagonist, acting in the opposite direction. The agonist and antagonist are lognormals convolved with their corresponding input commands, represented by the subscript 1 and 2 in equation (B.1) respectively. The subtraction of the two system responses produces the velocity profile $\Delta\Lambda(t; p)$ described by the seven parameters P shown in equation (B.3).

$$P = [t_0, D_1, \mu_1, \sigma_1, D_2, \mu_2, \sigma_2] \quad (\text{B.3})$$

In (B.3), t_0 represents the system activation time, and $\mu_1, \sigma_1, \mu_2, \sigma_2$ correspond to the timing properties (on a logarithmic scale) of the two neuromuscular commands, in reaction to their input amplitudes $D_1 > 0$ and $D_2 > 0$ respectively.

To bound the model parameters from velocity data, we need to find the bounding box P that contains all the possible values of the seven parameters p that fit the original velocity movement $v_t(t)$ (collected from synthetic human movements) most accurately. This problem can be solved by building an envelope according to the nonlinear least square function f (B.4) that maximizes the signal to noise ratio SNR (B.5).

$$f = \int (v_t(t) - \Delta\Lambda(t; p))^2 dt \quad (\text{B.4})$$

$$SNR = 10 \log \left(\frac{\int v_t^2(t) dt}{f} \right) \quad (\text{B.5})$$

C. INTERVAL ARITHMETIC

Before formulating the proposed algorithm, some definitions and notations must be established. Consider the following problem:

Bound $f(x)$ subject to $x \in \mathbf{X}^I$, where f is an n dimensional continuously differentiable function $\mathbb{R}^N \rightarrow \mathbb{R}$ and $\mathbf{X}^I \subseteq \mathbb{R}^N$ is an n dimensional interval vector.

We denote by $I = [a, b] = \{x \in \mathbb{R} : a \leq x \leq b\}$ the set of real numbers and one dimensional closed intervals. In this chapter, an interval will be denoted with a variable in upper case and the set of values contained in the interval, in lower case. Thus $X = [\underline{x}, \bar{x}]$ is an interval variable where \underline{x} and \bar{x} represent the left and right end points of X respectively. The width of X is defined and denoted by $w(X) = \bar{x} - \underline{x}$ and the midpoint is given by:

$$m(X) = \frac{1}{2}(\bar{x} + \underline{x}).$$

We denote by $\mathbf{X}^I = (X_1, X_2, \dots, X_n)$ the set of n dimensional interval vectors or box such that:

$$\mathbf{X}^I = (X_1, X_2, \dots, X_n) = ([\underline{x}_1, \bar{x}_1], [\underline{x}_2, \bar{x}_2], \dots, [\underline{x}_n, \bar{x}_n]).$$

The width and the midpoint of an interval vector are defined as:

$$w(\mathbf{X}^I) = \max_i w(X_i^I)$$

$$m(\mathbf{X}^I) = (m(X_1), m(X_2), \dots, m(X_n)).$$

We call F an inclusion function of $f: \mathbb{R}^N \rightarrow \mathbb{R}$, the interval extension of f , such that for $x \in X \Rightarrow f(x) \subseteq F(X)$. $f(X)$ is the real range of f on X and $f(X) \subseteq F(X)$. Now that we have these definitions and notations, it is possible to use interval arithmetic [59], [58] to bound a function defined by a mathematical expression such as the Delta-Lognormal function.

1. INTERVAL ARITHMETIC OPERATIONS

In this section, we summarize some principles of interval arithmetic needed to derive the bounding operations of the objective function stated in **(B.4)**. For more details, the reader can consult [59], [58]. The basic arithmetic operations between two intervals $X = [\underline{x}, \bar{x}]$ and $Y = [\underline{y}, \bar{y}]$ are defined in Table B.1. Apart from interval arithmetic operations, interval functions are also defined as the image that corresponds to the set of interval arguments of a given function into an interval. The

basic operations on usual (rational and transcendental) functions of one variable are summarized in Table B.2.

Table B.1 Basic arithmetic operations between two intervals

Operations	Results
$X + Y$	$[\underline{x} + \underline{y}, \bar{x} + \bar{y}]$
$X - Y$	$[\underline{x} - \bar{y}, \bar{x} - \underline{y}]$
$X \times Y$	$[\min S, \max S]$ with $S = \{\underline{x}\underline{y}, \underline{x}\bar{y}, \bar{x}\underline{y}, \bar{x}\bar{y}\}$
$X \div Y$	$X \times \left(\frac{1}{Y}\right)$, with $\left(\frac{1}{Y}\right) = \left[\frac{1}{\bar{y}}, \frac{1}{\underline{y}}\right]$ and $0 \notin Y$

Table B.2 Basic arithmetic functions

Functions	Results
$\exp(X)$	$[\exp(\underline{x}), \exp(\bar{x})]$
$\ln(X)$	$[\ln(\underline{x}), \ln(\bar{x})]$ for $0 < \underline{x}$
$X^{1/2}$	$[\underline{x}^{1/2}, \bar{x}^{1/2}]$
X^n	$\begin{cases} [\underline{x}^n, \bar{x}^n] & \text{if } \underline{x} > 0 \text{ or } n \text{ is odd} \\ [\bar{x}^n, \underline{x}^n] & \text{if } \bar{x} < 0 \text{ and } n \text{ is even} \\ [0, \max \underline{x}^n, \bar{x}^n] & \text{if } 0 \in X \text{ and } n \text{ is even} \end{cases}$

Now that we have defined the basic interval operations, we now extend the concept to more general functions of n interval variables such as the Delta-Lognormal (see equation B.1 and B.2).

Definition 1: the natural interval extension of a given function $f(x_1, x_2, \dots, x_n)$ of n variables is given by the interval function $F(X_1, X_2, \dots, X_n)$, which is obtained by replacing the real variable x with the corresponding interval variable X .

Example 1: The natural interval extension of

$$f(x, y) = (1 - x)^2 + 100(y - x^2)^2 \quad (\text{B.6})$$

$$g(w, z) = \exp\left(\frac{w+z}{z^2+w}\right) \quad (\text{B.7})$$

Are:

$$F(X, Y) = (1 - X)^2 + 100(Y - X^2)^2$$

$$G(W, Z) = \exp \left(\frac{W + Z}{Z^2 + W} \right)$$

Where X, Y, W and Z are intervals. If we assign for example $X = [-1, 2]$, $Y = [-1, 3]$, $W = [0, 2]$, $Z = [-1, 1]$ we have:

$$F([-1, 2], [-1, 3]) = (1 - [-1, 2])^2 + 100([-1, 3] - [-1, 2]^2) = [0, 2504] \quad (\text{B.8})$$

$$\begin{aligned} G([0, 2], [-1, 1]) &= \exp \left(\frac{[-1, 3]}{[0, 1] + [1, 1]} \right) \\ &= \exp([-1, 3]) = [\exp(-1), \exp(3)] \end{aligned} \quad (\text{B.9})$$

It is of interest that, in these examples:

In (B.8), if we take any interval, say $X' = [-1, 1] \subseteq X$, the value of $F(X', Y) = [0, 904]$ is contained in $F(X, Y)$. This important property is called inclusion isotonicity.

In the right-hand side of equation (B.9), we use the monotonicity of the exponential function to get the result. This property is also very important for bounding operations and is called inclusion monotonicity.

To this end, the following fundamental theorem of interval arithmetic can be stated:

Fundamental Theorem. Let $\mathbf{F}(X_1, X_2, \dots, X_n)$ be the natural interval extension of $\mathbf{f}(x_1, x_2, \dots, x_n)$ then

$$\mathbf{f}(X_1, X_2, \dots, X_n) \subseteq \mathbf{F}(X_1, X_2, \dots, X_n),$$

and for all intervals,

$$Y_k \subset X_k, \text{ for } k = 1, \dots, n, \mathbf{f}(Y_1, Y_2, \dots, Y_n) \subseteq \mathbf{F}(Y_1, Y_2, \dots, Y_n),$$

where

$$\mathbf{f}(Y_1, Y_2, \dots, Y_n) = \{ \mathbf{f}(x_1, x_2, \dots, x_n) : x_k \in Y_k \text{ for } k = 1, \dots, n \}.$$

This theorem due to Moore [58] was extended by Hansen [59]. It states that, the evaluation of a function in any point inside a closed interval where both the function and its variables are defined is guaranteed to produce a result contained in the natural interval extension of the function. Its proof can be found in [59]. This Fundamental Theorem shows how easy it is to bound the range

of a function, and it makes possible the resolution of the bounding problem stated in Section 1. In the next sub-section, we will apply this theorem to the Delta-Lognormal function as a specific sequence of interval arithmetic operations to bound the range of equation (B.4) within an interval vector.

2. ALGORITHM FOR COMPUTING THE UPPER AND LOWER BOUND OF THE DELTA-LOGNORMAL FUNCTION.

To compute a bound for equation (B.4), we first define the bounding space of its variables as:

$$\mathbf{P}^I = \{[\underline{D}_1, \bar{D}_1], [\underline{\mu}_1, \bar{\mu}_1], [\underline{\sigma}_1, \bar{\sigma}_1], [\underline{D}_2, \bar{D}_2], [\underline{\mu}_2, \bar{\mu}_2], [\underline{\sigma}_2, \bar{\sigma}_2], [\underline{t}_0, \bar{t}_0]\} \quad (\text{B.10})$$

Then, through the Fundamental Theorem, we apply the natural interval extension on f . This yield:

$$\begin{aligned} f(\mathbf{P}^I) \in F(\mathbf{P}^I) &= [\underline{F}(\mathbf{P}^I), \bar{F}(\mathbf{P}^I)] \\ &= \int (\nu_t(t) - \Delta\Lambda(t; \mathbf{P}^I))^2 dt \end{aligned} \quad (\text{B.11})$$

Equation (B.11) denotes the quadratic error between a hand-written velocity profile and its Delta-Lognormal approximation. We have chosen this error rather than other because it takes into account the fact that a large error on a small portion of the signal is considered more problematic than a small error on a large portion of the signal.

As provided by the theorem and the natural interval extension, the result of the right-hand side of equation (B.11) is an interval $[\underline{F}(\mathbf{P}^I), \bar{F}(\mathbf{P}^I)]$ that bounds f . To compute that result, we break down the last term $\Delta\Lambda(t; \mathbf{P}^I)$ in the integral of equation (B.11), into a unique finite sequence of interval arithmetic operations [57] presented below as Algorithm 1.

Algorithm 1 : Eval_Lognormal($[\underline{\mu}_1, \bar{\mu}_1], [\underline{\sigma}_1, \bar{\sigma}_1], [\underline{t}_0, \bar{t}_0], t$)

Set the constant variable const= $\sqrt{2\pi}$;

Compute equation (B.2) using the following list of expressions (‘;’ are used as separators for brevity).

$$T_1 = [\underline{\sigma}_1, \bar{\sigma}_1]^2 ; \quad T_2 = t - [\underline{t}_0, \bar{t}_0] ;$$

$$T_3 = \ln(T_2) ; \quad T_4 = T_3 - [\underline{\mu}_1, \bar{\mu}_1] ;$$

```

 $T_5 = T_4^2 ; T_6 = \frac{T_5}{T_1} ; T_7 = 0,5 \times T_6 ;$ 
 $T_8 = -T_7 ; T_9 = \exp(T_8) ;$ 
 $T_{10} = \text{const} \times [\underline{\sigma}_1, \bar{\sigma}_1] ; T_{11} = T_2 \times T_{10} ;$ 
 $T_{12} = \frac{T_9}{T_{11}} ;$ 
return  $T_{12} ;$ 

```

This algorithm starts by expressing f as a code list, where each arithmetic function (\pm, \times, \div , etc.) and unary elementary functions of the type (\exp, \ln, \dots) are expressed with only one unique variable in their respective expressions. In this way, we ensure that the desirable properties (inclusion isotonicity, etc.) of the Fundamental Theorem are satisfied. Another advantage of using this method is the sharpness of the bound that results from this calculation process. In summary, Algorithm 1 takes as input each lognormal parameter P_i and the sample time t . From steps 1 to 3, the Delta-Lognormal function is computed by calling `Eval_Lognormal` for each lognormal.

D. APPLYING ALGORITHM 1 IN A GENERAL BRANCH AND BOUND SCHEME

The general idea behind the branch and bound technique is to bound sub-boxes of a feasible area in which we perform a search to try obtaining the sharpest one. The bounding operations can be carried out by procedures such as in Algorithm 1 and, if the bounds under consideration are not sufficiently tight, some boxes are divided into smaller one according to a splitting rule. The procedure is repeated until solutions is obtained within a specified target accuracy. Such a solution is called in this work the optimal solution. Of course, requesting a more accurate solution would normally lead to further refinement and thus a different final solution. What we aim to do in this work is prohibiting that such a different better solution would not be enclosed in some previous best solution obtained with a less stringent accuracy criterion.

In other words, in the context of this chapter, we seek to find the best solution that fits an input velocity profile inside a defined finite space. The optimal solution is a set of points that maximizes the **SNR**. Finding it exactly remains an elusive goal. In practice, the proposed algorithm returns a set of boxes for which the respective lowest **SNR** of each box is higher than a target value and the overall best solution is guaranteed to be in one of the returned boxes. The objective function to be minimized is provided by equation (B.4) and the accuracy is fixed by equation (B.5) to 100dB

SNR. To facilitate the reading and comprehension of this chapter, a summary of the algorithm explored here is listed below as algorithm 2. This algorithm is adapted from [50]. A notable difference adopted for simplicity is that algorithm 2 splits boxes in 2 while the one in [50] was trisecting boxes. In this summary, the various operations proposed by [50] have been replaced by the operations presented in table B.2, B.3 and algorithm1. For a more elaborate version of this description, we refer the reader to [50].

Algorithm 2: General multidimensional interval branch and bound

1. Set the initial parameter \mathbf{P}^I . Set the working list $L_1 := \{(\underline{F}(\mathbf{P}^I), \mathbf{P}^I)\}$, the final list $L_2 := \{\}$ and the upper bound $\bar{f}^* = f(m(\mathbf{P}^I))$
2. While ($L_1 \neq \{\}$)
3. Select box \mathbf{P}^I from L_1
4. $L_1 = \mathbf{P}^I$
5. Bisect \mathbf{P}^I into \mathbf{P}_i^I sub boxes $i = 1..2$
6. For $i = 1$ to 2
7. Compute the bound $f(m(\mathbf{P}_i^I)), \underline{F}(\mathbf{P}_i^I)$ (Algorithm 1)
8. If not ($\underline{F}(\mathbf{P}_i^I) > \bar{f}^*$)
9. $\bar{f}^* = \min\{\bar{f}^*, f(m(\mathbf{P}_i^I))\}$; Update upper bound
10. If \mathbf{P}_i^I satisfy the ending criterion (Equation (B.5))
11. $L_2 += \{(\underline{F}(\mathbf{P}_i^I), \mathbf{P}_i^I)\}$;
12. Else $L_1 += \{(\underline{F}(\mathbf{P}_i^I), \mathbf{P}_i^I)\}$;
13. Return $L_2, \bar{f}^* = \bar{f}$

As presented above, Algorithm 2 takes as input a feasible area or box P . It uses a working list L_1 , and produce a final list L_2 . The objective function f (see 4) and an upper bound \bar{f}^* are computed by evaluating f at the midpoint of the initial box. The feasible area is restricted to a search region where we know that the solution can be found. The values of the intervals defining the initial the bounding box are chosen as wide as necessary to almost certainly safely enclose the solution. The working list L_1 , is the tree generated throughout the search by the division of boxes into sub-boxes. The final list contains solutions that meets the ending criteria presented in step 10 of the algorithm. The algorithm starts by bisecting a box and compute the bounds of each resulting sub-box using Algorithm 1. This is done in step 5 and 7 respectively. Next, the sub-boxes previously obtained are subjected to a monotonicity test (see step 8) where we validate if their lower bound is less than the current best upper bound known. If it is the case, the box is kept in the working list and the process continues. Otherwise, the box is rejected because the test has just proven that the box cannot contain a better solution. The ending criterion is evaluated in step 10,

where Equation (B.5) is evaluated for each box having passed the monotonicity test. We add to the list L_2 the boxes that have satisfied a given **SNR**. This is illustrated in step 11, where the subscript $+=$ and $-$ (step 4) in front of each list, denote a push and a pop of a box through the list. The algorithm returns the final best solution f^* and the solution list L_2 where we can find all the possible solution satisfying the ending criterion.

