



HAL
open science

Amélioration d'ergonomie du télégeste, sur réseau à retard par utilisation de prédicteurs statistiquement pertinents

Ismael Belghit

► **To cite this version:**

Ismael Belghit. Amélioration d'ergonomie du télégeste, sur réseau à retard par utilisation de prédicteurs statistiquement pertinents. Interface homme-machine [cs.HC]. Université Joseph-Fourier - Grenoble I, 2003. Français. NNT: . tel-00006165

HAL Id: tel-00006165

<https://theses.hal.science/tel-00006165>

Submitted on 28 May 2004

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



UFR PHYSIQUE ET SCIENCES POUR L'INGENIEUR



N° attribué par la bibliothèque

THESE

présentée pour l'obtention du

Doctorat de l'Université Joseph Fourier de Grenoble

spécialité : Signal, image, parole et télécommunication

(arrêté du 25 avril 2002)

par

Ismaël BELGHIT

**Amélioration d'ergonomie du télégeste, sur réseau à retard, par
utilisation de prédicteurs statistiquement pertinents.**

soutenue le 14 janvier 2003

devant le jury composé de :

Monsieur	Bernard Hennion	<i>co-directeur</i>
Monsieur	Jacques Lemordant	<i>directeur</i>
Monsieur	Philippe Coiffet	<i>rapporteur</i>
Monsieur	Jean-Pierre Jessel	<i>rapporteur</i>
Madame	Sabine Coquillart	<i>examinatrice</i>
Monsieur	Augustin Lux	<i>examineur</i>

à mes parents, frère et soeurs,

à Anne



Résumé

Mots-clés : système à retour d'effort (SRE), toucher distant, boucle haptique, prédicteurs, réseau à retard, télégeste, pertinence statistique, latence incompressible.

Après la transmission de l'image et du son à travers les réseaux télécoms, il était logique de s'intéresser à la transmission du geste. Le " toucher 3D " est en train de révolutionner la façon de travailler des utilisateurs de la 3D sur ordinateur. Il permet aux gens de manipuler directement des objets et des données numériques de la même façon que dans la réalité en utilisant leur sens du toucher. Il existe des systèmes à retour d'effort, représentés par des manches munis de moteurs pilotés par des logiciels. Ils renvoient dans la main de l'utilisateur des sensations de forces, de vibrations, ils sont programmables en fonction d'une certaine tâche à effectuer dans le monde virtuel.

Lors d'un toucher distant, d'un télégeste, nous sommes confrontés au problème de l'instabilité de la boucle haptique (homme-système à retour d'effort-réseau-retour de la force dans la main du manipulateur). Cette instabilité est provoquée par le retard introduit par la latence incompressible du réseau de télécommunication. Idéalement on désirerait obtenir une réponse instantanée (inférieure à la milliseconde), or l'élaboration d'une telle réponse est irréalisable car elle est non causale. Afin de contrer le retard introduit par le réseau nous avons utilisé des prédicteurs dans le but d'obtenir une réponse simultanée à la sollicitation du manipulateur du système à retour d'effort. Cette amélioration d'ergonomie du télégeste est réalisée grâce à la pertinence statistique du modèle de l'algorithme de prédiction.



Remerciements

Je remercie en premier lieu Bernard Hennion, qui m'a permis de réaliser cette thèse au sein de France Télécom Recherche et Développement sur un sujet passionnant et enthousiasmant. Je remercie également Jacques Lemordant pour ses conseils avisés sur la présentation et le contenu de cette thèse.

Je remercie Agnès Guerraz et Alexandre Vienne pour leur collaboration aux articles qui ont été publiés, ainsi que pour l'aide et les conseils qu'ils m'ont promulgués dans les problèmes de programmation.

Je remercie aussi Bruno Dillenseger, pour son aide et sa patience dans ma formation à FrameMaker, le logiciel de traitement de texte qui a permis de réaliser ce document.

Je remercie Daniel Baudois, professeur à l'INPG, pour sa disponibilité et son aide dans le cadre du filtrage de Kalman.

Je remercie Christian Bayle pour ses dépannages "vite fait, sur le pouce", pour sa patience et sa disponibilité à résoudre des problèmes de soft mais aussi de hard.

Je remercie Dominique Bouchot pour les discussions pleines d'intérêts sur un sujet diabolique.

Table des matières

Chapitre 1	
Introduction	19
1.1 Problématique	19
1.1.1 Définition du toucher local - - - - -	19
1.1.2 Définition du toucher distant- - - - -	20
1.1.3 Position du problème- - - - -	21
1.2 Présentation du document	23
Chapitre 2	
L'haptique : l'interaction avec des objets virtuels	25
2.1 La proprioception	26
2.1.1 La force et le mouvement sont les plus importants signaux haptiques -	26
2.1.2 Capteur-effecteur : une Co-Dépendance - - - - -	26
2.1.3 Les mondes virtuels - - - - -	27
2.2 Le Système à Retour d'Effort (SRE)	27
2.2.1 La technologie du toucher - - - - -	27
2.2.2 L'interface haptique Phantom: un appareil pour sonder les objets virtuels - - - - -	27
2.2.3 Trois critères importants pour obtenir une interface efficace- - - - -	28
2.2.4 Les observations sur la perception.- - - - -	29
2.2.5 Explorer les nouveaux mondes de l'interaction haptique - - - - -	29
2.3 Les diverses applications	30
2.3.1 L'entraînement visant à opérer des patients virtuels - - - - -	30
2.3.2 Le télégeste- - - - -	33
La télé-échographie	33
L'escrime virtuelle	34

2.4 Wave Transform : une autre approche	35
2.5 Conclusion	36
Chapitre 3	
Le modèle d'algorithme de prédiction	37
3.1 Les politiques de prédiction	38
3.1.1 L'algorithme LPC - - - - -	38
3.1.2 Le filtre de Kalman - - - - -	38
3.1.3 L'algorithme personnalisé - - - - -	39
3.2 Présentation des différents prédicteurs	40
3.2.1 Le prédicteur d'ordre zéro - - - - -	40
3.2.2 Le prédicteur d'ordre 1 - - - - -	41
3.2.3 Le prédicteur d'ordre 2 - - - - -	45
3.2.4 Le prédicteur d'ordre 3 - - - - -	48
3.3 Conclusion	50
Chapitre 4	
Filtrage du bruit de quantification	51
4.1 Le bruit de quantification	52
4.1.1 Evaluation du spectre du bruit - - - - -	52
L'estimateur spectral simple	52
L'estimateur spectral moyenné	53
Présentation des différentes fonctions fenêtre	55
4.1.2 Les résultats de l'estimation spectrale - - - - -	59
Spectre de la dérivée première du signal (vitesse)	59
Spectre de la dérivée seconde du signal (accélération)	59
Spectre de la dérivée tierce du signal (jitter)	60
4.2 Le filtrage du bruit	61
4.2.1 Généralités - - - - -	61
Le système linéaire numérique	61
Le filtre passe bas	62
4.2.2 Le choix du type de filtre - - - - -	63
Le filtre passe bas à réponse impulsionnelle finie (RIF)	63
Le filtre passe bas à réponse impulsionnelle infinie (RII)	64
Le filtre passe bas (RII) elliptique	64
4.2.3 Le filtrage du bruit contenu dans la vitesse - - - - -	65
4.2.4 Le filtrage du bruit contenu dans l'accélération - - - - -	66
4.2.5 Le filtrage du bruit contenu dans le jitter - - - - -	68
4.3 Conclusion	69
Chapitre 5	
La prédiction : un problème de filtrage	71
5.1 Définition des limites du cadre de l'étude	72
5.2 Le prédicteur d'ordre 0	72

5.3 Le prédicteur d'ordre 1	73
5.3.1 Calcul du module et de la phase du filtre FP1	74
5.3.2 Représentation graphique du module et de la phase du filtre FP1	75
Caractéristiques de FP1 pour un délai de 10 ms	75
Caractéristiques de FP1 pour un délai de 30 ms	76
Caractéristiques de FP1 pour un délai de 100 ms	77
5.4 Le prédicteur d'ordre 2	77
5.4.1 Calcul du module et de la phase du filtre FP2	78
5.4.2 Représentation graphique du module et de la phase du filtre FP2	79
Caractéristiques de FP2 pour un délai de 10 ms	79
Caractéristiques de FP2 pour un délai de 30 ms	80
Caractéristiques de FP2 pour un délai de 100 ms	80
5.5 Le prédicteur d'ordre 3	81
5.5.1 Calcul du module et de la phase du filtre FP3	81
5.5.2 Représentation graphique du module et de la phase du filtre FP3	83
Caractéristiques de FP3 pour un délai de 10 ms	83
Caractéristiques de FP3 pour un délai de 30 ms	84
Caractéristiques de FP3 pour un délai de 100 ms	85
5.6 Les prédicteurs d'ordre plus élevé	85
5.7 Récapitulatif des résultats	87
5.7.1 Comparaison des performances des trois filtres pour un délai de 10 ms	87
Pour un geste quelconque	87
Pour un geste d'écriture	88
5.7.2 Comparaison des performances des trois filtres pour un délai de 30 ms	89
Pour un geste quelconque	89
Pour un geste d'écriture	90
5.7.3 Comparaison des performances des trois filtres pour un délai de 100 ms	91
Pour un geste quelconque	91
Pour un geste d'écriture	92
5.8 Conclusion	92
Chapitre 6	
Les supports techniques du télégeste	95
6.1 Le modèle OSI	96
6.2 Présentation du RNIS et de l'IP	98
6.2.1 Le RNIS	98
6.2.2 Le TCP/IP	100
Le protocole UDP	101
Le protocole TCP	101
6.3 Exemple : la télé-échographie	103
6.3.1 L'architecture réseau	103
6.3.2 Evaluation du flux haptique	103
6.4 Problèmes rencontrés dans la transmission du geste	104
6.5 Conclusion	105

Chapitre 7

Validation théorique du modèle	107
7.1 L'outil de comparaison	108
7.2 Evaluation des performances du prédicteur d'ordre 1	108
7.2.1 Le signal sinusoïdal de fréquence 3 Hz - - - - -	108
7.2.2 Le signal sinusoïdal de fréquence 8 Hz - - - - -	109
7.3 Evaluation des performances du prédicteur d'ordre 2	111
7.3.1 Le signal sinusoïdal de fréquence 3 Hz - - - - -	111
7.3.2 Le signal sinusoïdal de fréquence 8 Hz - - - - -	112
7.4 Evaluation des performances du prédicteur d'ordre 3	113
7.4.1 Le signal sinusoïdal de fréquence 3 Hz - - - - -	113
7.4.2 Le signal sinusoïdal de fréquence 8 Hz - - - - -	114
7.5 Conclusion	115

Chapitre 8

Evaluation pratique des modèles proposés	117
8.1 Architecture du codeur prédicteur d'ordre 1	118
8.2 Architecture du codeur prédicteur d'ordre 2	119
8.3 Architecture du codeur prédicteur d'ordre 3	120
8.4 Discussion sur le choix des architectures	121
8.4.1 L'architecture du codeur prédicteur d'ordre 1 (CP1) - - - - -	121
8.4.2 L'architecture du codeur prédicteur d'ordre 2 (CP2) - - - - -	124
8.4.3 L'architecture du codeur prédicteur d'ordre 3 (CP3) - - - - -	126
8.5 Résultats expérimentaux sur un signal gestuel	128
8.5.1 Résultats du codeur prédicteur d'ordre 1 - - - - -	128
Erreur quadratique moyenne en fonction du délai d.	128
8.5.2 Résultats du codeur prédicteur d'ordre 2 - - - - -	129
Erreur quadratique moyenne en fonction du délai d	129
8.5.3 Résultats du codeur prédicteur d'ordre 3 - - - - -	129
Erreur quadratique moyenne en fonction du délai d	129
8.5.4 Résultats récapitulatifs- - - - -	130
Comparaison pour un délai de 15 ms	130
Comparaison pour un délai de 40 ms	131
Comparaison pour un délai de 70 ms	132
8.5.5 Tableau récapitulatif : - - - - -	134
8.6 Conclusion	134

Chapitre 9

Application: le circuit automobile	135
9.1 Description de l'application proposée	136
9.1.1 Description graphique - - - - -	136
9.1.2 Description haptique - - - - -	137
9.1.3 Description conceptuelle - - - - -	138