1. TESTING THE ALGORITHM UNDER IDEAL CONDITIONS

The algorithm has been tested through two experiments to evaluate the ability of the resulting envelope:

- To maintain, in a defined finite space, all the enclosed spaces of this set.
- To enhance the performance of the extractor in [50], when used as a bounding operation. In this latter test, we did not take into account the dependencies between the different parameters. The bounding interval of parameter space was set according to its domain.

2. TESTS CONDITIONS

The experiments performed to validate Algorithm 1 use the database proposed in [22] and [50]. This database is composed of 7000 Delta-Lognormal curves, grouped into categories. These ideal curves were generated automatically using parameter sets derived from human handwriting

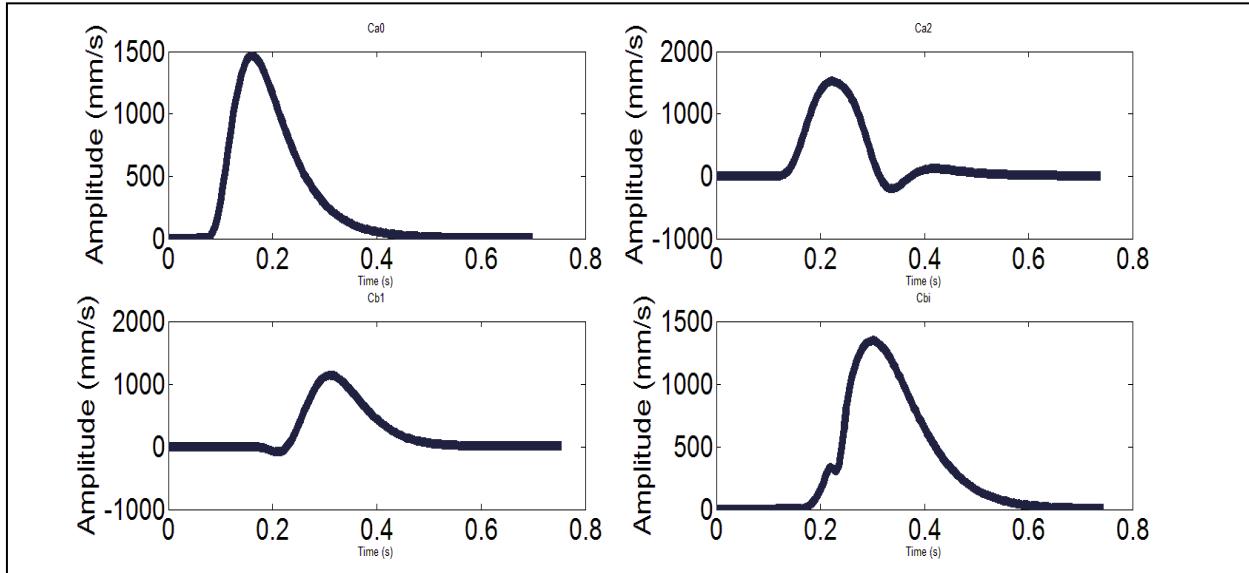


Figure B.1 Delta-Lognormal velocity profile examples

movements, previously extracted using the **IIX** algorithm presented in [22]. The choice to use synthetic data for the test is justified by the fact that the exact values of the parameters of the speed

profiles to be tested are known. Therefore, the extraction algorithm can be validated based on its ability to find the same values as in the ground-truth table. We chose randomly 500 ideal Delta-Lognormal parameters from class Ca0, Ca2, Cb1 and Cbi (see Figure B.1). Each ideal velocity profile constructed from the ideal Delta-Lognormal parameters (by evaluating equation B.1), were sampled at 200 Hz to simulate the data collected with a digitizer [65]. In the second experiment, we replaced the boundary operations contained in the branch and bound algorithm proposed in [50] by Algorithm 1. The created speed profiles were then fed to the modify algorithm (Algorithm 2) to try to recover the true parameters values. We carried out the tests in a 3.60 GHz Intel core (i7) with 32 GB of RAM. The implementation was done in C++ using our custom interval arithmetic library based on [58] and [74]. All computations were performed using rounded interval arithmetic.

E. RESULTS AND DISCUSSION

For the first experiment, Table B.3 presents the search region, which is restricted for simplicity to a finite box where the solution can be found. The bounding box was constructed by taking the minimum of the minimums and the maximum of the maximums of each parameter value for all the classes.

Table B.3 The initial box

Parameters	t_0	D_1	μ_1	σ_1	D_2	μ_2	σ_2
Minimum	0.0001	50	-5	0.001	10	-5	0.001
Maximum	0.9	500	0.005	1.0	300	0.005	0.9

As shown in Figure B.2, thick blue lines represent the envelope produced by Algorithm 1 using the natural interval extension and the bounding box values of Table B.3. Inside this envelope, more than 3000 Delta-Lognormal curves are shown as thin red lines. The parameters of these curves have been generated by splitting the bounding box into sub-boxes such that $P_k \subset P$ for $k=1 \dots n$. As can be seen, the envelope bounds the possible values of equation (B.2) inside the given space quite well, since no curve extends beyond these borders.

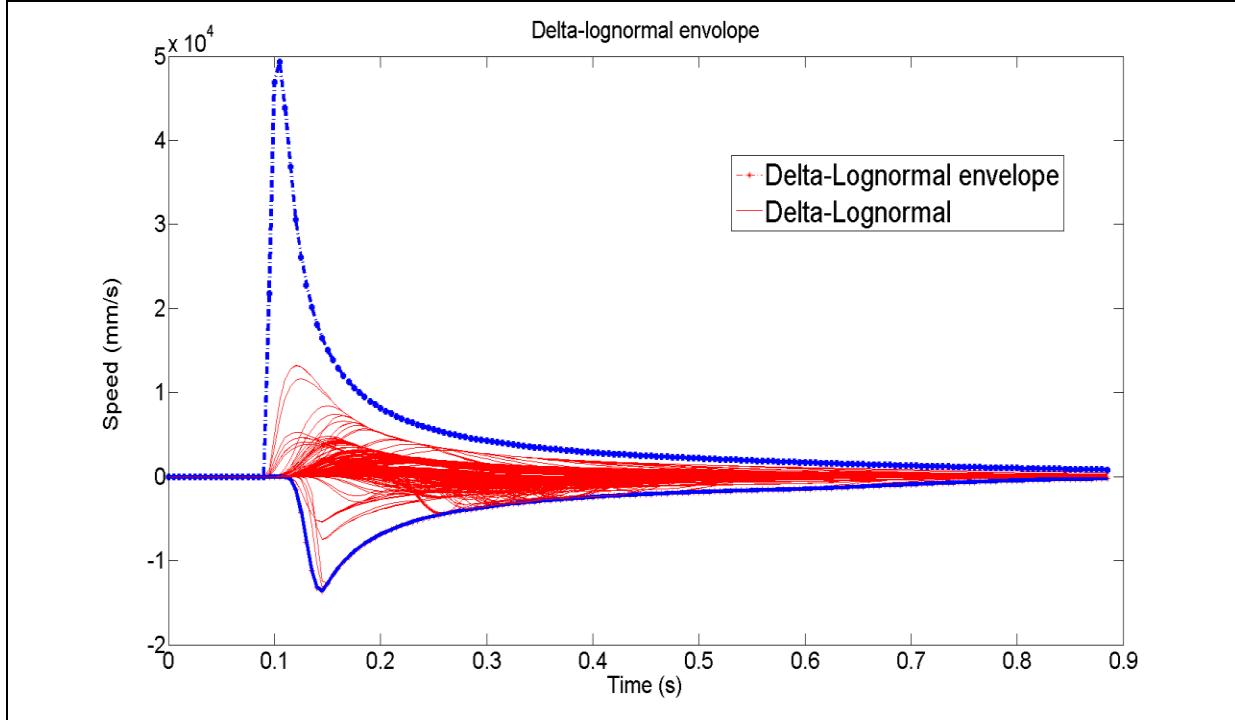


Figure B.2 Experimental testing of the Delta-Lognormal envelope

In the second experiment, a box was considered as a solution if the upper bound of its quadratic error, or its maximum width, was less than 10^{-5} mm/s . Another criterion was based on equation (B.5), where the signal to noise ratio of the chosen box has to be larger than 100 dB. To this end, we computed the noise by evaluating equation (B.4) with real arithmetic using as reference, the middle of the satisfying box.

Table B.4 Statistics of the absolute error on the parameters estimated from ideal Delta-Lognormal velocity profiles

Parameters	Mean error	STD	Minimum	Maximum
$t_0(s)$	1,18E-09	1,0E-09	1,6E-10	2,1E-09
$D_1(\text{ mm})$	2,90E-06	2,2E-06	6,7E-07	5,1E-06
$\mu_1(s)$	1,08E-08	3,5E-09	7,1E-09	1,4E-08
$\sigma_1(s)$	4,95E-09	5,2E-10	4,4E-09	5,4E-09
$D_2(\text{ mm})$	2,01E-06	1,2E-06	7,6E-07	3,2E-06
$\mu_2(s)$	1,00E-08	3,1E-09	6,8E-09	1,3E-08
$\sigma_2(s)$	8,58E-09	5,4E-09	3,1E-09	1,4E-08

Table B.5 Statistics of the absolute error on the parameters estimated from ideal Delta-Lognormal velocity profiles (the bounding box)

Bounding box		
Parameters	Minimum	Maximum
t_0	0,0	9,4E-06
D_1	0,0	2,1E-05
μ_1	0,0	1,9E-06
σ_1	0,0	7,2E-06
D_2	0,0	3,3E-05
μ_2	0,0	3,1E-07
σ_2	0,0	1,9E-07

Table B.4 and B.5 present the main statistics of the absolute error on each estimated parameter. As can be seen for the 500 tested curves, the errors on the estimates of t_0 and D_1 are on average 10^{-9} s and 10^{-6} mm respectively. We can also observe that the new algorithm always finds not only the true value within an accuracy of $\varepsilon = 10^{-6}$, but a confidence interval around the true value (Table B.5). This interval contains all the possible solutions satisfying at least the ending criteria. Thus, if we decide to generate a solution with a smaller error (up to the limit of numerical precision), we just need to continue the search from that bounding box. This is an advantage of using interval arithmetic. All the results compiled in this table were obtained for a signal to noise ratio larger than 100 dB.

Table B.6 shows a performance comparison in terms of processing time and number of boxes explored for the proposed algorithm and the one proposed in [50]. In [50], the algorithm was executed in parallel, on a 12-CPU node grid. As can be seen in Table B.6, this algorithm requires a run time of about 50s to perform 11 iterations of its main loop (ie. Passing 11 times through all the operation defined in algorithm1), while the algorithm proposed in this chapter requires only 702ms on a single core. Of course, the processing time could be reduced with the new algorithm if it is parallelized to exploit multiple cores. That refinement is left for future research. On the other hand, the number of boxes needed to find a solution of the desired quality could reach one million with the previous algorithm, while ours requires at most 70 thousand boxes to extract such a solution.

Table B.6 Comparison of the number of boxes explored to extract one velocity profile and the processing time required to perform 11 iterations in the main loop

Performance metric	[50]	Proposed
Number of Boxes	Min 800000	Max 70000
Processing time (s)	50 (12 cores)	0.702 (3 core)

1. DISCUSSION

The results presented above show the efficiency of Algorithm 1 as well as the power of interval arithmetic in resolving nonlinear high-dimensional problems such as the Delta-Lognormal parameters extraction on ideal data (synthetic). The different rules used to reject unpromising solutions are unique in the field. They guarantee at all times through the construction of a bounding box, the unique and global character of the final solution with a reasonable computer cost. In order to validate the accuracy of Algorithm 1 and prove its efficiency, we anticipated a test on real human hand-written speed profiles of class c1. Despite the fact that this experiment is part of a subsequent

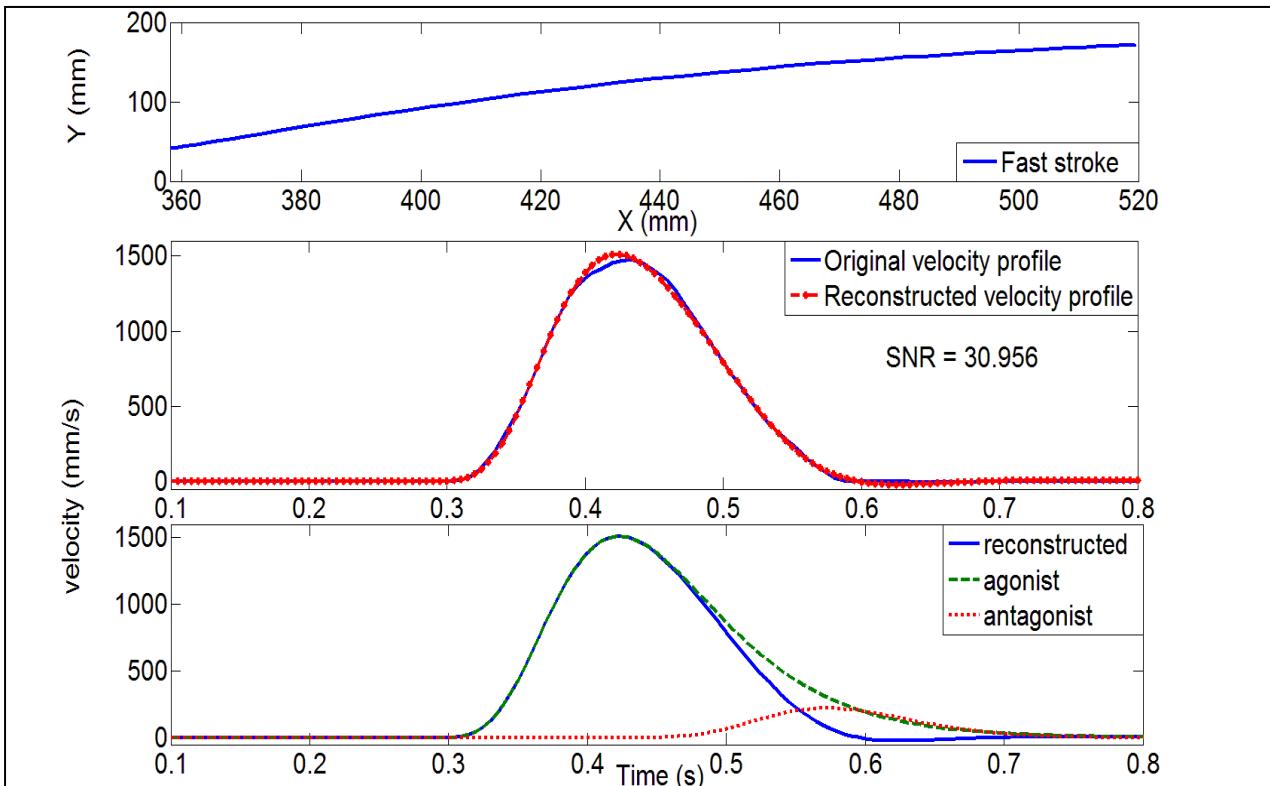


Figure B.3 Example of application of algorithm 1 in extracting real human pen-tip movement (the top trace in blue is the fast stroke, the middle traces are respectively the original (blue solid line) and the reconstructed (red stars line) velocity profile

study and not the subject of this chapter, we present in Figure B .3. An example in which the Delta-Lognormal parameters of a real human generated velocity profile of a pen-tip movement is extracted using Algorithm1 combined with [50]. As can be seen in the middle trace of Figure B.3, the extracted velocity profile fits quite well (with an SNR=30dB) the human speed profile of the pen-tip movement shown in the top trace of Figure B.3. The bottom trace in Figure B.3 shows the extracted velocity profile decomposed into two neuromuscular systems acting in direct opposition as predicted by the Delta-Lognormal model. The agonist and the antagonist outputs are respectively plotted in the figure by dashed and thin lines around the continuous line movement.