9.2 Mesure quantitative : distance position locale-position prédite	140
9.2.1 Distance position locale-position prédite, pour un délai de 10 ms - - -	140
9.2.2 Distance position locale-position prédite, pour un délai de 30 ms - - -	141
9.2.3 Distance position locale-position prédite, pour un délai de 50 ms - - -	142
9.3 L'apport des prédicteurs en terme de confort dans un télégeste	143
9.4 Conclusion	145

Chapitre 10

Conclusion.....	147
10.1 Bilan	147
10.2 Perspectives	148
Références bibliographiques	153

Annexe A

La Transformée en Z	159
----------------------------------	------------

Annexe B

Quelques procédures	163
B.1 La dérivation par différence finies	163
B.2 Le filtre RIF 10	164
B.3 Le filtre elliptique d'ordre 5	165

Annexe C

La prédiction de Smith	169
-------------------------------------	------------

Table des figures

Introduction

figure 1.1 : synopsis de la boucle haptique pour un toucher local, elle doit s'effectuer en lms.	20
figure 1.2 : synopsis de la boucle haptique pour un toucher distant, elle doit s'effectuer en lms.	21
figure 1.3 : Existence d'une distance entre la position de la main du manipulateur dans la scène distante et la position de la main dans la scène locale	22

L'haptique : l'interaction avec des objets virtuels

figure 2.1 : Le système à retour d'effort Phantom de chez Sensable.	28
figure 2.2 : Modèle virtuel 3D d'un oeil humain, incisions faites au bistouri.	30
figure 2.3 : Modèle virtuel 3D d'un coeur humain.	31
figure 2.4 : Station de commande haptique pour la chirurgie dentaire.	32
figure 2.5 : Données géophysiques 3D.	33
figure 2.6 : Station de commande haptique pour la télé-échographie.	34
figure 2.7 : Interface du jeu d'escrime	35

Le modèle d'algorithme de prédiction

figure 3.1 : le prédicteur d'ordre 0.	40
figure 3.2 : erreur de prédiction.	41
figure 3.3 : explication graphique de la prédiction d'ordre 1	42
figure 3.4 : circuit permettant de calculer la vitesse	43
figure 3.5 : gain en fréquence du filtre permettant de calculer la vitesse.	44
figure 3.6 : vitesse de la composante x du signal original.	44
figure 3.7 : circuit permettant de calculer l'accélération	46
figure 3.8 : gain en fréquence du filtre permettant de calculer l'accélération.	47
figure 3.9 : accélération de la composante x du signal original.	47
figure 3.10 : circuit permettant de calculer le jitter.	48
figure 3.11 : gain en fréquence du filtre permettant de calculer le jitter.	49
figure 3.12 : jitter de la composante x du signal original	50

Filtrage du bruit de quantification

figure 4.1 : enveloppe et spectre de la fenêtre Rectangulaire.	55
figure 4.2 : enveloppe et spectre de la fenêtre de Bartlett.	56

figure 4.3 : enveloppe et spectre de la fenêtre de Hanning. 57

figure 4.4 : enveloppe et spectre de la fenêtre de Hamming. 58

figure 4.5 : enveloppe et spectre de la fenêtre de Blackman. 58

figure 4.6 : densité spectrale de puissance de la vitesse. 59

figure 4.7 : densité spectrale de puissance de l'accélération. 60

figure 4.8 : densité spectrale de puissance du jitter. 61

figure 4.9 : synopsis d'un système linéaire. 62

figure 4.10 : Gabarit d'un filtre passe bas. 63

figure 4.11 : filtrage elliptique du spectre de la vitesse. 66

figure 4.12 : filtrage elliptique du spectre de l'accélération. 67

figure 4.13 : filtrage elliptique du spectre du jitter. 69

La prédiction : un problème de filtrage

figure 5.1 : synopsis du filtrage. 72

figure 5.2 : circuit permettant de réaliser la prédiction d'ordre 1. . 73

figure 5.3 : gain et phase du filtre FP1 pour un d de 10 ms. 75

figure 5.4 : gain et phase du filtre FP1 pour un d de 30 ms. 76

figure 5.5 : gain et phase du filtre FP1 pour un d de 100 ms. 77

figure 5.6 : circuit permettant de réaliser la prédiction d'ordre 2 . 77

figure 5.7 : gain et phase du filtre FP2 pour un d de 10 ms. 79

figure 5.8 : gain et phase du filtre FP2 pour un d de 30 ms. 80

figure 5.9 : gain et phase du filtre FP2 pour un d de 100 ms. 80

figure 5.10 : circuit permettant de réaliser la prédiction d'ordre 3. 81

figure 5.11 : gain et phase du filtre FP3 pour un d de 10 ms 83

figure 5.12 : gain et phase du filtre FP3 pour un d de 30 ms. 84

figure 5.13 : gain et phase du filtre FP3 pour un d de 100 ms. 85

figure 5.14 : en haut, la filtrée de la dérivée tierce d'un geste d'écriture. (comp y) en bas, la filtrée de la dérivée quatrième de ce même geste. 86

figure 5.15 : gain et phase des filtres FP1, FP2, FP3 avec un délai de 10 millisecondes pour un geste quelconque. 87

figure 5.16 : gain et phase des filtres FP1, FP2, FP3 avec un délai de 10 millisecondes pour un geste d'écriture. 88

figure 5.17 : gain et phase des filtres FP1, FP2, FP3 avec un délai de 30 millisecondes pour un geste quelconque. 89

figure 5.18 : gain et phase des filtres FP1, FP2, FP3 avec un délai de 30 millisecondes pour un geste d'écriture. 90

figure 5.19 : gain et phase des filtres FP1, FP2, FP3 avec un délai de 100 millisecondes pour un geste quelconque. 91

figure 5.20 : gain et phase des filtres FP1, FP2, FP3 avec un délai de 100 millisecondes pour un geste d'écriture. 92

Les supports techniques du télégeste

figure 6.1 : Modèle OSI - Norme ISO 7498 (Recommandation X200 du CCITT).	96
figure 6.2 : Architecture réseau pour le poste de télé-échographie (poste serveur car le flux d'images est sortant).	103

Validation théorique du modèle

figure 7.1 : signal initial (sinus de fréquence 3 Hz) et ses prédictions par le prédicteur FP1 pour des délais de 10, 30 et 80 ms.	109
figure 7.2 : signal initial (sinus de fréquence 8 Hz) et ses prédictions par le prédicteur FP1 pour des délais de 10, 30 et 80 ms.	110
figure 7.3 : signal initial (sinus de fréquence 3 Hz) et ses prédictions par le prédicteur FP2 pour des délais de 10, 30 et 80 ms.	111
figure 7.4 : sinus initial (sinus de fréquence 8 Hz) et ses prédictions par le prédicteur FP2 pour des délais de 10, 30 et 80 ms.	112
figure 7.5 : signal initial (sinus de fréquence 3Hz) et ses prédictions par le prédicteur FP3 pour des délais de 10, 30 et 80 ms.	114
figure 7.6 : signal initial (sinus de fréquence 8Hz) et ses prédictions par le prédicteur FP3 pour des délais de 10, 30 et 80 ms.	115

Evaluation pratique des modèles proposés

figure 8.1 : Erreur de prédiction en fonction de l'ordre du filtre elliptique utilisé pour filtrer le bruit présent dans la dérivée première du signal original, pour un délai de 70 ms.	123
figure 8.2 : Signaux prédits par le codeur prédicteur d'ordre 1 dont l'ordre du filtre elliptique varie par pas de 1, n=3, 4, 5, 6, 7, 8.	124
figure 8.3 : Erreur de prédiction en fonction de l'ordre du filtre elliptique utilisé pour filtrer le bruit présent dans la dérivée seconde du signal original, pour un délai de 70 ms.	125
figure 8.4 : Signaux prédits par le codeur prédicteur d'ordre 2 dont l'ordre du filtre elliptique varie par pas de 1, n=3, 4, 5, 6.	125
figure 8.5 : Erreur de prédiction en fonction de l'ordre du filtre elliptique utilisé pour filtrer le bruit présent dans la dérivée tierce du signal original, pour un délai de 70 ms.	126
figure 8.6 : Signaux prédits par le codeur prédicteur d'ordre 3 dont l'ordre du filtre elliptique varie par pas de 1, n=2, 3, 4, 5.	127
figure 8.7 : en haut le mot "désir" numérisé à l'aide d'une application haptique, en dessous la composante y de ce mot.	128
figure 8.8 : Erreur moyenne de prédiction (en mm) effectuée par le codeur prédicteur d'ordre 1, en fonction du délai réseau.	128
figure 8.9 : Erreur moyenne de prédiction (en mm) effectuée par le codeur prédicteur d'ordre 2, en fonction du délai réseau.	129
figure 8.10 : Erreur moyenne de prédiction (en mm) effectuée par le codeur prédicteur d'ordre 3, en fonction du délai réseau.	129
figure 8.11 : Erreur quadratique moyenne en fonction du type de codeur prédicteur utilisé , CP1, CP2, CP3 pour un délai réseau de 15 ms.	130

figure 8.12 : Signaux prédits par les différents codeurs prédicteurs CP1, CP2, CP3 pour un délai réseau de 15 ms. 130

figure 8.13 : Erreur quadratique moyenne en fonction du type de codeur prédicteur utilisé , CP1, CP2, CP3 pour un délai réseau de 40 ms. . 131

figure 8.14 : Signaux prédits par les différents codeurs prédicteurs CP1, CP2, CP3 pour un délai réseau de 40 ms. 132

figure 8.15 : Erreur quadratique moyenne en fonction du type de codeur prédicteur utilisé , CP1, CP2, CP3 pour un délai réseau de 70 ms. . 132

figure 8.16 : Signaux prédits par les différents codeurs prédicteurs CP1, CP2, CP3 pour un délai réseau de 70 ms. 133

Application: le circuit automobile

figure 9.1 : Le circuit automobile, la boule verte correspond à la position locale de la main du manipulateur, la boule rouge correspond à la position prédite distante de la main dans la scène distante. 136

figure 9.2 : synopsis de la boucle haptique pour l'application : "le circuit automobile" 138

figure 9.3 : Le menu déroulant permettant de régler l'ordre de prédiction. 139

figure 9.4 : Le menu déroulant permettant de régler le délai réseau. 139

figure 9.5 : distance entre la position locale de la main de l'utilisateur et sa position distante prédite, avec un délai réseau de 10 ms, cette distance étant évaluée pour les prédicteurs d'ordre 0, 1, 2 et 3. . 140

figure 9.6 : distance entre la position locale de la main de l'utilisateur et sa position distante prédite, avec un délai réseau de 30 ms, cette distance étant évaluée pour les prédicteurs d'ordre 0, 1, 2 et 3. . 141

figure 9.7 : distance entre la position locale de la main de l'utilisateur et sa position distante prédite, avec un délai réseau de 50 ms, cette distance étant évaluée pour les prédicteurs d'ordre 0, 1, 2 et 3. . 142

Chapitre 1

Introduction

1.1 Problématique

1.1.1 Définition du toucher local

A l'aide d'un système à retour d'effort (SRE), on peut toucher des objets virtuels locaux ou distants. L'implémentation d'une interface graphique et haptique permet de calculer la position de la main du manipulateur dans un espace de travail qui possède une représentation virtuelle dans l'ordinateur. Ce calcul de position se fait par l'intermédiaire de capteurs de position intégrés dans le système à retour d'effort. Le programme informatique associé au SRE permet alors de calculer une force que le SRE renvoie dans la main de l'utilisateur (Figure 1.1, page 20).

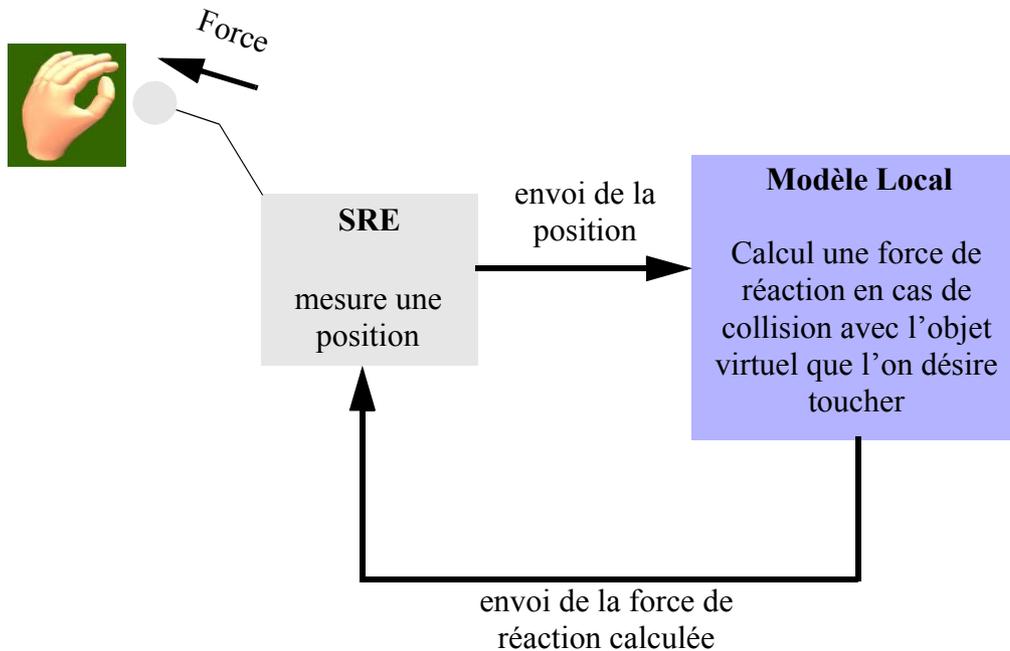


figure 1.1 : synopsis de la boucle haptique pour un toucher local, elle doit s'effectuer en 1ms.