F. CONCLUSION

In this chapter, we have shown that, with a good bounding method applied to the Delta-Lognormal function, when embedded in a branch and bound algorithm, the parameters of Delta-Lognormal velocity profiles can be extracted with less computational costs than previous works can achieve. The effectiveness of the proposed bounding method is quite remarkable. The number of evaluations was reduced at least 70-fold to obtain results of a same target quality. This algorithm exploits the natural interval extension and the fundamental theorem of interval arithmetic to compute the bounding operations of the Delta-Lognormal function. Although the proposed algorithm is promising in its ability to recover a global solution in a processing time much smaller than a previously reported method, more work remains to be done. The proposed algorithm should be tested on all classes of velocity profiles that the Delta-Lognormal expression can model, and it should be more thoroughly tested on real human movement data. This will be the goal of a follow up study.

ANNEXE C ARTICLE 3: COMBINING INTERVAL ARITHMETIC WITH THE BRANCH AND BOUND ALGORITHM FOR DELTA-LOGNORMAL PARAMETER EXTRACTION

Simon Pierre BOYOGUENO BIDIAS, Jean-Pierre DAVID, Yvon SAVARIA, Réjean
PLAMONDON

*Authors are with the Electrical Engineering Department, Polytechnique Montréal, Montréal,
H3C 3A7 CANADA.*

*simon-pierre.boyogueno-bidias@polymtl.ca, JPDavid@polymtl.ca, yvon.savaria@polymtl.ca,
rejean.plamondon@polymtl.ca*

***This paper is published by (IGS) on June 13 2019, for the International Conference on pattern
Recognition and Artificial Intelligence Workshop on the Lognormality: (IGS) Cancún Mexico
2019.***

Abstract. In this paper, an interval arithmetic approach is proposed to compute guaranteed enclosures of the sets of points that bound a multivariate Delta-Lognormal function. The proposed algorithm is used in a branch and bound scheme, to extract parameters of the Delta-Lognormal function from ideal handwriting velocity profiles. Following the Kinematic Theory of rapid human movements, the basic timing properties of the neuromuscular system is represented by the seven parameters which characterize the delta lognormal function. The basic idea behind the proposed algorithm is to use the natural interval extension to sharply bound the range of the Delta-Lognormal function. The proposed new algorithm is described, tested (under various handwriting strokes) and compared with a previous algorithm developed for the same purpose. The numerical results show that the proposed algorithm gives better results in terms of speed and accuracy. This new algorithm, when embedded in a branch and bound scheme is useful for pointing movement analysis like gesture commands and handwriting strokes.

A. INTRODUCTION

Basic gestures like pointing movements are part of our daily interaction with the world. They can be defined as a coordinated set of actions performed by small muscles usually involving the synchronization of hands and fingers with the eyes. The complex levels of manual dexterity that humans exhibit is demonstrated in tasks controlled by the nervous system, to which they can be

attributed [87] [88]. These simple movements can be seen as the basic primitives [75] that are combined to produce more complex movements like handwriting and signatures. These movements were deeply studied by psychologists, neuroscientists, physicists, and computer scientists over the last century [89] [1]. Modelling them is the cornerstone to understand the basic principles of human motor control and its disorders [87].

In this perspective, the need to study the elementary properties of a single stroke for realistic analysis becomes key to understand how the motor control system accomplishes complex movements. The study of graphomotor behavior based on the Kinematic Theory of rapid human movements, [19], produces quantitative, precise, and objective measures of writing [90]. Among all existing handwriting generation models in the literature, the Delta-lognormal model (1), has proven over the years to be one of the most powerful to describe pointing movements because of its ability to reproduce, with minimal error, the velocity profile of a handwritten stroke. This model derived from the Kinematic Theory of rapid human movements, [12], [19], [14] considers a single stroke as a pen tip movement with a velocity profile expressed as in equation (C.1).

$$\Delta\Lambda(t; p) = D_1\Lambda(t; t_0, \mu_1, \sigma_1^2) - D_2\Lambda(t; t_0, \mu_2, \sigma_2^2) \quad (\text{C.1})$$

Where

$$\Lambda(t; t_0, \mu, \sigma^2) = \begin{cases} \frac{1}{\sigma\sqrt{2\pi}(t-t_0)} \exp^{-\frac{(\ln(t-t_0)-\mu)^2}{2\sigma^2}}, & \text{for } t_0 < t, \\ 0, & \text{elsewhere.} \end{cases} \quad (\text{C.2})$$

With $\sigma > 0; \mu, t_0 \in \mathbb{R}$

is the lognormal impulse response function. In this theory, a stroke is produced by a synergy of two neuromuscular systems: the agonist, acting in the direction of the movement and the antagonist, acting in the opposite direction. The agonist and antagonist are lognormal convolved with their corresponding input commands, represented in equation (C.1) by the subscripts 1 and 2 respectively. The parameters in equation (C.3) are defined as follows:

- t_0 : represents the system activation time.

- $\mu_1, \sigma_1, \mu_2, \sigma_2$ correspond to the timing properties (on a logarithmic scale) of the two neuromuscular commands
- D_1 and D_2 are the input amplitudes of the agonist and antagonist neuromuscular systems respectively.

The study of such strokes relies on the extraction of the parameters that characterize their velocity profile [19] [91] [14]. This problem can be achieved by minimizing the nonlinear least square function f (C.3), that maximizes the signal to noise ratio **SNR** (C.4)

$$f = \int (v_t(t) - \Delta\Lambda(t; p))^2 dt \quad (\text{C.3})$$

$$SNR = 10 \log \left(\frac{\int v_t^2(t) dt}{f} \right) \quad (\text{C.4})$$

Many algorithms have been developed in the past decade following this idea. Although the parameter extractors proposed in [48], [22], [92] show good data fitting, none of them can guarantee the optimality of the solution obtained. To address this need, a first branch and bound algorithm was developed in [50]. This algorithm is based on a Delta-Lognormal envelope which bounds all the possible Delta-Lognormal parameters within a subspace, and then computes the error between the signal and its nearest envelope. This method, even though it is promising, suffers from long computation time, very large memory requirements and lower bounding problems to extract a solution. In this paper, we propose a new algorithm, based on interval arithmetic, to solve the boundaries problems of the extractor proposed in [50] and show the effectiveness of the proposed algorithm when combined in a branch and bound scheme to extract model parameters from ideal velocity profile data.

The rest of the paper is organized as follows. Section 2 reviews some theoretical aspects of interval arithmetic and presents an algorithm for computing verified enclosures of the bound of the objective function. The uses of this algorithm in a branch and bound scheme will also be highlighted. Section 3 presents the numerical results and performance comparisons with [50] and [22]. Section 4 proposes possible future research and summarizes the main contributions of this paper.

B. ALGORITHM FOR COMPUTING THE UPPER AND LOWER BOUND OF THE DELTA-LOGNORMAL FUNCTION

Some definitions and notations need to be presented to support the proposed algorithm. We denote by $I = [a, b] = \{x \in \mathbb{R}: a \leq x \leq b\}$ the set of real numbers and one dimensional closed intervals. In this paper, an interval will be denoted by a variable in upper case while the set of values contained in the interval will be denoted by a lower case. Thus $X = [\underline{x}, \bar{x}]$ is an interval variable where \underline{x} and \bar{x} represent the left and right end points of X respectively. The width of X is defined and denoted by $w(X) = \bar{x} - \underline{x}$ and the midpoint is given by $m(X) = \frac{1}{2}(\bar{x} + \underline{x})$.

We denote by $\mathbf{X}^I = (X_1, X_2, \dots, X_n)$ the set of n dimensional interval vectors or hyper box such that:

$$\mathbf{X}^I = (X_1, X_2, \dots, X_n) = ([\underline{x}_1, \bar{x}_1], [\underline{x}_2, \bar{x}_2], \dots, [\underline{x}_n, \bar{x}_n]).$$

The width and the midpoint of an interval vector are defined as:

$$w(\mathbf{X}^I) = \max_i w(X_i^I) \text{ and } m(\mathbf{X}^I) = (m(X_1), m(X_2), \dots, m(X_n)).$$

We call F an inclusion function of $f: \mathbb{R}^N \rightarrow \mathbb{R}$, the interval extension of f , such that for $x \in X \Rightarrow f(x) \subseteq F(X)$. $f(X)$ is the real range of f on X and $f(X) \subseteq F(X)$.

Now that we have these definitions and notations, it is possible to use interval arithmetic [58], [59] to bound a function defined by a mathematical expression such as the Delta-Lognormal function.

1. INTERVAL ARITHMETIC

In this section, we introduce the interval tools needed to derive the bounding operation of equation (C.3). For more details, we refer the reader to [59], [58].

Theorem (fundamental theorem). Let $F(X_1, X_2, \dots, X_n)$ be the natural interval extension of $f(x_1, x_2, \dots, x_n)$ then

$$f(X_1, X_2, \dots, X_n) \subseteq F(X_1, X_2, \dots, X_n).$$

And for all intervals,

$$\text{if } Y_k \subset X_k \text{ for } k = 1, \dots, n, \text{ then } f(Y_1, Y_2, \dots, Y_n) \subseteq F(Y_1, Y_2, \dots, Y_n).$$

Where

$$f(Y_1, Y_2, \dots, Y_n) = \{ f(x_1, x_2, \dots, x_n) : x_k \in Y_k \text{ for } k = 1, \dots, n \}$$

This theorem is due to Moore and extended by Hansen, the proof can be found in [59]. It shows how easy it is to bound the range of a function and make finding the solution to the global optimization problem possible. In the next sub-section, we are going to apply this theorem to the Delta-Lognormal function as a specific sequence of interval arithmetic operations to bound the range of equation (C.3) within an interval vector.

2. ALGORITHM FOR COMPUTING THE UPPER AND LOWER BOUNDS

To compute the bounds of equation (C.3) we first define the bounding space of its variables as:

$$\mathbf{P}^I = \{[\underline{d}_1, \bar{d}_1], [\underline{\mu}_1, \bar{\mu}_1], [\underline{\sigma}_1, \bar{\sigma}_1], [\underline{d}_2, \bar{d}_2], [\underline{\mu}_2, \bar{\mu}_2], [\underline{\sigma}_2, \bar{\sigma}_2], [\underline{t}_0, \bar{t}_0] \} \quad (\text{C.5})$$

Then, through the fundamental theorem, we apply the natural interval extension to it. This yield:

$$\begin{aligned} f(\mathbf{P}^I) \in F(\mathbf{P}^I) &= [\underline{F}(\mathbf{P}^I), \bar{F}(\mathbf{P}^I)] \\ &= \int (\nu_t(t) - \Delta\Lambda(t; \mathbf{P}^I))^2 dt \end{aligned} \quad (\text{C.6})$$

As provided by the theorem and the natural interval extension, the result of the right-hand side of equation (C.10) is an interval $[\underline{F}(\mathbf{P}^I), \bar{F}(\mathbf{P}^I)]$ that bounds f . To compute that result, we have broken down the last term $\Delta\Lambda(t; \mathbf{P}^I)$ in the integral of equation (C.10), into a unique finite sequence of interval arithmetic operations as proposed in [57].

Function_compute_A $([\underline{\mu}_1, \bar{\mu}_1], [\underline{\sigma}_1, \bar{\sigma}_1], [\underline{t}_0, \bar{t}_0], t)$

1. constant = $\sqrt{2\pi}$;
2. Compute equation (C.2) using the following list of expressions (';' are used as separators for brevity).

$$T_1 = [\underline{\sigma}_1, \bar{\sigma}_1]^2; \quad T_2 = t - [\underline{t}_0, \bar{t}_0];$$

$$T_3 = \ln(T_2); \quad T_4 = T_3 - [\underline{\mu}_1, \bar{\mu}_1];$$

$$T_5 = T_3^2; \quad T_6 = \frac{T_5}{T_1}; \quad T_7 = 0,5 \times T_6;$$

$$T_8 = -T_7; \quad T_9 = \exp(T_8);$$

$$T_{10} = \text{constant} \times [\underline{\sigma}_1, \bar{\sigma}_1]; \quad T_{11} = T_2 \times T_{10};$$

$$T_{12} = \frac{T_9}{T_{11}};$$

3. return T_{12} ;

Algorithm1: Compute Bound ($P_i^I, v_t(t), t, N$)

1. set the initial variables: $\Delta\Lambda[N], F = []; \Lambda_1[N]; \Lambda_2[N]$
2. $P = P_i^I = \{[\underline{d}_1, \bar{d}_1], [\underline{\mu}_1, \bar{\mu}_1], [\underline{\sigma}_1, \bar{\sigma}_1], [\underline{d}_2, \bar{d}_2], [\underline{\mu}_2, \bar{\mu}_2], [\underline{\sigma}_2, \bar{\sigma}_2], [\underline{t}_0, \bar{t}_0] \}.$
3. for $i=0$ to $N-1$
4. if $(t_i - [\underline{t}_0, \bar{t}_0]) > 0$
5. $\Lambda_1[i] = \text{Function_compute_}\Lambda([\underline{\mu}_1, \bar{\mu}_1], [\underline{\sigma}_1, \bar{\sigma}_1], [\underline{t}_0, \bar{t}_0], t_i);$
6. $\Lambda_2[i] = \text{Function_compute_}\Lambda([\underline{\mu}_2, \bar{\mu}_2], [\underline{\sigma}_2, \bar{\sigma}_2], [\underline{t}_0, \bar{t}_0], t_i);$
7. $\Delta\Lambda[i] = \Lambda_1[i] - \Lambda_2[i];$
- else
- $\Lambda_1[i] = 0; \Lambda_2[i] = 0; \Delta\Lambda[i] = 0;$
8. compute the integral
9. if $(t_i - [\underline{t}_0, \bar{t}_0]) > 0$
- $((v_t[0] - \Delta\Lambda[0])^2 + (v_t[N-1] - \Delta\Lambda[N-1])^2) \times 0.5$
- For $i=1$ to $N-1$
10. If $(t_i - [\underline{t}_0, \bar{t}_0]) > 0$
- $F += (v_t[i] - \Delta\Lambda[i])^2;$
- Else
11. $F += 0;$
12. return $\frac{F}{\text{sample frequency}};$

Algorithm1 starts by expressing f as a code list (see `Function_compute_` Λ) [57] where each arithmetic (\pm, \times, \div , etc.) and unary elementary functions are expressed with only one unique variable in their expression. In this way we ensure that, the desirable properties (inclusion isotonicity, etc.) of the fundamental theorem are satisfied. Another advantage of using this method is the sharpness of the bound that results from it. In this algorithm, the integral is computed numerically (see steps 8-11) as proposed in [73]. The condition in steps 4, 9 and 10 in Algorithm1 served to preserve the inclusion of the objective function (when computing the integral) to lie in the whole domain where it is defined, regardless of the subspace to be evaluated. This is necessary because our objective function (see equation C.3) is piecewise defined.