La boucle "haptique" est constituée par la capture de la position de la main du manipulateur du SRE et le renvoi de la force calculée dans le cas d'une collision avec l'objet virtuel que l'on désire toucher. Afin d'obtenir une sensation de toucher acceptable et réaliste pour l'être humain, la fréquence de la boucle haptique a été fixée à 1 kilohertz. En dessous de cette valeur, on ressent une vibration, lors du toucher, similaire à l'effet d'une brosse à dents électrique.

1.1.2 Définition du toucher distant

Pour toucher un objet virtuel distant, il est nécessaire de dupliquer l'environnement de travail. Ainsi on crée une réplique locale de l'objet virtuel distant que l'on désire toucher. Le modèle local estampille les données de position qu'il va transmettre au modèle distant. L'ordinateur distant sur lequel se trouve l'objet virtuel que l'on désire toucher et l'ordinateur local qui est relié au système à retour d'effort (SRE) sont au préalable synchronisés, ce qui permet de mesurer finement le délai de transmission. On transmet, via le réseau, au modèle distant la position de la main de l'utilisateur dans la scène locale. A partir des données de position transmises, le modèle distant évalue la force de réaction, puis il la renvoie à travers le réseau (d'où la notion de bi-directionnalité) vers le modèle local qui lui-même la transmet au système à retour d'effort (Figure 1.2, page 21).

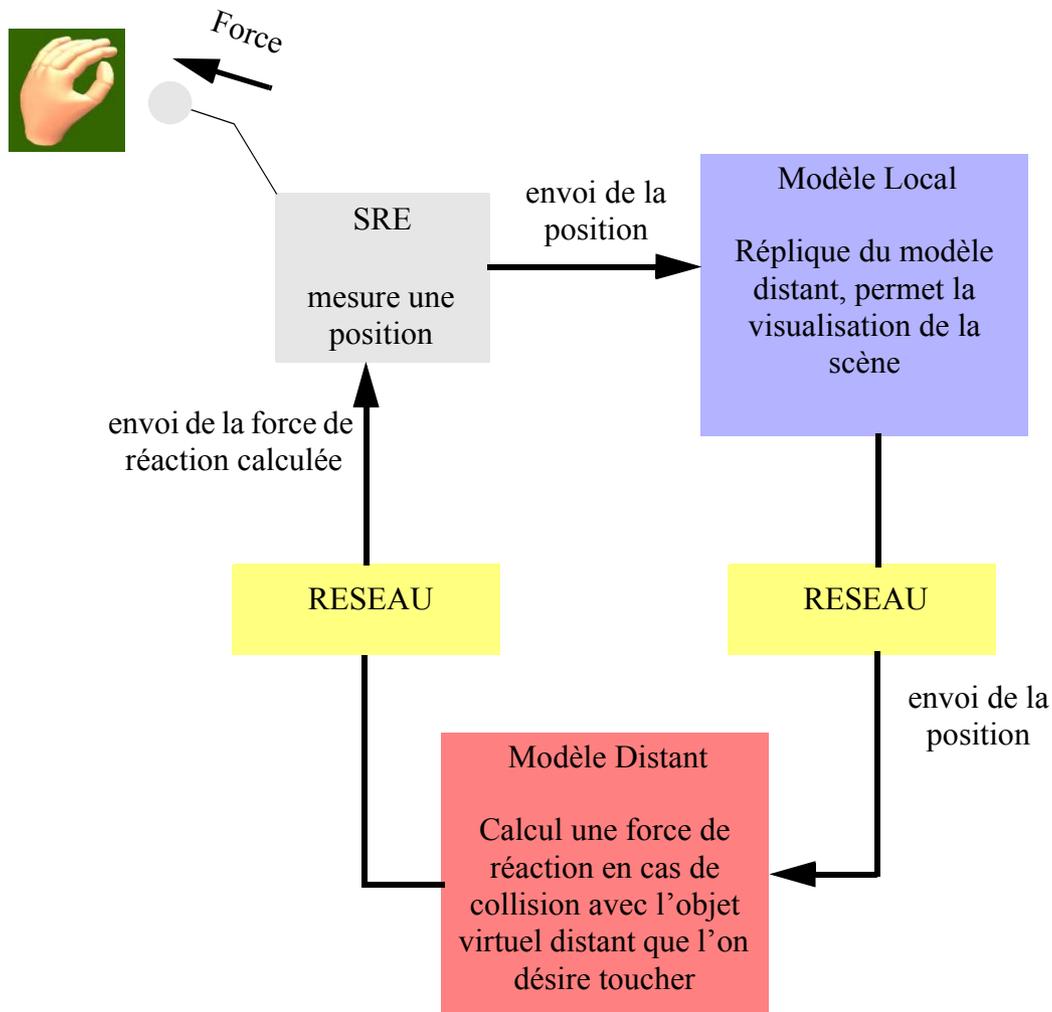


figure 1.2 : synopsis de la boucle haptique pour un toucher distant, elle doit s'effectuer en 1ms.

1.1.3 Position du problème

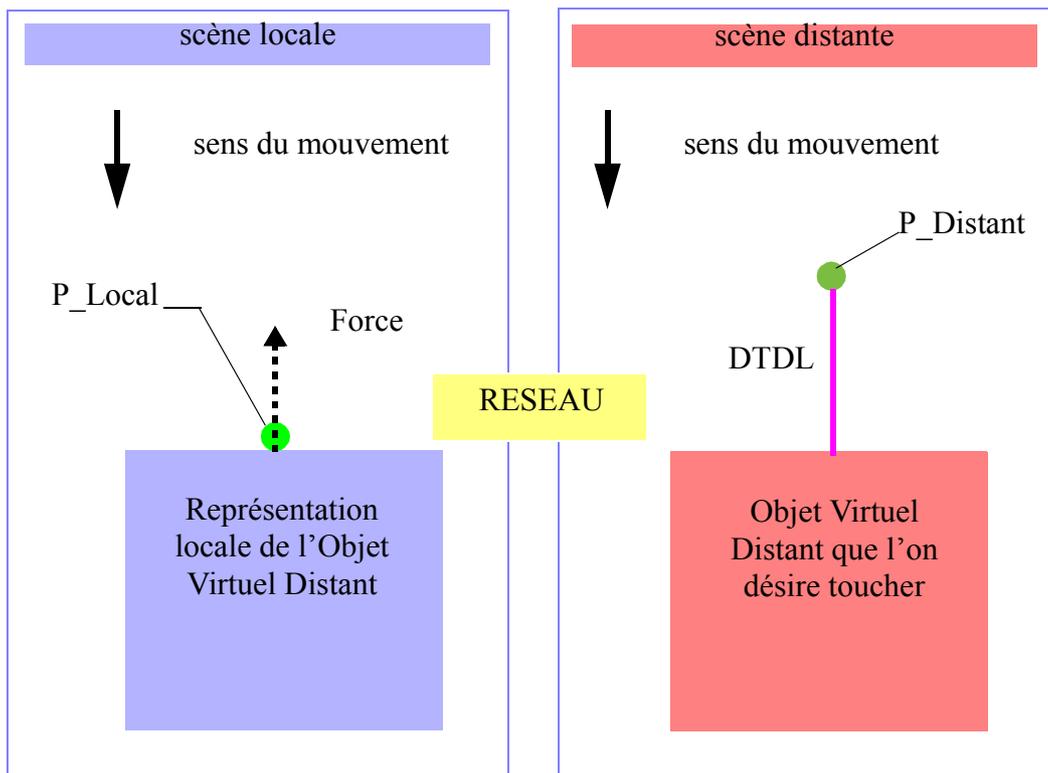
Le retour d'effort via les réseaux est un instrument important dans les systèmes robotiques répartis, utilisant des systèmes maître/esclave connectés à des appareils haptiques pour des simulations de réalité virtuelle à distance. Dans ce type d'application, les implémentations standards réparties nécessitent le transfert des forces et des données cinématiques à travers un lien de communication reliant les ordinateurs.

La latence incompressible des réseaux, réduit la largeur de bande efficace de la boucle haptique. En fait la boucle haptique dans le cas d'un toucher distant ne peut pas s'effectuer en 1ms (fréquence 1 kiloHertz) car la transmission de la position de la main du manipulateur vers le modèle distant introduit un retard

proportionnel au délai réseau. De même la transmission de la force calculée par le modèle distant introduit un retard toujours proportionnel au temps de passage des données cinématiques à travers le réseau.

En fait le modèle distant reçoit les données de position avec un retard de δ ms variant de 1 à 500 ms selon le type de réseau et selon la distance séparant l'objet virtuel distant du poste local d'opération.

On ne peut donc pas "nourrir" le système à retour d'effort avec des données pertinentes. En fait, il existe une distance entre la position de la main du manipulateur et la position distante de cette main dans la scène distante (Figure 1.3, page 22).



Légende :

P_Local : position de la main du manipulateur du SRE, dans la scène locale.

P_Distant : position de la main du manipulateur du SRE, dans la scène distante.

Il existe une distance DTDL entre les deux positions P_Local et P_Distant, car la position de la main dans la scène distante correspond à la position de la main dans la scène locale à la date $t-\delta$. δ correspond au délai introduit par le réseau.

figure 1.3 : Existence d'une distance entre la position de la main du manipulateur dans la scène distante et la position de la main dans la scène locale.

Le but de l'étude est de réduire au maximum cette distance directement dépendante de la vitesse de la main de l'utilisateur et du délai réseau. L'approche dite "WaveTransform" proposée par John Wilson [WIL 99] consiste à gérer l'instabilité de la boucle haptique (homme-système à retour d'effort-réseau-retour de la force dans la main du manipulateur) lors d'un toucher distant, en adaptant la vitesse de la main du manipulateur du SRE au délai réseau, en introduisant une contrainte de viscosité directement dépendante du délai réseau (Section 2.4, page 35).

Nous avons souhaité recourir à une autre approche afin de laisser le manipulateur libre de ses mouvements, et donc de sa vitesse de déplacement dans l'espace de travail, pour un délai réseau allant jusqu'à une centaine de millisecondes. Cette approche se base sur notre capacité à prédire le mouvement de la main de l'utilisateur. Comme l'illustre la Figure 1.3, page 22, on ne dispose au niveau du modèle distant que de la mémoire du mouvement de la main du manipulateur et ceci jusqu'à une date $t-\delta$, où δ correspond au délai réseau. On se propose de se servir de ces données reçues afin de prédire la position actuelle de la main de l'utilisateur dans le modèle local qui est une réplique du modèle distant.

1.2 Présentation du document

En premier lieu, nous proposons une présentation généraliste du "monde de l'haptique" et de ses applications diverses dans les domaines de la robotique et de la réalité virtuelle. Nous nous attachons également à présenter l'approche "Wave transform" de John Wilson.

Puis nous présentons la démarche suivie dans l'élaboration d'un algorithme de prédiction personnalisé. On se propose également d'étudier les prédicteurs directement dérivés de cet algorithme qui doivent permettre de gérer l'instabilité du système à retour d'effort lors d'un télégeste.

Par la suite, nous mettons en exergue les défauts des prédicteurs proposés. Ces derniers amplifient de façon importante le bruit de quantification présent dans le signal numérisé par le SRE. Nous proposons donc des solutions de filtrage afin d'obtenir un signal prédit exploitable par le système à retour d'effort.