In summary the inputs for Algorithm1 are a box P , the ideal velocity profile $v_t(t)$, the sample time t , and the sample size N . From step 3 to 8, it computes the Delta-Lognormal function by calling `Function_compute_` Λ for each lognormal sample and then computes the integral to return the bounds on F . This is done in steps 8 to 11.

3. APPLYING ALGORITHM1 IN A GENERAL BRANCH AND BOUND SCHEME

The general idea behind the branch and bound technique is to bound sub-boxes of a feasible area in which we perform a search to get the sharpest one. The bounding operations can be carried out by operations such as in Algorithm1 and, if the bounds are not sharp enough, some sub-boxes are divided into smaller one according to a splitting rule. The procedure repeats until the optimal solution is obtained within the required accuracy. For our purpose, the objective function to be minimized is provided by equation (C.3) and the accuracy is fixed by equation (C.4) to an SNR = 100dB. To verify that the developed algorithm produces valid results, we replaced in [50] the computation of the lower and upper limits of the lognormal parameters by Algorithm1. We also changed the division rule of a subspace from three to two sub boxes.

The feasible area is restricted to a region of search where we know that the solution can be found. The values of the interval within the bounding box are chosen as wide as possible, to see how the algorithm will perform in finding the solution.

C. TETING THE ALGORITHM UNDER IDEAL CONDITIONS

1. TEST CONDITIONS

To test Algorithm1, we used the database proposed in [22] .This database is composed of 7000 Delta-Lognormal curves, grouped into categories. These ideal data are derived from human handwriting movements, previously extracted using the **IIX** algorithm presented in [50]. From this database, we randomly choose 300 ideal Delta-Lognormal velocity profiles, from Ca0 and Ca2 classes. Each of them is sampled at 200Hz to simulate the data collected with a digitizer [65] and satisfy a signal-to-noise ratio of 25dB with respect to the real (collected from real human movement with a digitizer) velocity profile. The implementation is done in C++ using our custom interval arithmetic library based on [58] and [74]. All computations are performed using rounded interval arithmetic in a 3.60 GHz Intel core (i7) with 32 Go of Ram.

2. RESULTS AND DISCUSSION

In this experience, a box is considered a solution and stored in the list of results if the maximum width or the quadratic error of the chosen box is less than 10^{-5} . That is the signal to noise ratio of the optimal box must be at least 100dB. Table C.1 presents the performance comparison in terms of accuracy between the proposed algorithm, [22] and [50]. As can be seen in table C.1, for the

class Ca0, our algorithm performs better than **Xzero** [22] and produces the same results as in [50]. Table C.2 shows the performance comparison in terms of processing time and number of boxes explored between the proposed algorithm and [50]. In [50], the algorithm is a branch and bound executed in parallel, on a 12 CPU node grid. As can be seen in table C.2 below. This algorithm requires a run time of about 50s to perform 11 iterations of its main loop while the one we propose requires only 702ms. On the other hand, the number of boxes needed to find the optimal solution can reach a million with their algorithm while ours requires at most 70 thousand boxes to extract the optimal solution.

Table C.1 Extraction of the exact value of the parameters for an SNR of >100db as ending criterion

Method			
Classes	Xzero	[50]	Proposed
Ca0 in %	95.2	100	100
Ca2 %	100	100	100

Table C.2 Performance comparison of the number of boxes explored to extract one velocity profile and the processing time required to perform 11 iterations in the main loop

Method		
Performances	[50]	Proposed
Number of Boxes	Min 800000	Max 70000
Processing time (in second)	50	0.702

3. CONCLUSION

This paper shows that a good bounding algorithm used in a branch and bound scheme can be used to extract the Delta-Lognormal function parameters of ideal velocity profile data with a lower computational cost. We have focused on the analysis of individual synthesized movements using the kinematic theory of rapid human movement. The developed algorithm that bounds the Delta-Lognormal function and guarantees the inclusion when used in a branch and bound for global optimization is illustrated and evaluated. The proposed algorithm exploits the natural interval extension and the fundamental theorem of interval arithmetic to compute the bounding operations of the Delta-Lognormal function. We also highlighted the use of that algorithm in a branch and

bound scheme and compared the performance of the parameter estimation with two previous algorithms developed for the same purpose. Although the proposed algorithm is promising in its ability to recover the global solution in a less time (in comparison to the previous one), more work remains to be done. Another step could be to test the algorithm on several classes of velocity profiles as provided by the Delta-Lognormal model as well as on real human movement data.

ANNEXE D ARTICLE 5: A NEW EXPERIMENTAL SET-UP TO RUN NEUROMUSCULAR TESTS

Nadir Faci <i>Electrical engineering</i> <i>Polytechnique</i> <i>Montréal</i> Montréal, Canada nadir.faci@polymtl.ca	Simon <i>PierrenBoyogueno</i> <i>Bidias</i> <i>Electrical engineering</i> <i>Polytechnique</i> <i>Montréal</i> Montréal, Canada simon- pierre.boyogueno- bidias@polymtl.ca	Réjean Lamondon <i>Electrical engineering</i> <i>Polytechnique Montréal</i> Montréal, Canada rejean.lamondon@polymtl.ca	Nicolas Bergeron <i>Scribens Project Inc</i> Montréal, Canada nicolas @scribensproject.co
-------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------

This paper is published by (cenparmi) on May 13 2018 for the International Conference on pattern Recognition and Artificial Intelligence: (ICPRAI) Montréal 2018.

The paper has been submitted on February 16, 2018.

Abstract—This article proposes an integrated and updated version of the human writing movement acquisition system **Sign@medic**. Based on the Wacom Cintiq technology, this new system takes advantage of all the features available in this tablet to integrate the visual and audio stimulus generator of the **Sign@medic** software into the tablet. The resulting embedded system which combined the acquisition system and the audio-visual stimulus in a unique tablet environment, allows users to work directly on the screen of the Cintiq 13HD tablet. The methodology, as well as a comparison between the old and new system is described and presented. This new system provides an efficient and affordable tool for researchers involved in various domains of pattern recognition and artificial intelligence, relying on studies of human movements.

Keywords— Sigma-Lognormal model, Kinematic Theory of rapid human movement, kinematic analysis, fine motor control, kinematic analysis, handwriting

A. INTRODUCTION

Human movement can be very complex, and their studies in different areas of medicine and human science has been intensively used. In neurosciences handwriting strokes, which constitute a specific class of rapid human movement, are used to study neurodegenerative processes involved by diseases such as Parkinson and Alzheimer's [15] [5]. Moreover, the early diagnosis of cerebral lesions appears to be possible by detecting slight deviations from the norm which are not evident by simple visual inspection.

Many of these examples emphasize the need of building robust and easily workable equipment for movement analysis. Among the various sets present in the literature and the market, the Sign@medic acquisition system developed in the Scribens laboratory has been found to be successful in acquiring handwriting movement most accurately. According to the Kinematic Theory of rapid human movement these strokes constitute the elementary movement necessary to understand how the motor control deal in performing complex movement. Over the last fifteen years, all the neuromuscular studies performed at Scribens laboratory have been done using the Sign@medic acquisition devices, based on Wacom Intuos2 technology under Windows XP operating system. This was done to make sure that we could get comparative results from one experiment to another. Such a system is no longer adequate and is about to be unsupported by the manufacturers.

This paper proposes an integrated and updated version of the acquisition device Sign@medic based on Wacom Cintiq technology. This new system takes advantage of all the recent innovations offered by Wacom Cintiq 13HD technology to provide a versatile and comfortable acquisition system. Allowing the users to work directly on the screen of this compact tablet. Excellent color and resolution, professional ergonomics and pressure-sensitive control have all come together in this affordable model.

The rest of the paper is organized as follows: section B reviews the old acquisition device and its main features. Section C introduces some gap of the old equipment and proposed a method to relieve these deficiencies as well as some new features accompanying the new system. The global system including the lognormal parameter extraction will also be presented. Section D presents numerical results and performance comparisons with the previous system. Section E summarizes the main contributions of this paper and possible future research.

B. DESCRIPTION OF THE PREVIOUS SYSTEM

Figure D.1 represents the bloc diagram of the old acquisition system. It is composed of the homemade software Sign@medic, which managed the synchronization between the tablet digitizer and the stimulator. A Wacom Intuos2 is used as tablet digitizer to record the coordinates of the pen tip movement, while the subject is executing a handwriting movement. This tablet records the data at a sampling rate of 200 Hz with a spatial resolution of 100 lines per millimetre. Each movement performed on the tablet follows a test protocol based on different neuromuscular tests [66]. . Guide

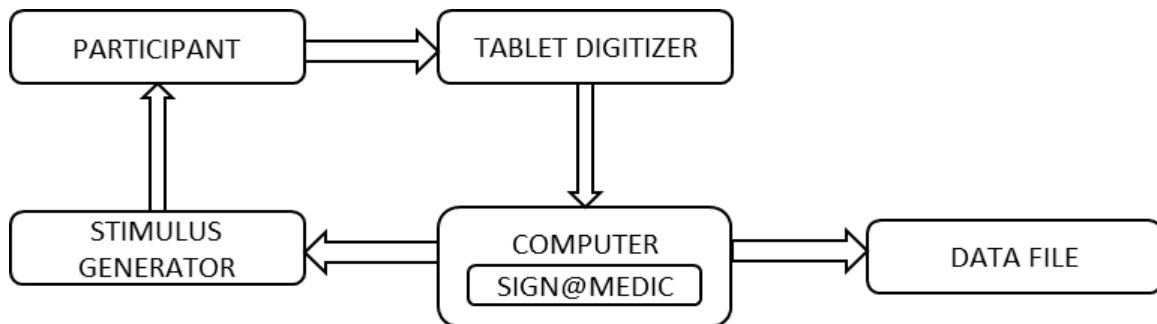


Figure D.1 The bloc Diagram of the old acquisition system

sheets are placed under the transparent plastic fold of the tablet to reduce friction and indicate the starting point and the target zone to subjects. The beginning of each test is given by the stimulator, a proprietary design. Two types of stimulus are used: a visual and an audio. The visual stimulus is displayed by a matrix of 8x10 light-emitting diodes (LED) with a delay of 1 ms [66]. The auditory stimulus is a 1 KHz beep with duration of 500 ms. The stimulator is linked to the computer by a serial to parallel converter. As it can be seen in Figure D.1, the system takes a handwriting movement as an input and return information about that movement, including the Cartesian coordinates of the stiletto, the pressure, etc. Despite its effectiveness, this system suffers from compatibility with new Windows operating system.

The parallel port used by the stimulus generator is outdated, so the stimulus device cannot be connected to modern computers anymore. Moreover, the stimulus generator is cumbersome. Furthermore, and the Wacom Intuos2 tablet is no longer supported by the manufacturer. These problems point the need to update the old system so it would become compatible with modern computers.

C. DESCRIPTION OF THE NEW SYSTEM

1. TRANSFER PROCESS

The goal of this project is to update the old acquisition system such that it will be ergonomic and compatible with the Windows 10 operating system. To achieve this goal, we have designed a graphical interface that displays the different functionality of the visual stimulator on a Wacom Cintiq 13HD tablet. In addition, the various neuromuscular tests present in our database have to be included. The interface is implemented in C++ using the Microsoft foundation class library and the communication is done by our custom software employing Wacom drivers and libraries. The

interface is divided in three sections, two sides sections displaying the visual stimulator and a middle section used to display the guide sheet and the stroke. Since the Cintiq does not have a transparent plastic fold as the Intuos2, the guide sheets were resized to fit into the active area of the Cintiq 13HD. Those guide sheets were included directly as a C++ function in the software. A new button is added in the interface to switch between right-handed and left-handed guide sheets. Finally, a new feature exploiting the features offered by the Cintiq 13HD is added to the software. It consists of showing, in real time, on the screen of the tablet the stroke being executed by a subject.

2. DESCRIPTION OF THE ACQUISITION SYSTEM

As presented in Figure D.2, the new acquisition system is a compact Update version of the old one. This system includes the stimulus generator in the Sign@medic software and will use the Wacom Cintiq 13HD as the stimulator tools. This makes the overall system composed of only a computer containing the homemade software, and the Wacom Cintiq 13HD tablet. In addition, the tablet screen acts as an additional monitor for the computer. The visual stimulus is displayed on the tablet,

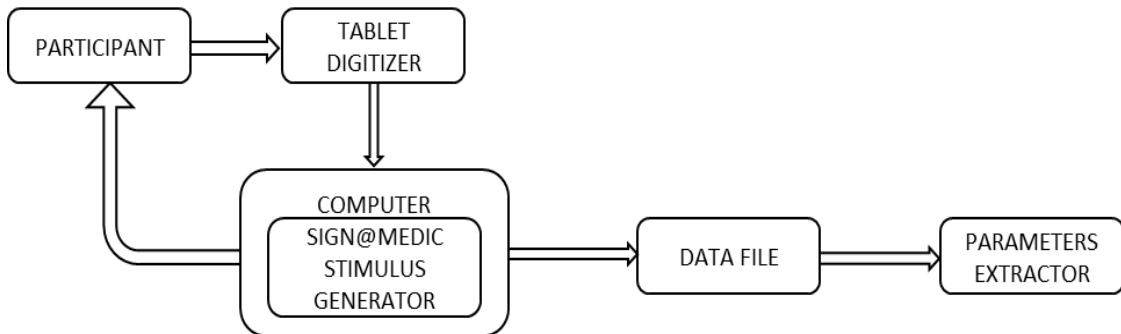


Figure D.2 The bloc diagram of the global system including the new acquisition system

at a response rate of 25 ms. In regard of the audio stimulus, the speakers connected to the computer emits an audio cue at 1 KHz during 500 ms. The stroke made by the participant is displayed on the tablet, so that the operator could see his movement. The Wacom Cintiq 13HD record data at 200 Hz with 200 lines per millimetre spatial resolution. The new system take as an input a handwriting movement and deliver as an output a data file including the information about the movement.