Ensuite, nous entrons dans une étude plus approfondie des caractéristiques principales des prédicteurs directement issus de l'algorithme personnalisé. Nous étudions donc les prédicteurs d'un point de vue purement traitement du signal afin de dégager des informations pertinentes d'amplification et de décalage temporel du signal prédit.

Puis nous faisons une présentation générale des supports techniques de télécommunication, sur lesquels évoluent des applications mettant en jeu le télégeste. Nous décrivons notamment les supports et l'architecture des moyens de communication mis en oeuvre dans le cadre de la télé-échographie [TER 01].

Nous continuons notre étude par une validation théorique du modèle de notre algorithme de prédiction. Nous présentons les résultats des différents prédicteurs dérivés de l'algorithme personnalisé de prédiction, en utilisant comme signaux d'entrée deux sinusoides ayant des fréquences différentes, afin de mettre en

évidence la dépendance des performances des prédicteurs et la fréquence maximale contenue dans le signal que l'on désire prédire.

Suite à la validation théorique du modèle, nous procédons à une évaluation pratique des prédicteurs proposés, en se servant du critère de l'erreur quadratique moyenne. Ce chapitre nous permet de mettre à l'épreuve les compétences des divers prédicteurs, dans le cadre d'un signal gestuel et plus précisément d'un geste d'écriture.

Enfin, nous présentons l'implémentation d'une application concrète à travers le "circuit automobile". Cette application consiste à parcourir un circuit fermé en un temps chronométré. Des pénalités temporelles dues aux sorties de route sont rajoutées au temps de parcours. La force retournée par le système à retour d'effort n'est pas calculée à partir de la position locale de la main de l'utilisateur mais à partir de la mémoire de cette position locale jusqu'à une date de $t-\delta$, ceci afin de respecter les conditions d'un télégeste.

Nous concluons notre étude en présentant un bilan des avantages et des inconvénients de l'approche que nous avons choisi de suivre. Nous proposons aussi des perspectives d'évolution du modèle proposé.

Chapitre 2

L'haptique : l'interaction avec des objets virtuels

Introduction. Notre capacité à nous exprimer dans les performances musicales, dans la peinture, dans la sculpture et dans la gymnastique dépend de notre habileté à réussir des touchers physiques et à apprendre par le biais des interactions. Encore aujourd'hui, dans le monde des ordinateurs, notre mode principal afin d'accéder à l'information reste étroitement lié au visuel et dépend directement d'informations abstraites tels que les mots, les diagrammes ou les images. Pire, pour entrer une information dans l'ordinateur nous sommes restreints à utiliser le clavier ou la souris.

Nous débutons ce chapitre, en définissant les concepts de proprioception, de la co-dépendance "capteur-effecteur" liée au sens du toucher, puis en présentant la force et le mouvement comme les signaux les plus importants dans le domaine de l'haptique. Nous poursuivons par une description des systèmes à retour d'effort. Ces derniers sont directement associés au concept de technologie du toucher. Ils permettent de sonder des objets virtuels. Ensuite, nous présentons un échantillon des diverses applications utilisant les systèmes à retour d'effort, et les avantages à

les utiliser comme modules d'entraînement grâce à leur capacité à simuler le réel. Enfin nous terminons en décrivant précisément la démarche de John Wilson, appelée "WaveTransform", qui permet de gérer l'instabilité, introduite par le délai réseau, de la boucle haptique (homme-système à retour d'effort-réseau-retour de la force dans la main du manipulateur) lors d'un télégeste.

2.1 La proprioception

Dans le monde réel, à chaque fois que l'on touche un objet, ce dernier impose des forces sur notre peau. Ces forces de contact, plus la posture et le mouvement des mains et des bras, sont des informations kinesthésiques transmises au cerveau. C'est ainsi que nous sentons les propriétés grossières des objets. Par contre, les récepteurs sensoriel de la peau, transmettent des informations tactiles, telles que les variations spatiales et temporelles des distributions des forces des objets qui sont en contact avec la peau.

2.1.1 La force et le mouvement sont les plus importants signaux haptiques

Le terme "haptique" a été utilisé depuis plusieurs années par les chercheurs dans le domaine de la psychophysique humaine qui étudie la manière dont les gens utilisent leurs mains pour sentir et manipuler des objets [SAL 97]. Unique parmi nos modalités sensorielles, l'haptique compte sur l'action pour stimuler nos perceptions.

Un élément important de notre capacité à visualiser, à se souvenir et à établir des modèles cognitifs de la structure physique d'un environnement provient des interactions haptiques avec les objets qui forment notre environnement. Les sens kinesthésiques, de force et cutanés combinés à la capacité de mouvement nous permettent de sonder, de percevoir et de déplacer les objets de notre environnement. Les informations, qui correspondent à savoir comment un objet bouge en réponse à une force qui lui est appliquée et l'augmentation de la force que l'on doit déployer pour le bouger, nous donnent des indices sur la géométrie (forme, position, identité), sur les attributs (contrainte, friction, texture, etc) et les événements (contrainte, changement, contact, glissement) dans l'environnement. A l'inverse des autres sens, les interactions haptiques sont duales : capteur et effecteur. Un travail contrôlé peut être accompli sur des objets dynamiques et ce travail peut être modulé afin d'accomplir une tâche.

2.1.2 Capteur-effecteur : une Co-Dépendance

Pour sentir la forme d'une tasse nous ne nous contentons pas d'un simple toucher rapide de l'index, nous attrapons l'objet et nous le manipulons, laissant courir nos doigts sur sa surface et autour de ses formes afin de nous construire une image mentale de la tasse. Cette co-dépendance entre manipuler et percevoir est le facteur principal qui permet de comprendre comment les êtres humains interagissent aussi profondément avec le monde physique [SRI 97].

Les images haptiques sont composées en même temps d'informations tactiles et kinesthésiques. Un environnement virtuel n'a pas besoin d'être aussi parfait que la réalité, il lui suffit d'être en adéquation avec les capacités et les limites

sensorielles et motrices de l'être humain. Par exemple, il suffit de projeter une série d'images à la cadence de 30 images par seconde pour que l'impression de continuité soit atteinte. Il est assez difficile de quantifier les capacités haptiques humaines qui déterminent les spécifications des interfaces haptiques, ceci à cause de la multiplicité des sous-systèmes et de la nature sensorio-motrice des tâches à accomplir. On sait que le système tactile peut résoudre des vibrations supérieures au kilohertz, avec des amplitudes de l'ordre du micron détectables autour des 250 Hz. La résolution kinésique de la sensation de position du bout des doigts est de l'ordre du mm, avec une capacité à discerner des différences de l'ordre de 10% pour la vitesse et de 20% pour l'accélération. La bande passante du système moteur, pour des mouvements contrôlés, est inférieure à 10 Hz, et la force maximale exercée par les doigts humains est comprise entre 50 et 100 newtons, selon les individus.

2.1.3 Les mondes virtuels

La génération de signaux haptiques afin de créer des objets virtuels requièrent quatre capacités:

- (1) le suivi du mouvement de l'utilisateur.
- (2) la détection de collision entre le point ghost et des objets virtuels.
- (3) le calcul de la force de réaction en réponse au contact ou au mouvement.
- (4) le renvoi de la force calculée dans la main de l'utilisateur.

2.2 Le Système à Retour d'Effort (SRE)

2.2.1 La technologie du toucher

L'haptique est une nouvelle technologie, elle appartient au domaine des appareils interfacés avec un ordinateur. Elle promet d'apporter de profonds changements dans la façon humaine d'interagir avec des idées d'information et de communication [MAS 93]. Les progrès récents dans la technologie des interfaces d'ordinateur nous permet dorénavant de toucher et de manipuler des objets imaginaires créés par ordinateur en simulant le toucher réel.

Grâce à cette technologie nous pouvons nous asseoir derrière un ordinateur et toucher ce qui existe uniquement dans "l'esprit" de cet ordinateur [MAS 94]. Ces interactions peuvent être aussi simples que toucher un mur ou un bouton, ou être aussi complexes que de réaliser un acte chirurgical.

Le SRE Phantom permet aux utilisateurs de ressentir les forces d'interactions qui agissent lorsqu'ils touchent des objets avec le bout d'un stylo ou avec le bout de l'index.

2.2.2 L'interface haptique Phantom: un appareil pour sonder les objets virtuels

Cet appareil permet d'interagir avec une grande variété d'objets virtuels et peut être utilisé pour les contrôler à distance.



figure 2.1 : Le système à retour d'effort Phantom de chez Sensable.

L'un des objectifs principal, en recherche robotique, est traditionnellement de développer des systèmes autonomes, qui opèrent sans aucune interaction humaine. Toutefois, les systèmes robotiques contrôlés par l'humain ont repris de l'intérêt depuis l'avancement des interfaces robot-humain. Ces nouveaux systèmes promettent d'augmenter les capacités humaines, en augmentant la force, en améliorant la dextérité, en augmentant les sens, en projetant les utilisateurs humains dans des environnements abstraits ou éloignés.

Ces interfaces permettent l'interaction avec des environnements virtuels ou réels éloignés de l'utilisateur. Le SRE (Figure 2.1, page 28) capture le mouvement de la main de l'utilisateur et peut renvoyer de façon active une force dans les doigts de l'utilisateur qui tiennent le SRE, créant ainsi une illusion de contact avec des objets virtuels. Les interactions avec ces environnements virtuels sont nombreuses, on peut créer des contraintes de mouvement, des sensations de contact, des déformations d'objets, des sensations de friction, etc...

2.2.3 Trois critères importants pour obtenir une interface efficace

- Premièrement : un espace libre doit être ressenti comme tel.

Les utilisateurs ne doivent pas être encombrés par l'appareil. A savoir, l'appareil ne doit renvoyer aucune force dans la main de l'utilisateur si celui-ci se déplace dans un environnement virtuel libre. Il en résulte pour le SRE Phantom des frictions du bras inférieures à 0.1 Newton. L'inertie du bras est telle que l'utilisateur ne ressent pas plus de 100 gr de poids quand il bouge ce dernier dans un espace virtuel libre.

- Secondement : les objets solides doivent être perçus comme durs.

L'un des paramètres d'un système à retour d'effort est la dureté maximale de la surface des objets virtuels qu'il est capable de représenter. Ainsi le SRE Phantom est capable de renvoyer une dureté maximale de 35 Newton par cm. La plupart des utilisateurs sont convaincus de la rigidité d'un objet à partir d'un retour d'environ 20 Newton/cm. La dureté maximale que l'on peut atteindre dépend non

seulement de la fréquence naturelle de l'appareil mais aussi de la résolution des capteurs, des effecteurs et de la fréquence de la boucle haptique.

- Troisièmement : les contraintes virtuelles ne doivent pas être facilement saturées.

Il n'y a rien de plus perturbant que de s'appuyer contre un mur et de passer au travers. Dans le monde virtuel, les murs doivent également être solides. La force maximale exercée par les doigts d'une main humaine est de l'ordre de 40 Newton [SUT 89], mais durant une opération précise peu de personnes peuvent dépasser les 10 Newton, le phantom est capable de renvoyer et de subir au maximum une force de 10 Newton, variant selon les modèles. En fait, la force moyenne au cours du temps renvoyée lors d'une opération normale est de l'ordre de 1 Newton.

Dans sa forme la plus simple, le SRE Phantom peut être vu comme une transmission entre trois moteurs électriques pourvus d'encodeurs, et la main humaine. Les coordonnées spatiales x , y et z de la main sont suivies par les encodeurs, et les moteurs contrôlent les forces F_x , F_y et F_z renvoyées dans la main de l'utilisateur. De plus, comme la main de l'utilisateur est représentée dans le monde virtuel par un simple point, les collisions et les forces d'interactions avec l'environnement virtuel peuvent être aisément calculées.

Le phantom a été conçu afin d'obtenir des matrices de transformations entre les rotations des moteurs et les translations du point virtuel (appelé "point ghost"), représentant la position de la main de l'utilisateur dans l'espace virtuel de travail, qui soient pratiquement diagonales.

2.2.4 Les observations sur la perception.

Dans un sens, le fait de représenter le bout d'un doigt de l'utilisateur par un unique point (le point ghost), qui nous le rappelons est en interaction avec l'environnement virtuel, accroît la résolution du sens du toucher. Quand une personne manipule un objet dans le monde réel, la résolution effective est par conséquent limitée par la taille de la surface de contact de son doigt avec l'objet. En utilisant le SRE phantom, cette surface de contact peut être diminuée autant que le permet la quantification des données.

Les utilisateurs des SRE ont pu percevoir l'importance du lien, qui existe entre le sens du toucher, de la vue et de l'ouïe, lorsqu'ils naviguent dans un environnement virtuel.