3. DESCRIPTION OF THE GLOBAL SYSTEM

Figure D.2 presents the global system which is composed of the new acquisition software and the lognormal parameters extractors [22] [21] [50]. Those extractors were developed by the

Scribens laboratory following the Kinematic Theory of rapid movement [93] [12] [19] [20] [14]. The global system is able to acquire data related to fast stroke movement executed by a participant. After that, it allows extraction of the lognormal parameters by using the different extractors. In conclusion, the global system takes a fast stroke movement as an input and gives lognormal parameters as an output.

D. COMPARATIVE EVALUATION

1. ADVANTAGES AND DISADVANTAGES OF THE NEW SYSTEM IN COMPARED TO THE OLD ONE

The major advantage of the new system is that all the material everything is integrated in the software. Foremost, the guide sheets are integrated on the software, and they can be changed automatically while changing the test. There is also the option to choose between right-handed and left-handed guide sheet. The stimulator generator is also integrated in the software. The device used before to generate stimulation is now outdated and it is not needed anymore. Moreover, the stroke trajectory can be seen on the screen of the tablet while being executed. This feature will help the operator to judge if the stroke is well executed. After that, the operator can decide to reject it or not. Finally, the fact that the stimulus is not displayed on the previous hardware device anymore, and only displayed on the tablet, should give more focus to the participant. Indeed, the participant will only look at the tablet screen and will not need to move their head to see what is happening with the stimulator device anymore. The participant will only look at the tablet because all they need is to execute the stroke displayed on the tablet. The new tablet has a better spatial resolution than the old one.

On the other hand, the new system presents some disadvantage as compared to the old one. The stimulator device had a 1 ms response time for displaying the visual and audio stimulus. With the new system, the tablet has response rate of 25 ms to display the visual cue. In the case of the sound cue, the response rate depends of the computer to which the tablet is connected. Knowing that human reaction to visual and auditory stimuli is respectively around 247 ms (18.54) and 228 ms (16.49) for young adults [94] [95] [96], the difference between the response rate of the new and the old system should not influence the result. The limit that it put on the experimental protocol is that the resolution for observing changes in neuromotor control, for example the changes in t_0 will have to be greater than 25 ms instead of 1ms to be observed.

2. METHODOLOGY TO COMPARE THE TWO SYSTEMS.

To compare the two systems, we requested a subject to draw fast strokes movement for five days in a row. Each system was connected to different computers. The computer used for the new system is slower. Each day, the subject had to draw 30 valid strokes for the four following tests: visual stimulus test on the new system, auditory stimulus test on the new system, visual stimulus test on the old system and auditory stimulus test on the old system. The visual stimulus test was always executed before the audio one. Each day, we changed the order, the first day, the tests were executed on the new system before doing it on the old one. The second day, the tests were executed on the old system before doing it on the new one. The same process was done the other days. A new guide sheet was used to perform those tests. The Figure D. 3 depicts the new guide sheet used to perform the test while the visual cue is displayed on the tablet screen.

3. THE SIGMA-LOGNORMAL MODEL

After recording the data, the Sigma-Lognormal model was used to extract the parameters from the original data. In this purpose, Script Studio was used [21] [43] [97] [98]. This software extract 8 parameters linked with the Sigma-Lognormal model. In addition, the SNR/NbLog was calculated. The first three parameters describe the global state of the neuromotor system [12] [19] [14] [99]:

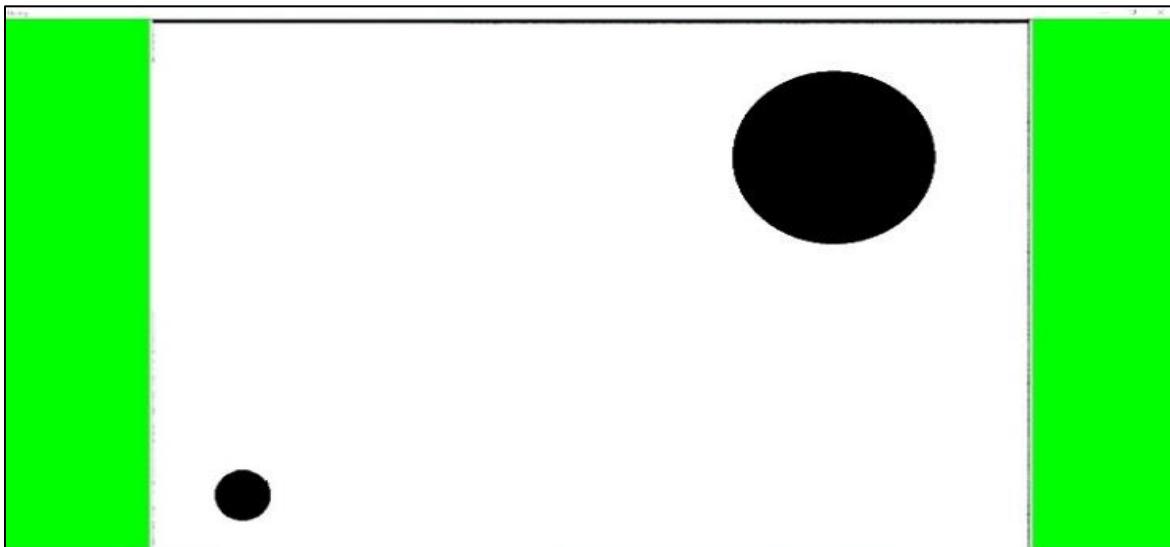


Figure D.3 The interface displayed on the screen of the tablet including the guide sheet on the center with the visual cue on each side

- SNR: The signal-to-noise ratio between the original speed profile and the reconstructed speed profile, computed in decibels (dB). This is a measure of the quality of the sigma-lognormal reconstruction. Higher SNR indicates better reconstruction.
- NbLog: The number of lognormal functions required to reconstruct the signal. This parameter represents fluidity of movement by the participant. Higher nbLog, indicates less fluid movement.
- SNR/NbLog: SNR/nbLog is used as a performance criterion. The capacity to reconstruct a movement's speed profile with lognormals can be interpreted as an indicator of the quality of motor control since the lognormal speed profile corresponds to perfect motor control [100]. Higher SNR/nbLog indicates that the movement is closer to ideal lognormal behavior predicted by the lognormality principle.

The other parameters are indexes of the internal functioning of the neuromotor system. The four following parameters represent the neuromotor action plan:

- t_0 : Time required (in seconds) for the brain to produce a motor command. t_0 should not be confused with the moment a sound cue is emitted ($t = 0$) nor with the response time (RT) measured by moment of movement onset. t_0 is the moment when a population of neurons sends a motor command. In other words, t_0 occurs after the start cue, given that the central nervous system must perceive the sound cue before sending the command, but before RT, given that a delay is necessary for the motor command to be transmitted and for enough muscle cells to be recruited to begin the movement effectively [99]. Longer t_0 are indicative of longer delays before a motor command is emitted and, therefore, longer time to perceive the stimulus and prepare the command. t_{0i} is the time required for the first motor command to be produced by the brain.
- D: The amplitude of the movement associated with each motor command, in millimeters. Greater D indicates greater amplitude of the movement associated with a command. D_{tot} is the total amplitude of the movement.

The two following parameters are intrinsic characteristics of each lognormal and related to motor program action plan [14] [76].

- θ_s : The starting angular direction of each stroke in radian.

- θ_e : The ending angular direction of each stroke in radian.

The other parameters are related to motor program execution:

- μ : Logtime delay, the time required to reach the middle of movement distance on a logarithmic scale. It can be associated with the general response speed of the neuromuscular system to a command.
- σ : Logresponse time, the response time on a logarithmic scale of the neuromuscular system's response to a command. Sigma (σ) is used to compute an estimate of movement duration, it also reflects the asymmetry of the lognormal.

4. STATISTICAL ANALYSIS

After extracting the parameters, the data for each day were regrouped together by test. Indeed, all the data recorded with new system using the visual stimulus were regrouped together, and so forth. A t-test for independent measures was conducted on the parameters extracted by Script Studio from all the strokes made. Bonferroni correction for multiple comparisons was used to determine if there is a significant difference. P-value must be less than 0.00625 ($\alpha = 0.05/8 = 0.00625$) to have a statistically significant difference between the two groups.

5. PRELIMINARY RESULTS AND DISCUSSION

Table D.1 Result of t-test for independent group between the new and the old system with the visual cue

Parameters	New system M(S.D)	Old system M(S.D)	t-value	df	p-value
SNR	30,44 (2,46)	29,61 (2,67)	2,87	309	0,0044 ^a
NbLog	2,06 (0,36)	2,00 (0,63)	0,99	309	0,3225
SNR/NbLog	15,25 (3,15)	16,37 (5,48)	-2,20	309	0,0283 ^b
t_0	0,24 (0,16)	0,23 (0,16)	0,31	629	0,7592
t_{0i}	0,22 (0,07)	0,21 (0,10)	0,59	309	0,5564
D	118,18(95,89)	115,24(94,35)	0,39	629	0,6973
D_{tot}	192,92 (8,59)	191,02 (8,08)	2,01	309	0,0455 ^b
μ	-1,54 (0,54)	-1,54 (0,56)	0,04	629	0,9721
σ	0,24 (0,14)	0,23 (0,15)	0,84	629	0,4002
θ_e	-0,31 (4,58)	-0,73 (4,64)	1,15	629	0,2506
θ_e	-0,60 (4,58)	-1,08 (4,64)	1,30	629	0,1930

Table D.2 Result of t-test for independent group between the new and the old system with the sound cue

Parameters	New system M (S.D)	Old system M (S.D)	t-value	df	p-value
SNR	30,99 (2,64)	29,19 (2,38)	6,28	306	0,0000 ^a
NbLog	2,01 (0,41)	2,03 (0,53)	-0,37	306	0,7106
SNR /NbLog	16,06 (3,73)	15,35 (4,30)	1,55	306	0,1212
t ₀	0,22 (0,14)	0,18 (0,14)	3,64	621	0,0003 ^a
t _{0i}	0,17 (0,06)	0,14 (0,07)	4,06	306	0,0001 ^a
D	118,71(96,14)	115,27(94,71)	0,45	621	0,6528
D _{tot}	194,15 (9,03)	190,46 (9,12)	3,57	306	0,0004 ^a
mu	-1,62 (0,51)	-1,66 (0,57)	0,97	621	0,3307
sigma	0,25 (0,12)	0,25 (0,14)	-0,54	621	0,5928
tetaD	-0,94 (4,29)	-0,01 (4,40)	-2,68	621	0,0075 ^b
tetaf	-1,26 (4,29)	-0,30 (4,40)	-2,76	621	0,0060a

^a.Significant with Bonferroni corrections for multiple comparisons (p < 0.00625)

b.Significant at p < 0.05

This preliminary test showed that there a significant difference, with the Bonferroni correction, on the SNR between the two system in both visual and audio stimulus. In both cases, the SNR mean is higher for the new system. This could be caused by the higher spatial resolution of the new Wacom Cintiq 13HD as compared to the Wacom Intuos2. The Dtot showed a significant difference, without the Bonferroni correction, between the two systems in both visual and audio stimulus. In both cases, the Dtot mean is higher for the new system. The subject stipulate that the Wacom Cintiq 13HD has a surface with less friction than the surface of the Wacom Intuos2. Drawing on the new tablet is smoother than the old one, so the subject have more difficulties to stop their movements, which led to higher Dtot. For the auditory stimulus, there is a significant difference, with the Bonferroni correction, in the t0 parameters. The t0 mean is lower for the old system compared to the new one. This could be caused by the fact that the computer used for the new system is slower than the other one. It could also be caused by the fact that the old stimulus generator had a response time of 1 ms, and that the response time of the new integrated stimulator depend of the computer. Further investigations are under way to clarify this point.

E. CONCLUSION

This project consisted of transferring the old Sign@medic system based on a Wacom intuos2 tablet under windows XP operating system to the new Wacom Cintiq 13HD to operate under windows

10. The new system have the advantage of including the external stimulus generator device and the guide sheets in the software to make the system more compact, ergonomics and user friendly. On the other side, the new system may be slower than the old one. Eventually for more concrete result, the two system should be tested by a drawing robot to remove the human impact. Finally, the system proposed in this paper could be used in different research projects dealing with the human neuromotor system.

Acknowledgements

This project was supported by NSERC grant RGPIN-2015-06409 to Réjean Plamondon and MESI and Univvalor support to Scribens laboratory.

ANNEXE E ARTICLE 6: AN INTERACTIVE TABLET-BASED SYSTEM TO RUN NEUROMUSCULAR TESTS

Nadir Faci¹, Simon Pierre Boyogueno Bidias, Ph.D.¹, Réjean Plamondon, Ph.D.¹ and Nicolas Bergeron²

¹Polytechnique Montréal,² Scribens Project Inc

nadir.faci@polymtl.ca, simon-pierre.boyogueno-bidias@polymtl.ca,,

rejean.plamondon@polymtl.ca, nicolas@scribensproject.com

This paper is published in the book « The Lognormality Principle and its applications in E-Security, E-learning and E-Health » on December 10, 2020 for the International Conference on Pattern Recognition and Artificial Intelligence: Workshop on the Lognormality Principles and its Applications (ICPRAI) Montréal 2018.

The paper has been submitted on February 16, 2018.

Abstract—This chapter proposes an integrated and updated version of the human writing movement acquisition system Sign@medic. Based on the Wacom Cintiq technology, this new system takes advantage of all the features available in this tablet to integrate the visual and audio stimulus generator of the Sign@medic software into the tablet. The resulting embedded system which combined the acquisition system and the audio-visual stimulus in a unique tablet environment, allows users to work directly on the screen of the Cintiq 13HD tablet. The methodology, as well as a comparison between the old and new system is described and presented. This new system provides an efficient and affordable tool for researchers involved in various domains of pattern recognition and artificial intelligence, relying on studies of human movements.

Keywords— Sigma-Lognormal model, Delta-Lognormal model, Kinematic Theory of rapid human movement, kinematic analysis, fine motor control, kinematic analysis, handwriting.

A. INTRODUCTION

Human movements can be very complex, and their studies in different areas of medicine and human science has been intensively used. In neurosciences, handwriting strokes, which constitute a specific class of rapid human movements, are used for example to study neurodegenerative

processes involved in diseases such as Parkinson and Alzheimer's [15] [5] [101] [102] [103] [104] [5], and for example child with ADHD [105] [106] [107] [18] [108]. Moreover, the early screening of cerebral lesions appears to be possible by detecting slight deviations from the norm which are not evident by simple visual inspection.

Many of these examples emphasize the need of building robust and easily workable equipment for movement analysis. Among the various sets present in the literature and market, the Sign@medic acquisition system developed at Scribens laboratory decades ago has been found to be successful in acquiring handwriting movements most accurately. According to the Kinematic Theory of rapid human movement these strokes constitute the elementary movements necessary to understand how the motor control deal in performing complex movements. Over the last fifteen years, all the neuromuscular studies performed at Scribens laboratory have been done using the Sign@medic acquisition devices, based on Wacom Intuos2 technology under Windows XP operating system. This was done to make sure that we could get comparative results from one experiment to another. Such a system is no longer adequate and is about to be unsupported by the manufacturers.

This paper proposes an integrated and updated version of the Sign@medic acquisition device based on the Wacom Cintiq technology. This new system takes advantage of all the recent innovations offered by Wacom Cintiq 13HD technology to provide a versatile and convenient acquisition system, allowing users to work directly on the screen of this compact tablet. Excellent colour displays and screen resolution, professional ergonomics and pressure-sensitive controls all come together in this affordable model.

The rest of the paper is organized as follows: section 2 reviews the old acquisition device and its main features. Section 3 reports some drawback of the old equipment and proposed a method to relieve these deficiencies as well as some new features incorporated in the new system. The global system including the lognormal parameter extraction is also described. Section 4 presents numerical results and performance comparisons with the previous system. Section 5 summarizes the main contributions of this paper and opens the door to future research.

B. DESCRIPTION OF THE PREVIOUS SYSTEM

Figure E.1 represents the bloc diagram of the old acquisition system. It is composed of the in house software Sign@medic, which managed the synchronization between the digitizer and the stimulator. A Wacom Intuos2 is used as a digitizer to record the coordinates of the pen tip

movement, while the subject is executing a handwriting movement. This tablet records the data at a sampling rate of 200 Hz with a spatial resolution of 100 lines per millimetre. Each movement performed on the tablet follows a test protocol based on different neuromuscular tests [66]. Guide sheets are placed under the transparent plastic fold of the tablet to reduce friction, and these indicate the starting point and the target zone of an experiment to subjects. The beginning of each test is initiated by the stimulator, a proprietary design. Two types of stimulus are used: a visual and an audio. The visual stimulus is displayed by a matrix of 8x10 light-emitting diodes (LED) with a delay of 1 ms [66]. The auditory stimulus is a 1 KHz beep with duration of 500 ms. The stimulator is linked to the computer by a serial to parallel converters. As it can be seen in Figure E.1, the system takes a handwriting movement as input and returns information about that movement, including the Cartesian coordinates of the stilus, the pressure, etc. Despite its effectiveness, this system suffers from compatibility with new Windows operating systems.

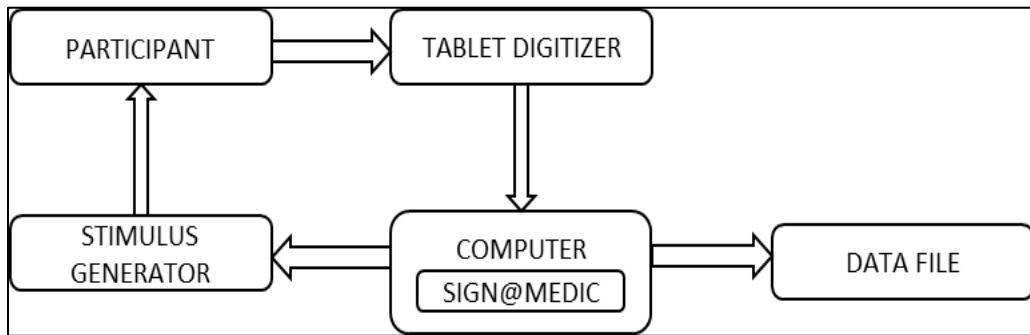


Figure E.1 The bloc diagram of the old acquisition system

The parallel port used by the stimulus generator is outdated, so the stimulus device cannot be connected to modern computers anymore. Moreover, this stimulus generator is cumbersome. Furthermore, the Wacom Intuos2 tablet is no longer supported by the manufacturer. These problems point out the need to update the old system to make it compatible with modern computer environments.

A. DESCRIPTION OF THE NEW SYSTEM

1. TRANSFER PROCESS

The goal of this project is to update the old acquisition system such that it will be ergonomic and compatible with the Windows 10 operating system. To achieve this goal, we have designed a

graphical interface that displays the different functionalities of the visual stimulator on a Wacom Cintiq 13HD tablet. In addition, the various neuromuscular tests present in our database are to be included as well as provisions for adding any new tests. The interface is implemented in C++ using the Microsoft foundation class library and the communication is done by our custom software employing Wacom drivers and libraries. The screen interface is divided in three sections, two side sections displaying the visual stimulator and a middle section used to display the guide sheet and the handwriting trajectories. Transparent plastic fold is not needed any longer, but the guide sheets, initially designed to match with the Intos2 dimensions (250x180 mm), were homothetically resized to fit into the active area of the Cintiq 13HD (220x160 mm). Those guide sheets were included directly as a C++ function in the software. A new button has been added to the interface to switch between right-handed and left-handed guide sheets. Finally, a new feature exploiting the electronic ink features offered by the Cintiq 13HD has been added to the software. It consists of showing, in real time, on the screen of the tablet the handwriting or drawing patterns being executed by a subject.

2. DESCRIPTION OF THE ACQUISITION SYSTEM

As presented in Figure E.2, the new acquisition system is a compact, updated and ergonomic new design elaborated from the old one. This system replaces the stimulus generator and the paper guide sheets of the Sign@medic software using the Wacom Cintiq 13HD as an integrated stimulator and data acquisition tool. This makes the overall system composed of only a computer containing an in house software, and the Wacom Cintiq 13HD tablet interface. In addition, the tablet screen acts

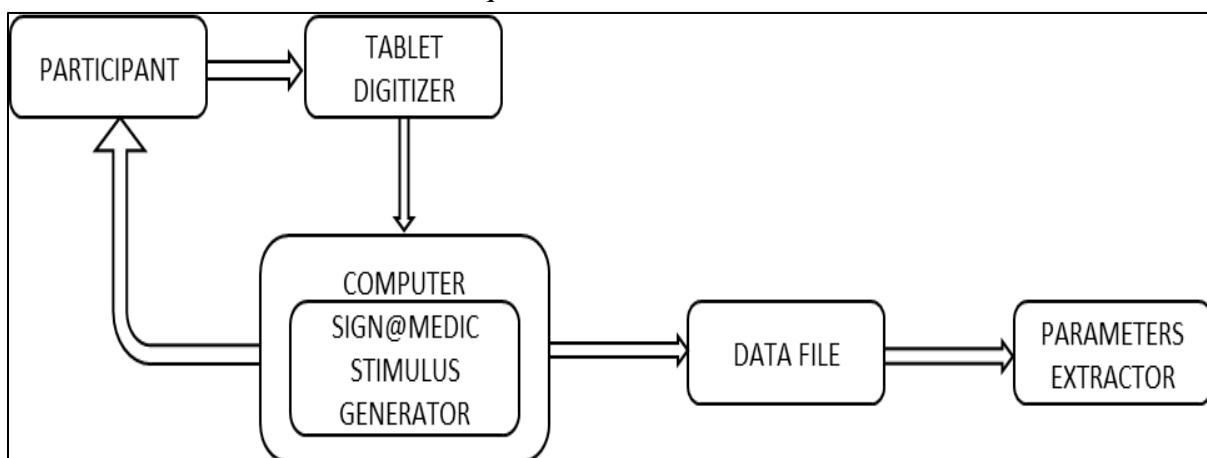


Figure E.2 The bloc diagram of the global system including the new acquisition system called the Lognometer

as an additional monitor for the computer. The visual stimulus is displayed on the tablet, at a response rate of 25 ms. In regard of the audio stimulus, the speakers connected to the computer emits an audio cue at 1 KHz during 500 ms. The trajectory made by the participant is displayed on-line on the tablet, which allows the test operator to see directly the subject responses. The Wacom Cintiq 13HD records data at 200 Hz with a 200 lines per millimetre spatial resolution. The new system takes as an input a handwriting movement and delivers as output a data file including the information about this movement.

3. DESCRIPTION OF THE GLOBAL SYSTEM

Figure E. 2 presents the global system which is composed of the new acquisition software and the lognormal parameters extractors [22] [21] [50]. Those extractors were developed at Scribens laboratory to take advantage of the Kinematic Theory of rapid movements [92] [12] [109] [20] [14]. The global system is able to acquire data related any simple or complex 1D or 2D movements executed by a participant. After that, it allows the extraction of the lognormal parameters by using the different extractors. In conclusion, the global system takes a single or a sequence of stroke movements as input and generates lognormal parameters lists as output, making the whole system acting like a lognometer.

B. COMPARATIVE EVALUATION

1. ADVANTAGES AND DISADVANTAGES OF THE NEW SYSTEM COMPARED TO THE OLD ONE

The major advantage of the new system is that all the material everything is integrated in the software. Foremost, the guide sheets are integrated on the software, and they can be changed automatically while changing the test. There is also the option to choose between right-handed and left-handed guide sheet. The stimulator generator is also integrated in the software. The device used before to generate stimulation is now outdated and it is not needed anymore. Moreover, the stroke trajectory can be seen on the screen of the tablet while being executed. This feature will help the operator to judge if the stroke is well executed. After that, the operator can decide to reject it or not. Finally, the fact that the stimulus is not displayed on the previous hardware device anymore, and only displayed on the tablet, should give more focus to the participant. Indeed, the participant will only look at the tablet screen and will not need to move their head to see what is happening with the stimulator device anymore. The participant will only look at the tablet because all they

need is to execute the stroke displayed on the tablet. The new tablet has a better spatial resolution than the old one.

On the other hand, the new system presents some disadvantage as compared to the old one. The stimulator device had a 1 ms response time for displaying the visual and audio stimulus. With the new system, the tablet has response rate of 25 ms to display the visual cue. In the case of the sound cue, the response rate depends on the computer to which the tablet is connected. Knowing that human reaction to visual and auditory stimuli is respectively around 247 ms (18.54) and 228 ms (16.49) for young adults [94] [95] [96],, the difference between the response rate of the new and the old system should not influence the result. The limit that it put on the experimental protocol is that the resolution for observing changes in neuromotor control, for example the changes in t_0 will have to be greater than 25 ms instead of 1ms to be observed.

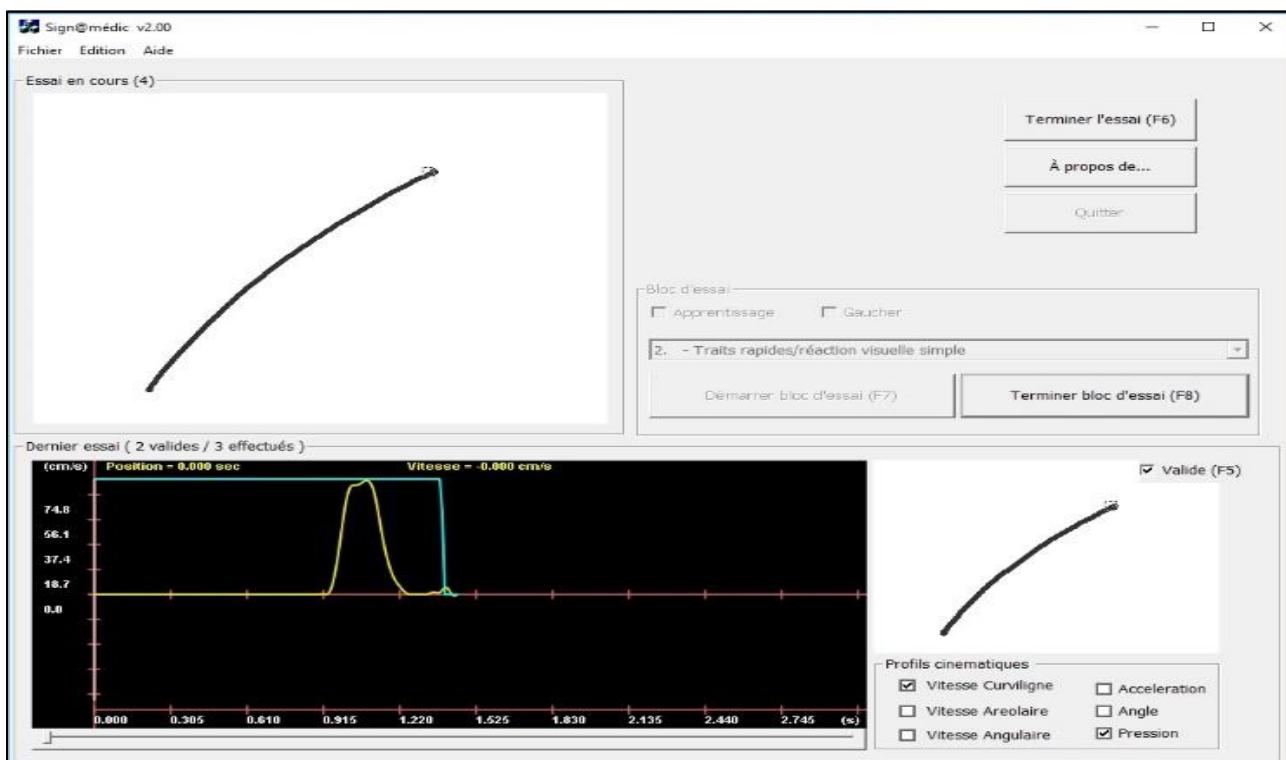


Figure E.3 The user interface displayed on the computer monitors shown to the user

2. METHODOLOGY TO COMPARE THE TWO SYSTEMS

To compare the two systems, we requested a subject to draw fast strokes movement for five days in a row. Each system was connected to different computers. The computer used for the new system is slower. Each day, the subject had to draw 30 valid strokes for the four following tests: visual

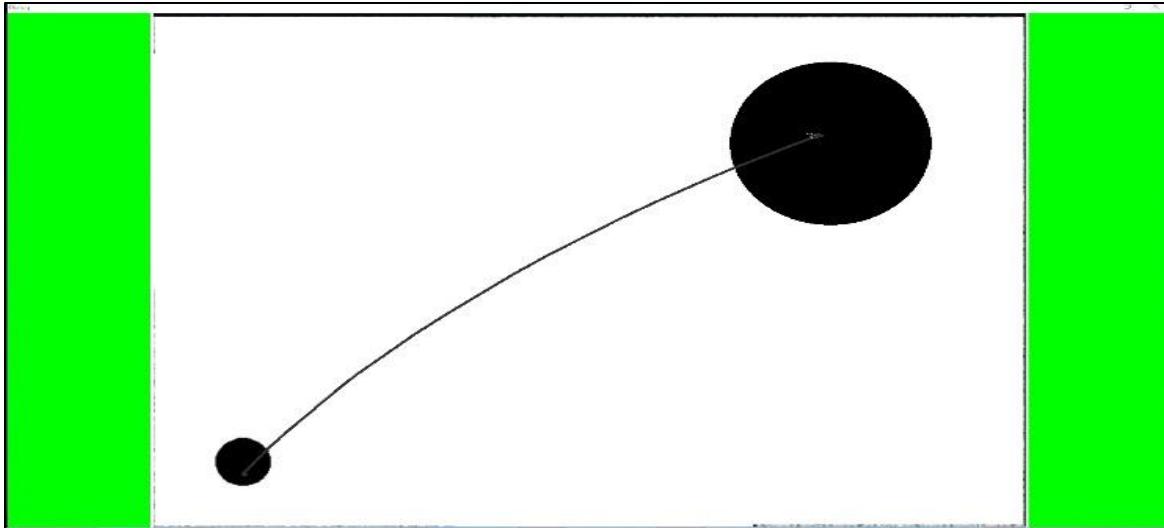


Figure E.4 The interface displayed on the screen of the tablet including the guide sheet on the center with the fast stroke, and the visual cue on each side

stimulus tests on the new system, auditory stimulus test on the new system, visual stimulus test on the old system and auditory stimulus test on the old system. The visual stimulus test was always executed before the audio one. Each day, we changed the order, the first day, the tests were executed on the new system before doing it on the old one. The second day, the tests were executed on the old system before doing it on the new one. The same process was done the other days. A new guide sheet was used to perform those tests in both systems. This new guide sheet has a smaller target zone to restrict the movement of subjects. Figure E. 3. and E.4. respectively depict the user interface and the new guide sheet used to perform the test while the visual cue is displayed on the tablet screen. The user interface allows to select different tests and validate the stroke performed. It also displays several pieces of information, such as the current stroke, the previous stroke, the velocity and pressure profiles.