2.2.5 Explorer les nouveaux mondes de l'interaction haptique

Les chercheurs ont déjà utilisé le SRE phantom afin de présenter une large variété d'objets à toucher et à manipuler. Des formes, aussi simples qu'une sphère ou complexes comme le modèle d'une navette spatiale, peuvent être touchées et explorées avec le SRE.

De plus en plus, les humains vont interagir avec des données visuelles en trois dimensions générées par ordinateur. La prolifération explosive des accélérateurs graphiques 3D temps-réel, promet de nourrir nos ordinateurs avec des scènes visuelles 3D réalistes. Les utilisateurs de la conception assistée par ordinateur, de visualisation de données sont très intéressés par la possibilité de toucher et de manipuler des objets réels ou virtuels (proches ou distants).

2.3 Les diverses applications

L'haptique forme la part essentielle de nos interactions avec le monde réel. Ainsi toute application de réalité virtuelle qui implique une simulation du monde réel bénéficie des interactions haptiques, spécialement si elles sont combinées avec des dispositifs visuels et auditifs.

2.3.1 L'entraînement visant à opérer des patients virtuels

L'un des premiers domaines d'application de la technologie de l'haptique est l'entraînement afin de réaliser des tâches dans le monde réel.

En procurant un outil permettant d'interagir avec des environnements virtuels, le SRE phantom offre la possibilité de développer des systèmes d'entraînement reconfigurable à souhait.

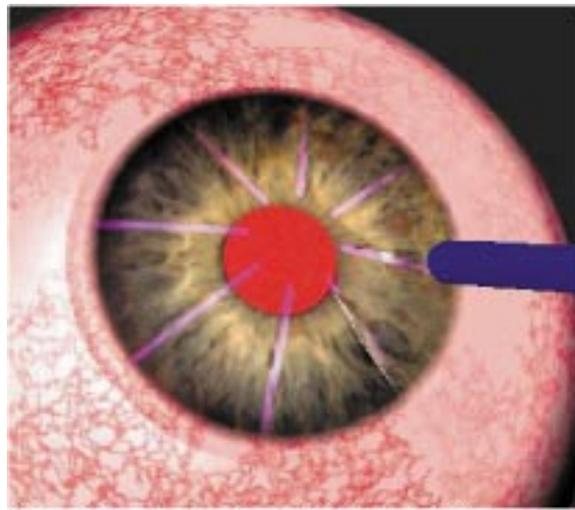


figure 2.2 : Modèle virtuel 3D d'un oeil humain, incisions faites au bistouri.

Des chercheurs de l'université du Colorado ont simulé une opération chirurgicale de l'oeil, spécialisée dans la réduction de la myopie, nommée kératotomie radiaire (Figure 2.2, page 30). Ceci permet aux chirurgiens de s'entraîner sur un modèle virtuel de l'oeil. Cet entraînement peut être réalisé à l'infini puisqu'il suffit de réinitialiser le modèle virtuel pour recommencer l'opération. De plus toute erreur survenue pendant la procédure n'est nullement néfaste pour le patient car il s'agit d'un oeil virtuel.

Ce style de chirurgie peut s'élargir à d'autres procédures telles que des opérations chirurgicales du coeur [SAL 96], en utilisant un modèle virtuel tel que celui représenté Figure 2.3, page 31.

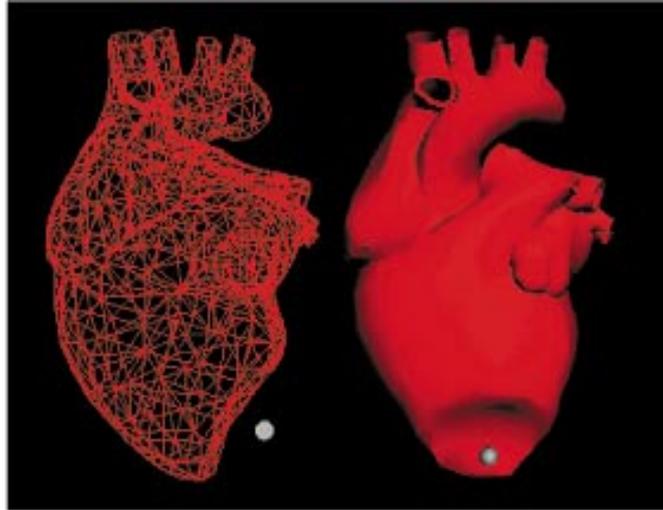


figure 2.3 : Modèle virtuel 3D d'un coeur humain.

Il existe également des débouchés dans la chirurgie dentaire. Teneo computing propose d'utiliser le SRE phantom couplé à un modèle virtuel de la dent et une interface permettant de simuler les différents outils du dentiste [RAN 99], [SCH 96]. En fait les étudiants en chirurgie dentaire disposent actuellement de dents artificielles en plastique pour s'entraîner et acquérir les techniques nécessaires à l'opération sur des patients réels. Ces dents artificielles sont des répliques assez grossières d'une dent réelle, une opération telle que l'extraction ne peut être correctement représentée par de tels modèles (absence d'écoulement de sang, par exemple). De plus ces modèles sont à utilisation unique. En créant un modèle virtuel de la gencive et de la dent, on peut obtenir un niveau de détails très proche de la réalité. De plus il suffit d'un simple clic de souris pour réinitialiser le modèle ce qui le rend réutilisable sans aucune limite, comme il est illustré sur la Figure 2.4, page 32.

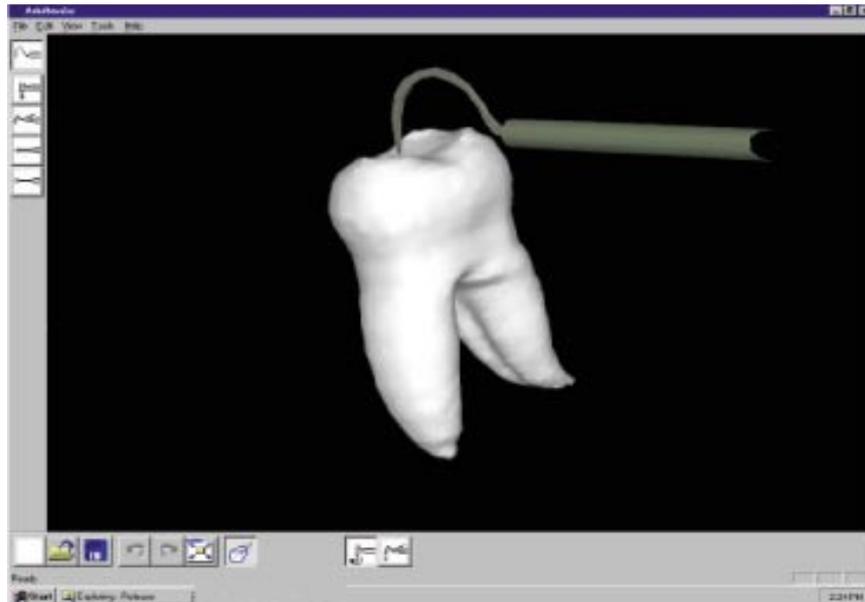


figure 2.4 : Station de commande haptique pour la chirurgie dentaire.

Le SRE phantom est tout à fait adapté dans son design pour simuler les outils permettant au praticien d'opérer. L'outil informatique apporte également la possibilité de quantifier les résultats du travail effectué sur la dent, il permet de multiplier les cas d'école, d'avoir un rendu haptique différent d'une zone saine et d'une zone cariée. Bref, l'étendue de l'apport de l'haptique dans ce genre d'entraînement est réellement très importante.

Les interactions haptiques peuvent être utilisées afin de procurer une interaction rapide et intuitive avec des données volumétriques.

En ajoutant l'haptique aux techniques de visualisation 3D existantes, on améliore la capacité des géoscientifiques (Figure 2.5, page 33) d'opérer en trois dimensions lors des procédés d'interprétation [AVI 99], [BRO 96], [LIN 98].

L'haptique apporte :

- la sélection directe et non-ambiguë d'entités individuelles (e.g, voxels) pour l'évaluation et la modification.
- l'utilisation du toucher afin de confirmer et d'augmenter la précision des modifications effectuées sur des données 3D.
- l'utilisation d'un nouveau mode sensoriel afin de mieux appréhender certaines données.

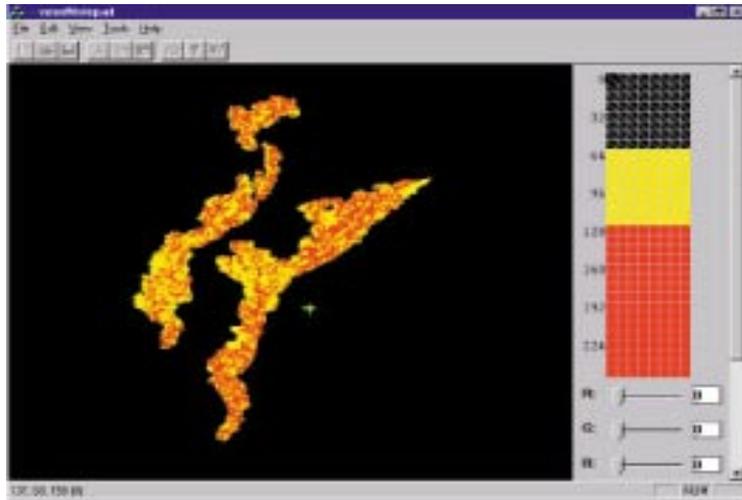


figure 2.5 : Données géophysiques 3D.

2.3.2 Le télégeste

Il s'agit de relier le SRE à un système mécanique distant (par exemple un autre SRE), les deux dispositifs sont reliés par le réseau. En fait, on cherche à télémanipuler un robot, un bras Phantom ou tout autre appareil qui permettrait de transporter le geste.

La télé-échographie

On cherche à pratiquer un examen échographique en ayant pour problématique : le patient et le praticien ne sont pas sur le même lieu géographique. Le patient se trouve dans un lieu appelé site distant, cet endroit est muni d'un robot télécommandé, le praticien est dans un lieu appelé site local, cet endroit est équipé d'un système à retour d'effort qui permet de commander le robot à travers un réseau de télécommunication [TER 01].

Le dispositif peut être décrit de la façon suivante : une sonde virtuelle est montée sur le dispositif principal d'interface. La sonde réelle est placée sur un robot du site distant [BEL 02a]. Les informations de position et de force sont transmises de manière bi-directionnelle par le réseau de télécommunication. L'opérateur expert peut déplacer la sonde virtuelle pour contrôler la sonde réelle distante, action principalement basée sur les images échographiques et l'information de la force de réaction qu'il reçoit par le biais du réseau.

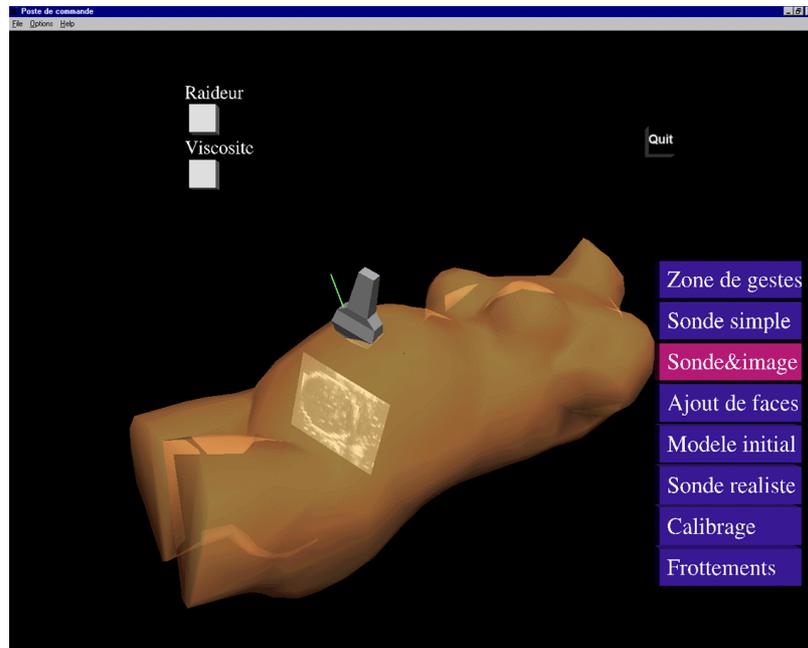


figure 2.6 : Station de commande haptique pour la télé-échographie.

Du côté praticien, la station de commande haptique (Figure 2.6, page 34) a été développée pour recréer un environnement réaliste, il est équipé d'une commande de sonde plus fine que la sonde réelle montée sur le robot du lieu distant.

D'autres applications ont été développées pour expérimenter et analyser les effets perturbateurs engendrés par l'introduction d'un réseau.

L'escrime virtuelle

Il s'agit d'un jeu développé au sein de l'équipe "gestes fins", par Alexandre Vienne, Bernard Hennion et Agnès Guerraz, à FT R&D [BEL 01]. Cette application permet de relier en vis à vis deux bras Phantom. Ces derniers sont reliés l'un à l'autre par le réseau télécom. Chaque utilisateur peut piloter un personnage et son épée, ce qui permet de simuler un combat d'escrime entre deux utilisateurs de SRE et ceci à distance (Figure 2.7, page 35).

La maquette a été écrite en Java, et fonctionne sur le réseau LAN, avec le protocole UDP, asynchrone, sans garantie de réception de message. Le temps d'un aller-retour est inférieur à 10 ms.

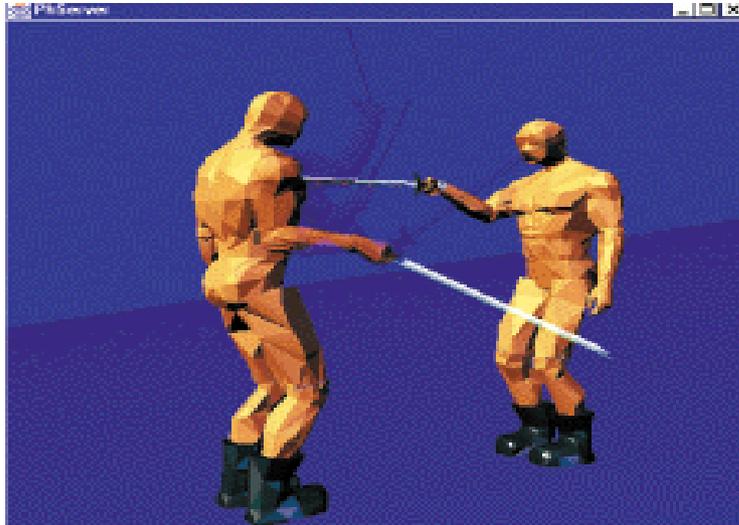


figure 2.7 : Interface du jeu d'escrime

2.4 Wave Transform : une autre approche

Il existe déjà une approche afin de minimiser les perturbations induites par le délai réseau dans un télégeste. Un premier travail fut effectué par Neville Hogan traitant de la difficulté de préserver la stabilité du système de contrôle (du manipulateur) lors d'un toucher de surface [HOG 88]. Robert Anderson et Mark Spong se sont servis des travaux de Neville Hogan afin de résoudre le problème suivant :

Comment maintenir la stabilité d'un télégeste en présence d'un délai réseau [AND 89] ?

Ainsi ils ont proposé une loi de contrôle du système :

$$\begin{aligned} F_{md}(t) &= F_s(t - \delta) + v_m(t) - v_{sd}(t - \delta) \\ v_{sd}(t) &= v_m(t - \delta) + F_{md}(t - \delta) - F_s(t) \end{aligned} \quad (\text{EQ 1})$$

- v_m : vitesse du maître.
- v_{sd} : vitesse "désirée" de l'esclave.
- d : délai réseau.
- F_{md} : force "désirée" exercée par le maître
- F_s : force rendue par l'esclave

Ceci revient à adapter la vitesse de l'esclave ainsi que la force exercée par le manipulateur en fonction du délai. En fait, plus le délai réseau augmente et plus les contraintes de mouvement appliquées au manipulateur seront grandes, afin de conserver la stabilité du système.

Cette méthode a été implémentée par John Wilson [WIL 99] et Robert Kline [KLI 99] dans le développement d'une interface permettant d'incorporer des systèmes à retour d'effort dans des applications distantes (cette réalisation a été

initialisée par une commande de U.S Army STRICOM dans le cadre du projet PHASE II SBIR).

Cette approche est souvent prise comme base dans l'élaboration de codeurs pour le télégeste. Niemeyer et Slotine ont ainsi développé une théorie s'inspirant directement de l'approche proposée par Anderson et Spong et utilisant les "waves variables" afin de stabiliser les systèmes de contrôle de robot en présence de délai réseau [NIE 97].

M. Saghir a proposé une technique afin de stabiliser une téléopération via internet [SAG 01] en mixant l'approche "Wave Transform" et la prédiction de Smith. Il a présenté une thèse à l'école George W. Woodruff of Mechanical Engineering.

Une autre approche est proposée par H. Arioui qui finalise sa thèse au Laboratoire des systèmes complexes. Sa thèse s'intitule : "Théorie de la commande des dispositifs à retour haptique avec un environnement virtuel en présence de retard de transmission". Il se sert des travaux de O. Smith [SMI 59] afin d'établir une loi de contrôle permettant de limiter l'instabilité de la boucle haptique lors d'un télégeste.

La fonction de transfert du système global d'une simulation haptique stable est donnée par la relation suivante :

$$\frac{F_e(s)}{F_h(s)} = \frac{\frac{1}{s}M(s)E(s)e^{-\delta_1}}{1 + E(s)\left(\frac{1}{s}M(s) + C(s)\right)} \quad (\text{EQ 2})$$

s correspond à la variable de la transformée de Laplace, Fe correspond à la réponse retardée du système et Fh la réponse prédite. M(s) correspond à l'appareil haptique (le maître), E(s) correspond à l'environnement virtuel distant (esclave), C(s) correspond au couple virtuel.

Le but principal de la loi de contrôle basée sur la prédiction de Smith est d'éliminer la réponse Fe (force de retour) et de la remplacer par une réponse non retardée [ARI 02].

2.5 Conclusion

Le domaine de l'haptique est depuis quelques années en pleine expansion. Les systèmes à retour d'effort sont des outils permettant une meilleure adaptation aux mondes virtuels en trois dimensions. Ils peuvent apporter beaucoup dans la formation des médecins tels que les chirurgiens, les dentistes, les vétérinaires. Les applications qui découlent de l'haptique offriront demain l'opportunité aux mal-voyants d'avoir un outil d'interaction avec le monde informatique. Par ailleurs, un projet d'apprentissage de l'écriture par le biais de SRE pour les mal-voyants a vu le jour chez FT R&D. Il existe en fait un grand nombre de débouchés pour l'haptique et ses applications dans la vie de tous les jours.

Chapitre 3

Le modèle d'algorithme de prédiction

Introduction. Afin de réduire l'instabilité induite par la latence du réseau dans la force renvoyée par le système à retour d'effort (SRE) lors d'un toucher distant, on se propose de prédire le mouvement de la main du manipulateur du SRE.

Le but de l'étude est de construire un signal $\hat{s}(n)$ en ayant à notre disposition que le signal $s(n-\delta)$ tout en minimisant l'erreur de prédiction $E(n)$ évaluée de la façon suivante : $E(n) = s(n) - \hat{s}(n)$.

Autrement dit, on désire prédire ce que sera le signal à l'instant n sachant que l'on dispose uniquement de la mémoire de ce signal jusqu'à l'instant $n-\delta$. Le δ correspond à la latence incompressible induite par le réseau. Nous allons donc discuter des différentes politiques de prédiction que nous avons analysé.

Dans un premier temps nous présentons dans ce chapitre, des politiques de prédiction couramment utilisées dans la littérature. Nous abordons en fait l'algorithme de prédiction linéaire LPC (Linear Predictive Coding) puis nous nous intéressons au filtrage de Kalman, et enfin nous proposons notre algorithme

personnalisé, directement dérivé du modèle de l'algorithme LPC. Ensuite nous ferons une présentation générale des divers prédicteurs dérivés du modèle de notre algorithme personnalisé.

3.1 Les politiques de prédiction

Il existe dans la littérature un grand nombre de modèles de prédiction. Les algorithmes les plus souvent utilisés sont du type LPC (Linear Prediction Coding) [TRE 82] ou du type filtrage de Kalman.

3.1.1 L'algorithme LPC

La forme du signal est prévue grâce à une combinaison linéaire d'échantillons antérieurs:

$$\hat{s}(n) = \sum_{i=1}^p a_i \times s(n-i) \quad (\text{EQ 3})$$

$$\hat{s}(n) = a_1 \times s(n-1) + a_2 \times s(n-2) + \dots + a_p \times s(n-p)$$

Ce type d'algorithme utilise les échantillons immédiatement antérieurs [MAK 75], il est basé sur le modèle des processus MA (Moving Average).

On considère le signal $x(t) = b_0 \times w(t) + b_1 \times w(t-1) + \dots + b_M \times w(t-M)$ où $w(t)$ désigne un processus aléatoire réel, centré, stationnaire, blanc de variance 1 et $\{b_m\}$ une suite de $(M+1)$ coefficients réels. Le processus ainsi construit apparaît comme la moyenne pondérée des $(M+1)$ dernières valeurs des entrées par la suite $\{b_m\}$. Tout se passe comme si on déplaçait cette suite de pondération devant le signal d'entrée. Pour cette raison, on parle de moyenne mobile, qui se dit en anglais Moving Average, ou MA et qui a d'ailleurs donné son nom à ce processus.

La solution que nous proposons s'inspire directement de l'algorithme LPC (EQ 6).

3.1.2 Le filtre de Kalman

Un autre algorithme utilisé fréquemment en estimation et plus précisément en prédiction, est le filtrage de Kalman [BER 98]. Le filtrage de Kalman permet de résoudre des problèmes de prédiction à un pas et des problèmes d'estimation de manière récursive dans le temps. Le processus doit être décrit par un modèle linéaire, représentation d'état, constitué d'une équation d'état et d'une équation d'observation ou de mesure.

$$\begin{aligned} \underline{X}_{k+1} &= \underline{A}_k \times \underline{X}_k + \underline{U}_k + \underline{G}_k \\ \underline{Y}_k &= \underline{C}_k \times \underline{X}_k + \underline{W}_k \end{aligned} \quad (\text{EQ 4})$$

\underline{X} est le vecteur qui représente l'état du système.

\underline{Y} est le vecteur qui représente les variables mesurées du système.

\underline{A} et \underline{U} représentent la matrice d'évolution.

\underline{G} représente le bruit du système.

\underline{W} représente le bruit de mesure.

Le bruit d'état \underline{G}_k et de mesure \underline{W}_k sont supposés blancs, centrés et de variances :

$$\begin{aligned}\Gamma_k &= E(\underline{G}_k \times \underline{G}_k^T) \\ \Omega_k &= E(\underline{W}_k \times \underline{W}_k^T)\end{aligned}\quad (\text{EQ 5})$$

On résout simultanément un problème de prédiction et un problème d'estimation. La récursivité se décompose en trois étapes :

- prédiction de \underline{X}_{k+1} à partir de la suite $\underline{Y}_0, \underline{Y}_1, \dots, \underline{Y}_k$. La valeur prédite sera notée $\underline{X}_{k+1/k}$ (Valeur prédite de \underline{X}_{k+1} connaissant les observations précédentes jusqu'à k).
- nouvelle mesure \underline{Y}_{k+1} .
- estimation de \underline{X}_{k+1} grâce à $\underline{X}_{k+1/k}$ et \underline{Y}_{k+1} . L'estimée sera notée $\underline{X}_{k+1/k+1}$.

Cette brève présentation suffit pour comprendre que le filtrage de Kalman s'appuie en fait sur deux équations : l'équation d'état et l'équation d'observation. Nous disposons assez facilement de l'équation d'état qui est une combinaison linéaire des positions successives de la main du manipulateur du système à retour d'effort. Le problème se situe dans l'élaboration de l'équation d'observation. L'utilisation du filtrage de Kalman est présentée en conclusion comme une perspective intéressante de l'évolution des modèles proposés.

3.1.3 L'algorithme personnalisé

Nous avons élaboré une forme de prédicteur s'appuyant sur la forme de l'algorithme LPC, il est de la forme :

$$\hat{s}(t) = \sum_{i=0}^p \frac{\delta^i}{i!} \times \frac{d^i}{dt^i} s(t - \delta) \quad (\text{EQ 6})$$

Dans cette relation, p représente l'ordre du prédicteur.

Le signal prédit à la date t, est ici construit comme une combinaison linéaire des dérivées successives du signal d'entrée retardé par le délai réseau δ [BEL 02b].