3. THE DIFFERENT LOGNORMAL MODELS

After recording the data, two different lognormal models were used to extract the parameters from the original data: The Delta-Lognormal and Sigma-Lognormal model. The difference between those two models is that the Delta-Lognormal model will reconstruct the velocity profile using only two lognormals. Instead of that, the Sigma-Lognormal model the model will use as many lognormals as necessary to reconstruct the signal in order to achieve at least a SNR of 25 dB. The parameters extracted from the two models are different. To extract the Delta-Lognormal

parameters, we used the XZERO extractor [22] [110]. This software extract 7 parameters plus the SNR for each stroke. In the case of the Sigma-Lognormal model, the Script Studio software was used to extract the parameters [21] [50] [111] [97] [145]. This software extract 8 parameters linked with this model. There are 5 common parameters:

- t_0 : Time required (in seconds) for the brain to produce a motor command. t_0 should not be confused with the moment a sound cue is emitted ($t = 0$) nor with the response time (RT) measured by moment of movement onset. t_0 is the moment when a population of neurons sends a motor command. In other words, t_0 occurs after the start cue, given that the central nervous system must perceive the sound cue before sending the command, but before RT, given that a delay is necessary for the motor command to be transmitted and for enough muscle cells to be recruited to begin the movement effectively [99]. Longer t_0 is indicative of greater delays before a motor command is emitted and, therefore, a longer time to perceive the stimulus and prepare the command. In the case of the Delta-Lognormal model, there is one t_0 by a stroke, while in the case of Sigma-Lognormal model there is a different t_0 for each lognormal used to reconstruct the stroke.
- SNR: The signal-to-noise ratio between the original speed profile and the reconstructed speed profile, computed in decibels (dB). This is a measure of the quality of the sigma-lognormal reconstruction. Higher SNR indicates better reconstruction.
- $D_{(n)}$: The amplitude of the movement associated with each motor command, in millimeters. Greater D indicates greater amplitude of the movement associated with a command. In the case of the Delta-Lognormal model, there is two D s (D_1 and D_2) per stroke, while in the case of Sigma-Lognormal model there is a different D for each lognormal used to reconstruct the stroke. D_{tot} is the total amplitude of the movement equal to the difference between D_1 and D_2 .
- $\mu_{(n)}$: Logtime delay, the time required to reach the middle of movement distance on a logarithmic scale. It can be associated with the general response speed of the neuromuscular system to a command. In the case of the Delta-Lognormal model, there is two μ_s (μ_1 and μ_2) by a stroke, while in the case of Sigma-Lognormal model there a different μ for each lognormal used to reconstruct the stroke.

- $\sigma_{(n)}$: Logresponse time, the response time on a logarithmic scale of the neuromuscular system's response to a command. Sigma (σ) is used to compute an estimate of movement duration, it also reflects the asymmetry of the lognormal. In the case of the Delta-Lognormal model, there are two σ_s (σ_1 and σ_2) by stroke, while in the case of Sigma-Lognormal model there is a different σ for each lognormal used to reconstruct the stroke.

In the case of the Sigma-Lognormal model, three supplementary parameters are extracted:

nbLog: The number of lognormal functions required to reconstruct the signal. This parameter represents fluidity of movement by the participant. Higher nbLog, indicates less fluid movement.

- θ_s : The starting angular direction of each stroke in radians.
- θ_e : The ending angular direction of each stroke in radians.

In addition, the SNR/nbLog was calculated for the Sigma-Lognormal model:

- SNR/nbLog: the SNR/nbLog is used as a performance criterion. The capacity to reconstruct a movement's speed profile with lognormals can be interpreted as an indicator of the quality of motor control since the lognormal speed profile corresponds to perfect motor control

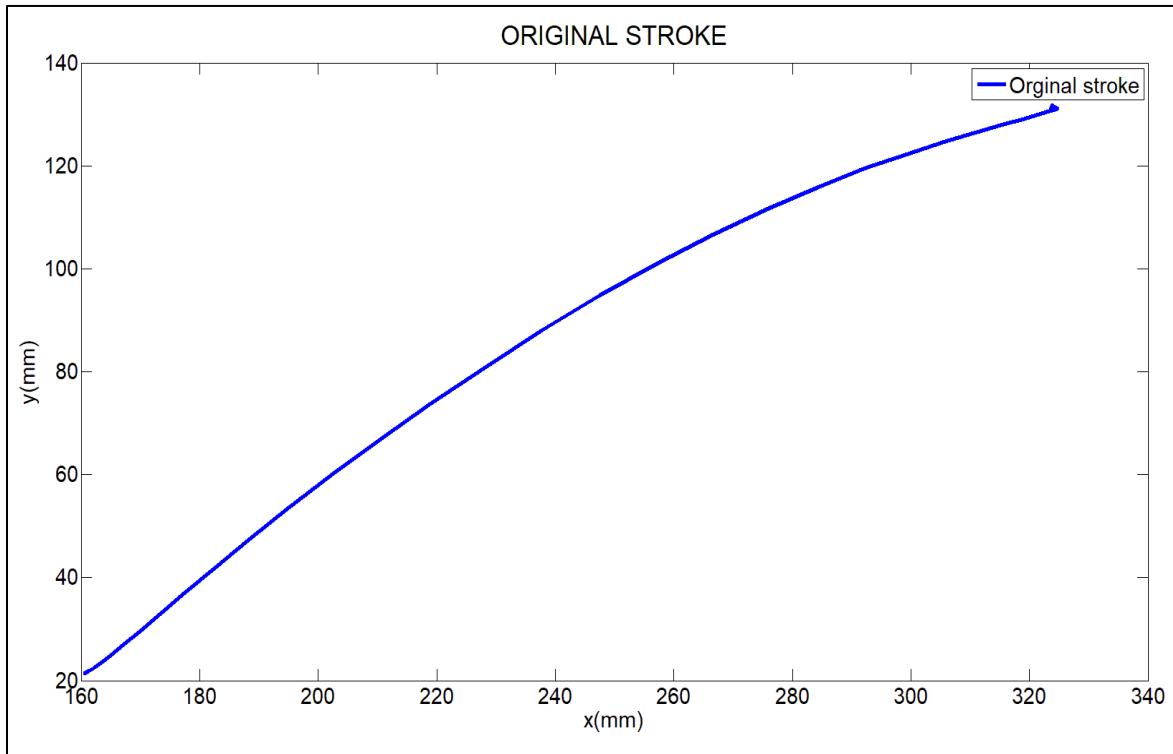


Figure E.5 An original stroke executed by a subject to be reconstructed by the two models

[112]. Higher SNR/nbLog indicates that the movement is closer to ideal lognormal behaviour predicted by the lognormality principle.

The nbLog, SNR and SNR/nbLog describe the global state of the neuromotor system lognormality [12] [109] [14] [99]. The t_0 and D_n are indexes of the internal high-level functioning of the neuromotor system and represent the neuromotor action plan. The μ_n and σ_n parameters are peripheral parameters related to motor program execution. Finally, θ_s and θ_e are intrinsic characteristics of each lognormal and also related to the motor program action plan [14] [113].

The following figures and tables present the results of the reconstruction of a stroke with both Lognormal models, the Delta-Lognormal and Sigma-Lognormal models. Figure E.5. represents the original stroke executed by a subject.

This stroke will be used to extract the parameters. Table E.1 and Table E.2 represent respectively the Delta-Lognormal and Sigma-Lognormal parameters extracted by the software. Figure E.6 and Figure E. 7 represent respectively the reconstruction of the velocity using the Delta-Lognormal and the Sigma-Lognormal model of the stroke shown above.

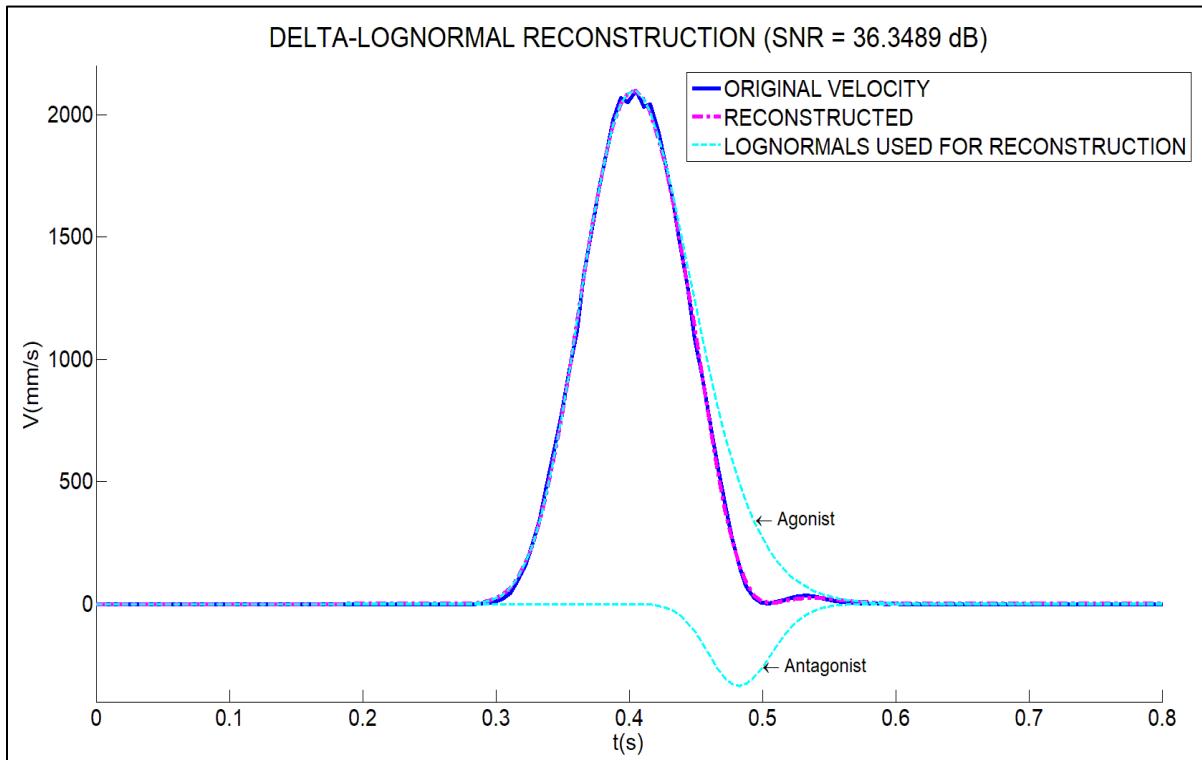


Figure E.6 The Delta-Lognormal reconstruction of the velocity of the stroke shown of Figure E.5

Table E.1 The Delta-Lognormal parameters extracted by the software for the stroke presented on Figure E.5

PARAMETERS							
$t_0(s)$	$D_1(\text{mm})$	μ_1	σ_1	$D_2(\text{mm})$	μ_2	σ_2	SNR
0,082	221,06	-1,119	0,129	19,86	-0,912	0,058	36,349

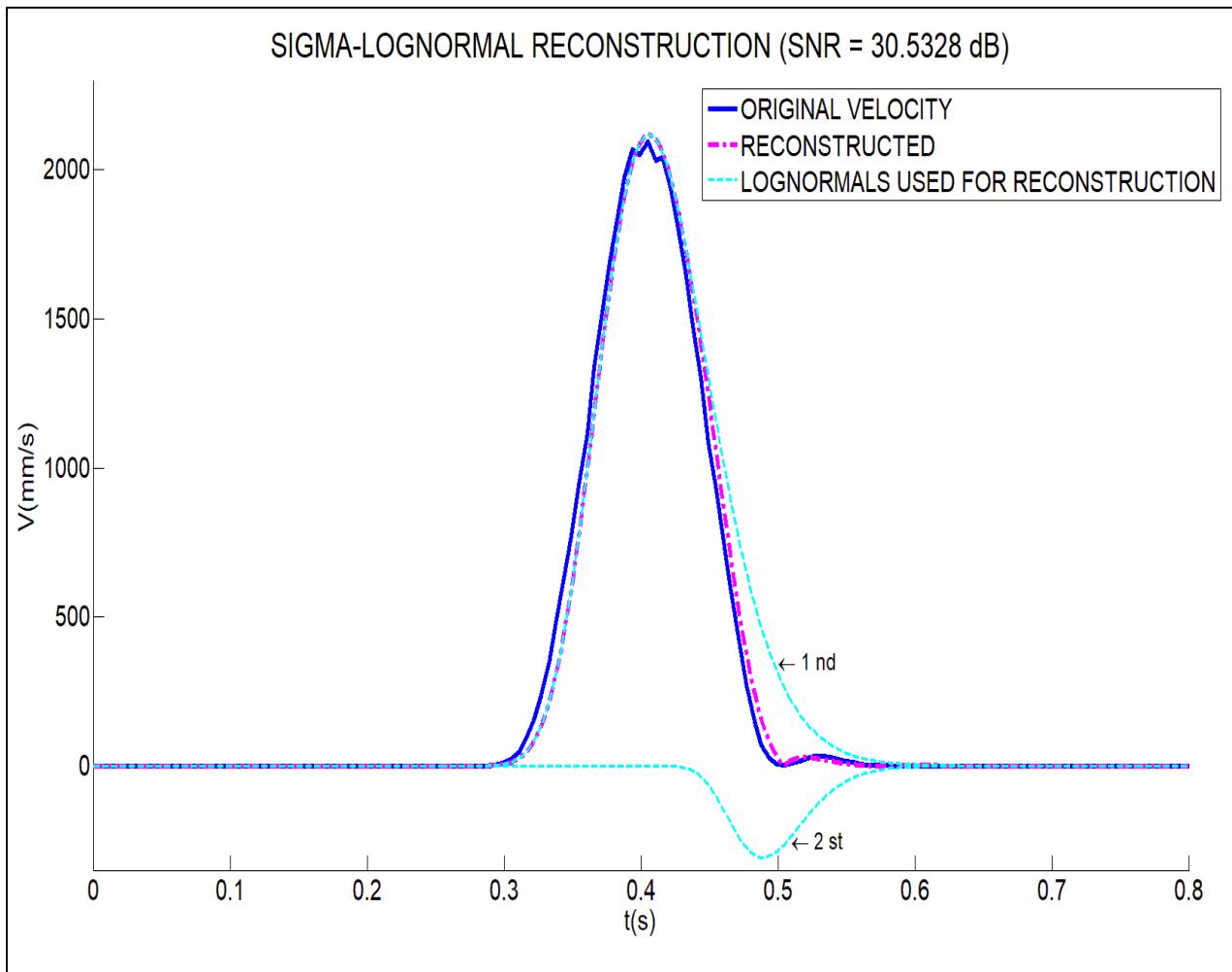


Figure E.7 The Sigma-Lognormal reconstruction of the velocity of the stroke shown of Figure E.5

Table E.2 The Sigma-Lognormal parameters extracted by the software for the stroke presented on Figure E.5

GLOBAL PARAMETERS					
nbLog	2	SNR (dB)	30,532		
LOGNORMAL PARAMETERS					
t_0 (s)	D (mm)	μ	σ	θ_s	θ_e
0,164	218,45	-1,392	0,168	0,844	0,324
0,355	20,89	-1,967	0,198	-2,334	-3,183

4. STATISTICAL ANALYSIS

After extracting the parameters, the data for each day were regrouped by tests. Indeed, all the data recorded with the new system using the visual stimulus were regrouped, and so forth. A t-test for independent measures was conducted on the parameters extracted by Script Studio from all the strokes made. Bonferroni correction for multiple comparisons was used to determine if there is a significant difference. P-value must be less than 0.00625 ($\alpha = 0.05/8 = 0.00625$) to have a significant statistical difference between the two groups.