3.2 Présentation des différents prédicteurs

3.2.1 Le prédicteur d'ordre zéro

C'est le plus simple des prédicteurs, il se contente de garder la dernière valeur du signal reçu comme étant la plus pertinente. De ce fait le signal prédit est exactement le signal d'origine mais décalé dans le temps, le décalage est égal au délai réseau.

On en tire donc:

$$\hat{s}(n) = s(n - \delta) \quad (\text{EQ 7})$$

on en déduit alors l'expression analytique de l'erreur de prédiction:

$$e(n) = s(n) - \hat{s}(n) = s(n) - s(n - \delta) \quad (\text{EQ 8})$$

Les schémas suivant illustrent les propos tenus ci-dessus:

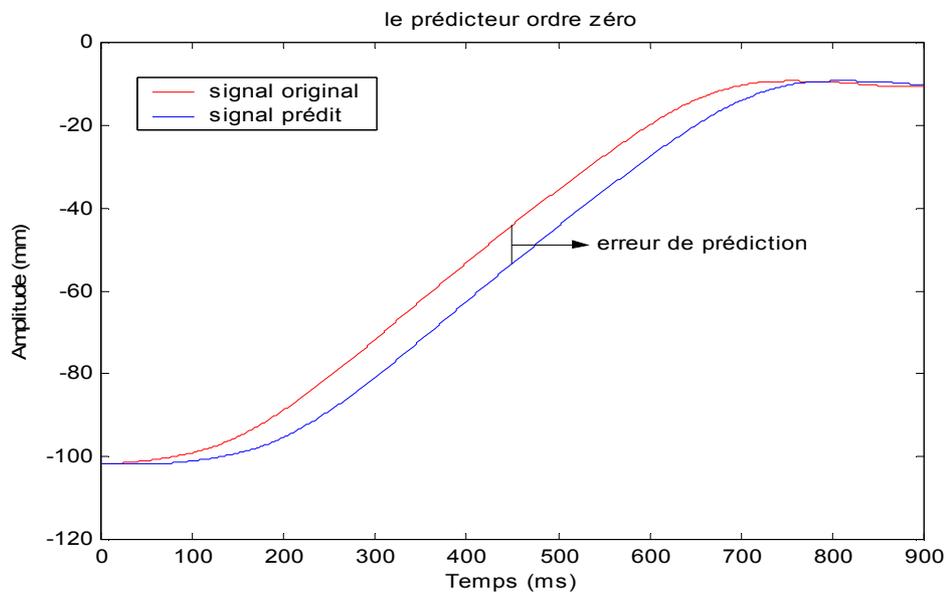


figure 3.1 : le prédicteur d'ordre 0.

On s'aperçoit donc que le délai réseau n'est pas traité et que l'utilisateur du SRE ne percevra pas les forces en temps voulu. Il souffrira du délai réseau.

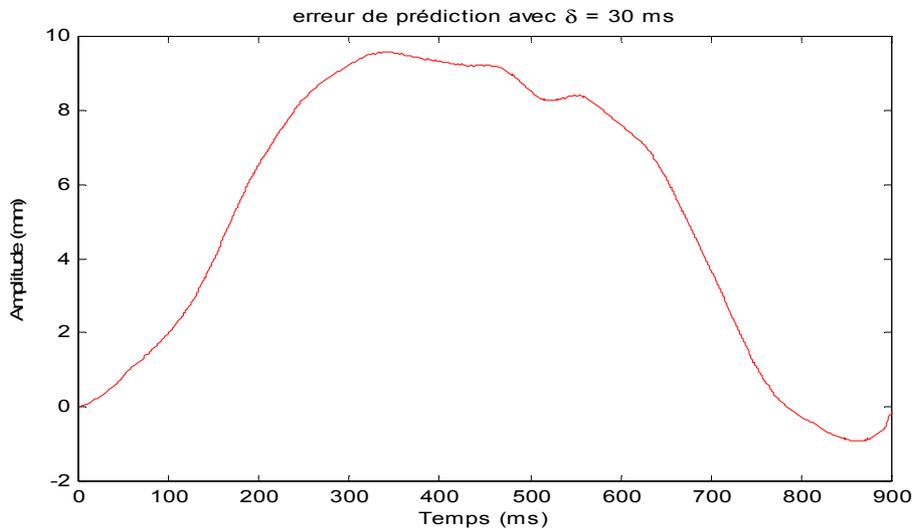


figure 3.2 : erreur de prédiction.

Le prédicteur d'ordre zéro correspond donc pratiquement à un traitement minimal des données, il a été introduit dans ce paragraphe afin de pouvoir déterminer le gain des autres prédicteurs en terme de minimisation de l'erreur de prédiction.

3.2.2 Le prédicteur d'ordre 1

Le signal prédit à partir du prédicteur d'ordre 1 est alors calculé à partir de la formule suivante:

$$\hat{s}(n) = s(n - \delta) + \delta \times v(n - \delta) \quad (\text{EQ 9})$$

Ceci revient donc à évaluer la pente instantanée de la tangente à la courbe représentant le signal à l'instant $n - \delta$ puis on multiplie cette pente par le délai et on rajoute ce produit à la valeur du signal à l'instant $n - \delta$.

La Figure 3.3, page 42 explique graphiquement la méthode de prédiction, exprimée par l'équation (EQ 9).

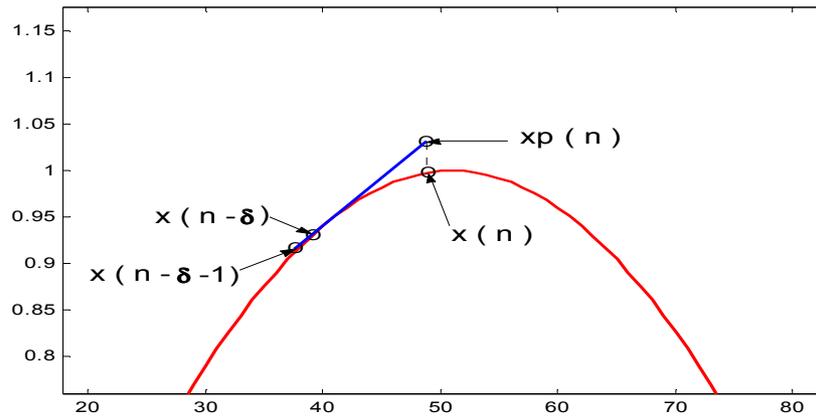


figure 3.3 : explication graphique de la prédiction d'ordre 1

On calcule la vitesse d'un signal $S(n)$ de la façon suivante:

$$v(n) = \frac{s(n) - s(n - \Delta n)}{\Delta n} \quad (\text{EQ 10})$$

On note que Δn correspond au pas d'échantillonnage du signal $S(n)$, ce pas est de 1ms, il en résulte :

$$v(n) = s(n) - s(n - 1) \quad (\text{EQ 11})$$

L'analogie avec l'algorithme LPC se fait de la façon suivante pour le prédicteur d'ordre 1:

$$\begin{aligned} \hat{s}(n) &= s(n - \delta) + \delta \times v(n - \delta) \\ \hat{s}(n) &= s(n - \delta) + \delta \times (s(n - \delta) - s(n - \delta - 1)) \\ \hat{s}(n) &= (\delta + 1) \times s(n - \delta) + (-\delta) \times s(n - \delta - 1) \end{aligned}$$

or l'algorithme LPC nous donne :

$$\hat{s}(n) = \sum_{i=\delta}^{\delta+1} a_i \times s(n - \delta)$$

par identification on obtient :

$$a_{\delta} = \delta + 1 \quad a_{\delta+1} = -\delta$$

Le calcul de la vitesse exprimé par la relation (EQ 11), peut être réalisé par le circuit suivant:

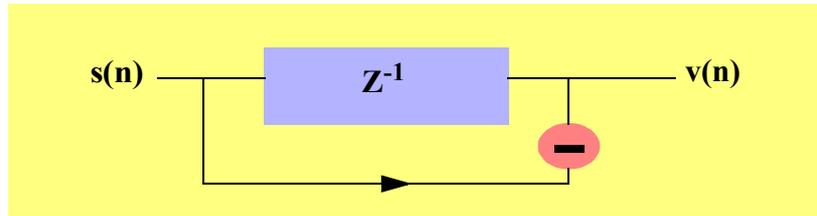


figure 3.4 : circuit permettant de calculer la vitesse.

Le signal émis par le dispositif du bras "phantom" est un signal numérique entaché d'un bruit de quantification. Ce bruit est un bruit haute fréquence car il est directement lié à la fréquence de la boucle haptique, et le signal d'entrée du système peut être vu comme l'addition d'un signal utile $S_U(n)$ et d'un bruit de quantification $b(n)$. Il en résulte donc que la vitesse est également de la forme:

$$V(n) = V_U(n) + W(n) \quad (\text{EQ 12})$$

Nous nous proposons de calculer la fonction de transfert du filtre qui permet d'évaluer la vitesse:

$$v(n) = s(n) - s(n-1) \quad (\text{EQ 13})$$

en passant par la transformée en Z (Section A, page 159) on obtient:

$$V(Z) = S(Z) - S(Z) \times Z^{-1} \quad (\text{EQ 14})$$

il vient alors pour la fonction de transfert:

$$H(Z) = \frac{V(Z)}{S(Z)} = 1 - Z^{-1} \quad (\text{EQ 15})$$

on en déduit alors la fonction de transfert isochrone par le changement de variable suivant: $Z = e^{j\omega T_e}$ ou T_e est la période d'échantillonnage, elle correspond à l'inverse de la fréquence de la boucle haptique.

D'où:

$$H(\omega) = 1 - e^{-j\omega T_e} = 1 - \cos\omega T_e + j\sin\omega T_e \quad (\text{EQ 16})$$

On peut maintenant calculer le module de la fonction de transfert:

$$|H(\omega)| = \sqrt{(1 - \cos\omega T_e)^2 + (\sin\omega T_e)^2} = \sqrt{2} \times \sqrt{1 - \cos\omega T_e} \quad (\text{EQ 17})$$

L'allure du gain en fréquence du filtre est représentée sur la Figure 3.5, page 44:

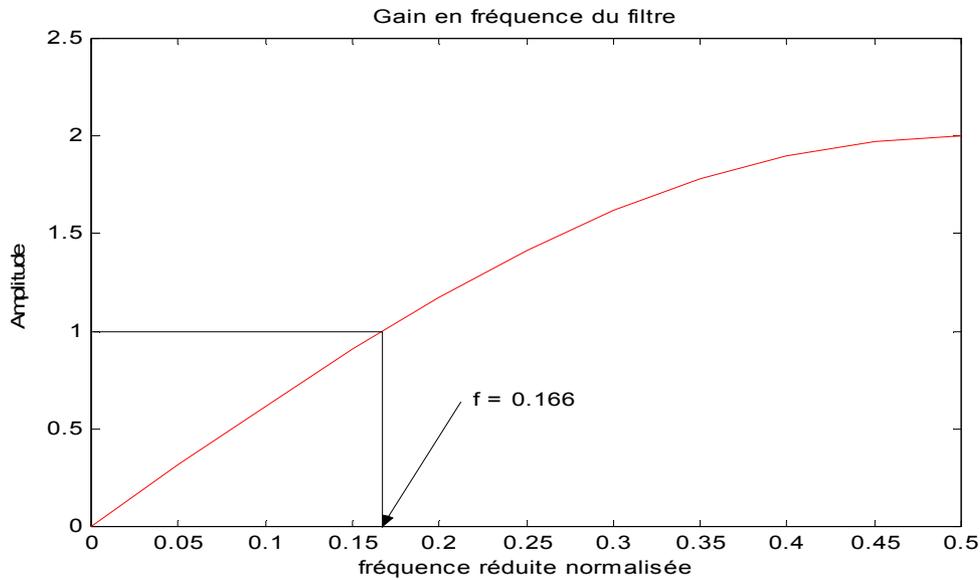


figure 3.5 : gain en fréquence du filtre permettant de calculer la vitesse.

On s'aperçoit alors que le filtre qui permet de calculer la vitesse est un filtre passe-haut. Il amplifie le spectre du signal d'entrée pour toutes les fréquences supérieures à $f > 166$ hz. Cette fréquence est solution de l'équation : $|H(w)| = 1$. De plus, il diminue l'amplitude de la partie basse fréquence du spectre du signal d'entrée. Or la partie significative du spectre du bruit de quantification est située dans les hautes fréquences.