5. PRELIMINARY RESULTS AND DISCUSSION

The following four tables present the results of the statistical analysis comparing the new system to the Old one. The first two tables are respectively the results of t-test for independent groups between with the visual and sound cue using the Delta-Lognormal model. The last two tables are respectively the results of t-test for independent groups between with the visual and sound cue using the Sigma-Lognormal model. This preliminary test showed that there is a significant difference, with the Bonferroni correction, on the SNR between the two systems in both visual and audio stimuli and for both models. In all cases, the SNR mean is higher for the new system. This could be caused by the higher spatial resolution of the new Wacom Cintiq 13HD as compared to the Wacom Intuos2. The Dtot showed a significant difference between the two systems in both visual and audio stimuli and for both models. In both cases, the Dtot mean is higher for the new system. The subject stipulates that the Wacom Cintiq 13HD has a surface with less friction than the surface of the Wacom Intuos2. Drawing on the new tablet is smoother than the old one, so the subject has more difficulties to stop their movements, which leads to higher Dtot. In the case of the Delta-Lognormal model, there is a significant difference between the two systems for D1, it is

caused by the fact that there is no friction when executing the movement and the person can reach its real velocity.

Table E.3 Result of t-test for independent groups between the new and the old system with the visual cue using the Delta-Lognormal model

Parameters	New system M(SD)	Old system M(SD)	t-value	df	p-value
t_0	0,144 (0,068)	0,15 (0,066)	-0,740	302	0,460
D_1	234,45 (30,52)	218,62 (31,64)	4,44	302	0,000 ^a
μ_1	-1,387 (0,212)	-1,447 (0,234)	2,348	302	0,020 ^b
σ_1	0,2 (0,053)	0,218 (0,06)	-2,886	302	0,004 ^a
D_2	39,87 (28,84)	28,55 (30,88)	3,30	302	0,000 ^a
μ_2	-1,114 (0,155)	-1,116 (0,184)	0,134	302	0,893
σ_2	0,101 (0,033)	0,095 (0,049)	1,228	302	0,221
SNR	32,088 (4,185)	30,272 (5,308)	3,318	302	0,001 ^a
D_{tot}	194,58 (7,33)	190,06 (7,77)	5,21	302	0,000 ^a

Table E.4 Result of t-test for independent groups between the new and the old system with the sound cue using the Delta-Lognormal model

Parameters	New system M(SD)	Old system M(SD)	t-value	df	p-value
t_0	0,1 (0,051)	0,091 (0,053)	1,496	290	0,136
D_1	228,16 (26,41)	219,47 (26,4)	2,811	290	0,005 ^a
μ_1	-1,415 (0,201)	-1,457 (0,211)	1,754	290	0,080
σ_1	0,196 (0,043)	0,215 (0,054)	-3,355	290	0,001 ^a
D_2	32,76 (25,19)	29 (26,29)	1,250	290	0,212
μ_2	-1,142 (0,165)	-1,149 (0,187)	0,375	290	0,708
σ_2	0,097 (0,035)	0,098 (0,047)	-0,288	290	0,773
SNR	32,259 (4,311)	30,23 (5,055)	3,696	290	0,000 ^a
D_{tot}	195,4 (8,57)	190,47 (9,53)	4,648	290	0,000 ^a

For the visual stimulus, there is a significant difference, with the Bonferroni correction, in the t_0 parameters for the Sigma-Lognormal model. The t_0 mean is lower for the new system compared to the old one. On the other hand, there is no significant difference for the t_0 in the Delta-Lognormal model. A difference exists between the two models for t_0 , because the Delta-Lognormal model stipulates that the lognormals composing the movement have opposed direction and start at the

same time. In contrast, the Sigma-Lognormal model will reconstruct the velocity using a vector addition of the different lognormals which their own t_0 .

Table E.5 Result of t-test for independent groups between the new and the old system with the visual cue using the Sigma-Lognormal model

Parameters	New system M(SD)	Old system M(SD)	t-value	df	p-value
SNR	30,47 (2,86)	28,43 (3,36)	5,84	320	0,0000 ^a
nbLog	1,98 (0,37)	2,03 (0,64)	-0,95	320	0,3419
SNR/nbLog	16,01 (3,63)	15,49 (5,43)	1,00	320	0,3137
t_0	0,169 (0,17)	0,211 (0,18)	-2,94	643	0,0036 ^a
t_{0i}	0,124 (0,139)	0,152 (0,106)	-1,97	320	0,0490 ^b
D	120,64(95,89)	112,03(94,35)	1,16	643	0,2452
D_{tot}	197,41 (19,0)	191,07 (15,5)	3,28	320	0,0011 ^a
μ	-1,52 (0,65)	-1,59 (0,58)	1,62	643	0,1045
σ	0,21 (0,14)	0,24 (0,15)	-2,47	643	0,0136 ^b
θ_s	0,26 (4,58)	-0,89 (4,64)	2,47	643	0,0136 ^b
θ_e	-0,08 (4,58)	-1,15 (4,64)	2,23	643	0,0256 ^b

Table E.6 Result of t-test for independent groups between the new and the old system with the sound cue using the Sigma-Lognormal model

Parameters	New system M(SD)	Old system M(SD)	t-value	df	p-value
SNR	30,19 (2,60)	28,59 (3,23)	4,92	322	0,0000 ^a
nbLog	1,88 (0,36)	2,17 (0,77)	-4,40	322	0,0000 ^a
SNR /nbLog	16,70 (3,73)	14,77 (5,57)	3,66	322	0,0003 ^a
t_0	0,16 (0,14)	0,18 (0,18)	-1,56	655	0,1186
t_{0i}	0,09 (0,08)	0,09 (0,08)	0,45	322	0,6519
D	123,63(95,33)	106,67(93,02)	2,31	655	0,0214 ^b
D_{tot}	197,49 (15,4)	188,52 (12,8)	5,69	322	0,0000 ^a
μ	-1,66 (0,6)	-1,67 (0,57)	0,25	655	0,7966
σ	0,23 (0,13)	0,26 (0,15)	-2,18	655	0,0294 ^b
θ_s	0,20 (4,57)	-0,29 (4,62)	1,38	655	0,1667
θ_e	-0,17(4,58)	-0,59 (5,02)	1,13	655	0,2572

^aSignificant with Bonferroni corrections for multiple comparisons ($p < 0.00625$)

^bSignificant at $p < 0.05$

C. THE BATTERY OF TESTS INTEGRATED ON THE NEW SYSTEM

Multiple tasks were integrated on the new system to investigate the neuromuscular system [66]. Two different stimuli are available: a sound cues and green light. Each row of Figure E.8 represents a different task. In all tasks, the person should execute a fast pen stroke from the black dot (the starting point) to the grey area (the arrival zone). For each task, the red blinking light indicates that the system is ready to use as it is shown on Figure E. 8. a. If the person starts its movement before the cue, an orange cross will be displayed as shown on Figure E.8. c. The first task is available with a visual and sound cue (Figure E.8. a). This task evaluates the ability of the subjects to react as quickly as possible to these stimuli. The second task, Figure E.8.d-e., examines the capacity of the person to react quickly and to make a good choice about the direction he draws the stroke. The second row of Figure E.8. shows the visual stimulus showed on each side of the guide sheet indicating the direction to draw the stroke. The task shown on the third row, Figure E.8. f-g-h-i., is related to the speed-accuracy trade off. [29]. It's used to assess the ability of the participants to coordinate a movement under the speed and accuracy requirement. In this case, each figure represents the different size of the target area, and for each target zone, there is four possible starting point. This task is executed with a sound cue. The triangle task, Figure E.8. j-k-l., examines the capacity of a person to produce a sequence of fast pen strokes with a combination of speed and accuracy. There are three different sizes of triangle to evaluate the capacity of drawing sequence of a stroke at different sizes. This task is executed with a sound cue. The last task is an oscillation at maximal speed. The oscillation task assesses the rhythmical properties of the subject. All those tests are included in the new system and will be used to investigate the different aspect of the human neuromuscular movement.

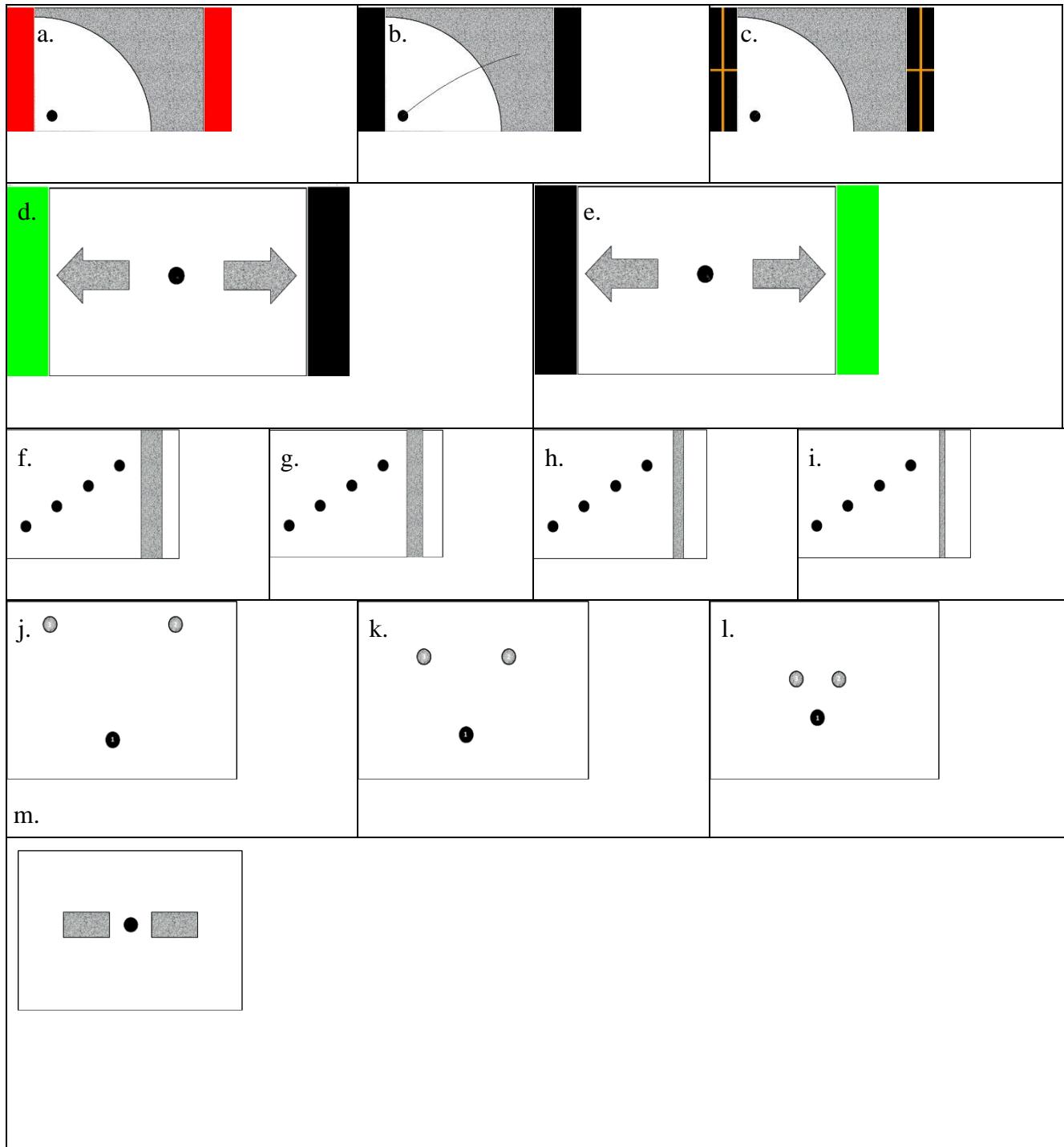


Figure E.8 Different guiding sheet and stimuli integrated in the new system. a. guiding sheet for the simple stroke test with the red light, this guide sheet is used with a visual and sound cue. b. A simple fast stroke displayed on the guide sheet. c. The orange cross displayed is not valid. Guiding sheets for the fast pen stroke on a choice reaction time protocol (d) , (e). f,g,h,i are guiding sheets for the speed/accuracy trade-off task. j,k,l are guiding sheets for the fast pen stroke sequences

D. CONCLUSION

This project consisted of transferring the old Sign@medic system based on a Wacom intuos2 tablet under Windows XP operating system to the new Wacom Cintiq 13HD to operate under windows 10. The new system has the advantage of including the external stimulus generator device and the guide sheets in the software. This makes the system more compact, ergonomics and user-friendly. Eventually for more concrete result, the two systems will be tested by a drawing robot to remove the human variability impact. Finally, the proposed system can be used in different research projects dealing with the human neuromotor system. The system is also available for non-commercial use after signing nondisclosure agreements.

E. ACKNOWLEDGEMENTS

This project was supported by NSERC grant RGPIN-2015-06409 to Réjean Plamondon and MESI and Univvalor support to Scribens laboratory.