Le signal de sortie du filtre est alors fortement bruité, comme on peut le constater sur la figure suivante:

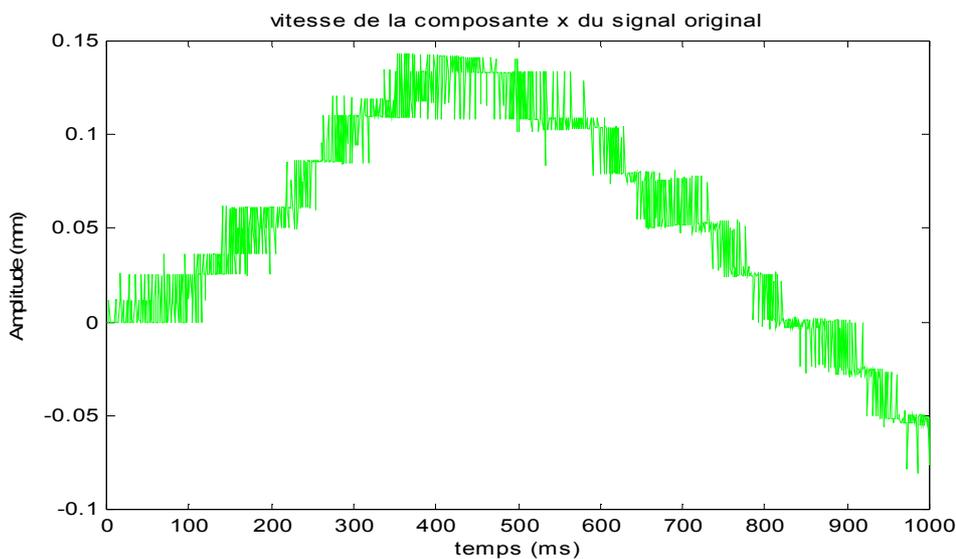


figure 3.6 : vitesse de la composante x du signal original.

Comme nous l'avons vu précédemment, le signal prédit par le prédicteur d'ordre 1 utilise la dérivée première autrement dit la vitesse. De plus cette vitesse est multipliée par le facteur δ qui correspond au délai réseau. Il devient donc très important de filtrer la sortie du filtre qui permet de calculer la vitesse. En fait, nous sommes tout particulièrement intéressés par la composante basse fréquence du signal de sortie $V(n)$.

Nous cherchons donc à isoler la composante $V_U(n)$ de l'équation (EQ 12) en éliminant $W(n)$. Nous avons alors estimé la composante basse fréquence de la vitesse en moyennant celle-ci sur une fenêtre d'une longueur de 10 échantillons puis en interpolant par une méthode d'interpolation linéaire entre les $N/10$ échantillons moyennés, on obtient $\widehat{V}_U(n)$. Cette estimation de la vitesse nous permet d'obtenir une estimation du bruit $W(n)$ (EQ 18).

$$\widehat{W}(n) = V(n) - \widehat{V}_U(n) \quad (\text{EQ 18})$$

Le calcul effectué pour l'obtention de l'estimation de la vitesse est un calcul fait a posteriori ce qui signifie que l'on attend d'avoir une portion du signal pour le moyenner et pour effectuer l'interpolation. Cette méthode d'évaluation est impropre au calcul temps réel du signal prédit $S_p(n)$. Les solutions de filtrage du bruit de quantification sont présentées dans la Section 4.2, page 61.

Nous avons alors estimé la densité spectrale du bruit de quantification, et nous l'avons analysée spectralement afin de déterminer le gabarit du filtre passe-bas que l'on appliquera sur la vitesse (Section 4.2.3, page 65).

3.2.3 Le prédicteur d'ordre 2

Comme pour le prédicteur d'ordre 1, on utilise le modèle de l'algorithme personnalisé, ce qui nous donne:

$$\hat{s}(n) = s(n - \delta) + \delta \times v(n - \delta) + \frac{\delta^2}{2} \times \Gamma(n - \delta) \quad (\text{EQ 19})$$

Nous avons vu dans le paragraphe précédent que la vitesse du signal est exprimée par la relation suivante:

$$V(n) = S(n) - S(n - 1) \quad (\text{EQ 20})$$

L'accélération est la dérivée de cette vitesse, on en tire donc :

$$\begin{aligned} \Gamma(n) &= v(n) - v(n - 1) \\ &= s(n) - s(n - 1) - (s(n - 1) - s(n - 2)) \\ &= s(n) - 2 \times s(n - 1) + s(n - 2) \end{aligned} \quad (\text{EQ 21})$$

Le calcul de l'accélération exprimé par la relation (EQ 21), peut être réalisé par le circuit suivant:

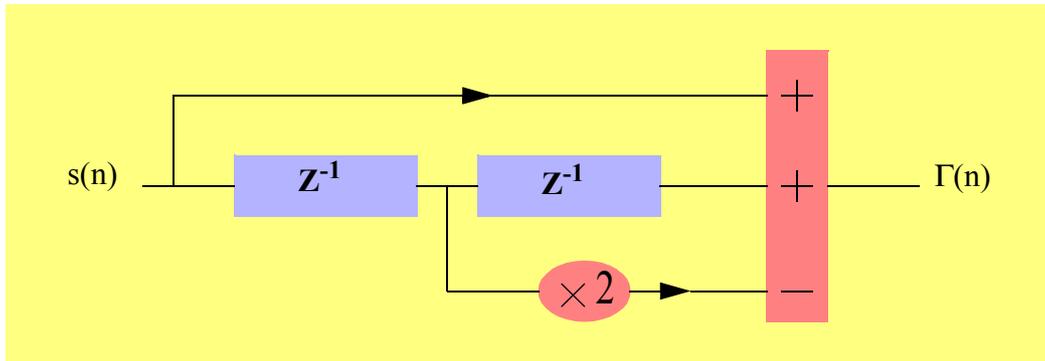


figure 3.7 : circuit permettant de calculer l'accélération

Comme pour le filtre calculant la vitesse, on se propose de calculer la fonction de transfert de ce filtre:

en passant par la transformée en Z on obtient:

$$\Gamma(z) = S(z) - 2 \times S(z) \times z^{-1} + S(z) \times z^{-2}$$

$$H(z) = \frac{\Gamma(z)}{S(z)} = 1 - 2 \times z^{-1} + z^{-2} \quad (\text{EQ 22})$$

on en déduit la fonction de transfert isochrone:

$$H(w) = 1 - 2 \times \cos wT_e + \cos 2wT_e + j \times (2 \times \sin wT_e - \sin 2wT_e) \quad (\text{EQ 23})$$

On en prend enfin le module:

$$|H(w)| = \sqrt{(1 - 2 \times \cos wT_e + \cos 2wT_e)^2 + (2 \times \sin wT_e - \sin 2wT_e)^2} \quad (\text{EQ 24})$$

L'allure du gain du filtre est représentée sur la figure suivante:

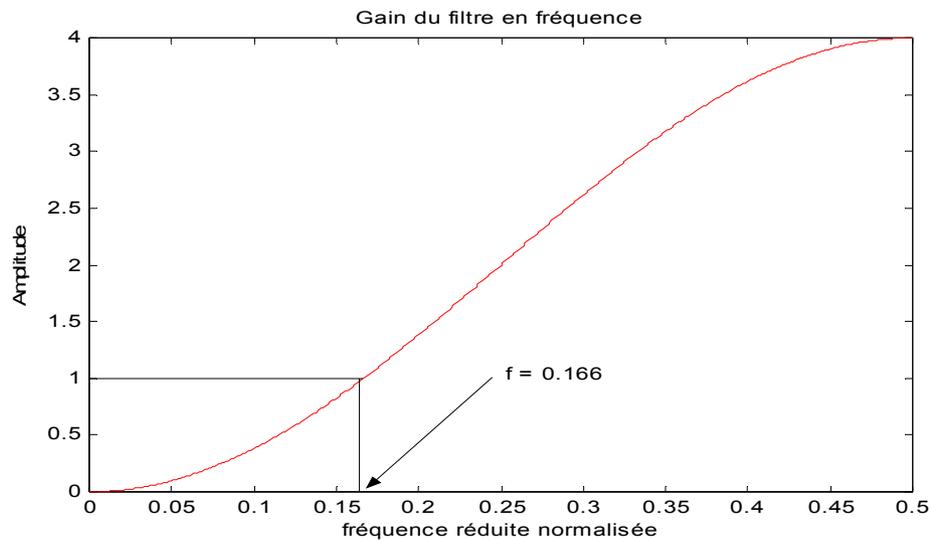


figure 3.8 : gain en fréquence du filtre permettant de calculer l'accélération.

On s'aperçoit que ce filtre amplifie le spectre du signal d'entrée pour toutes les fréquences supérieures à $f > 166$ Hz. Cette fréquence est solution de l'équation : $|H(w)| = 1$. A contrario il diminue l'amplitude de la partie basse fréquence du spectre. On note également que pour les hautes fréquences, le gain n'est plus de 2 comme pour la vitesse, mais de 4. Pour les basses fréquences, on note une atténuation encore plus forte que celle induite par le filtre permettant de calculer la vitesse. Or comme nous l'avons vu précédemment, le spectre du bruit de quantification se situe dans les hautes fréquences.

Le signal de sortie du filtre qui correspond à l'accélération est alors très fortement bruité:

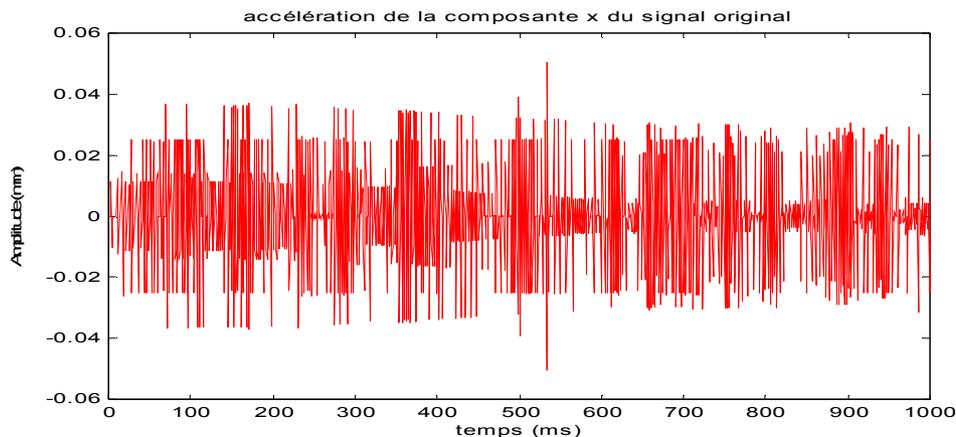


figure 3.9 : accélération de la composante x du signal original.

On s'aperçoit que le signal représenté Figure 3.9, page 47 est très chahuté, or le signal prédit est une fonction de l'accélération (EQ 19), de plus cette

accélération est multipliée par le facteur δ^2 ce qui amplifie énormément l'instabilité de l'accélération. La contribution de l'accélération au signal prédit doit être filtrée afin d'obtenir un signal lisse à la sortie. En fait comme pour la vitesse, ce qui nous intéresse c'est la composante basse fréquence de l'accélération $\Gamma(n)$. En utilisant la méthode du périodogramme moyenné, on évalue le spectre du signal représentant l'accélération, puis on détermine par l'analyse de ce spectre le meilleur filtre passe-bas qui nous permettra d'obtenir une composante lissée de l'accélération. Par conséquent on réalisera une contribution lissée de cette accélération au signal prédit (Section 4.2.4, page 66).

3.2.4 Le prédicteur d'ordre 3

Comme pour le prédicteur d'ordre 1, on utilise le modèle de l'algorithme personnalisé, ce qui nous donne:

$$\hat{s}(n) = s(n - \delta) + \delta \times v(n - \delta) + \frac{\delta^2}{2!} \times \Gamma(n - \delta) + \frac{\delta^3}{3!} \times J(n - \delta) \quad (\text{EQ 25})$$

Nous avons vu (EQ 21) que l'accélération du signal est exprimé par la relation suivante:

$$\Gamma(n) = s(n) - 2 \times s(n - 1) + s(n - 2) \quad (\text{EQ 26})$$

Le Jitter est la dérivée de cette accélération, on en tire donc:

$$\begin{aligned} J(n) &= \Gamma(n) - \Gamma(n - 1) \\ &= s(n) - 2 \times s(n - 1) + s(n - 2) - s(n - 1) + 2 \times s(n - 2) - s(n - 3) \quad (\text{EQ 27}) \\ &= s(n) - 3 \times s(n - 1) + 3 \times s(n - 2) - s(n - 3) \end{aligned}$$

Le calcul du jitter (dérivée tierce du signal) exprimé par la relation (EQ 26), peut être réalisé par le circuit suivant:

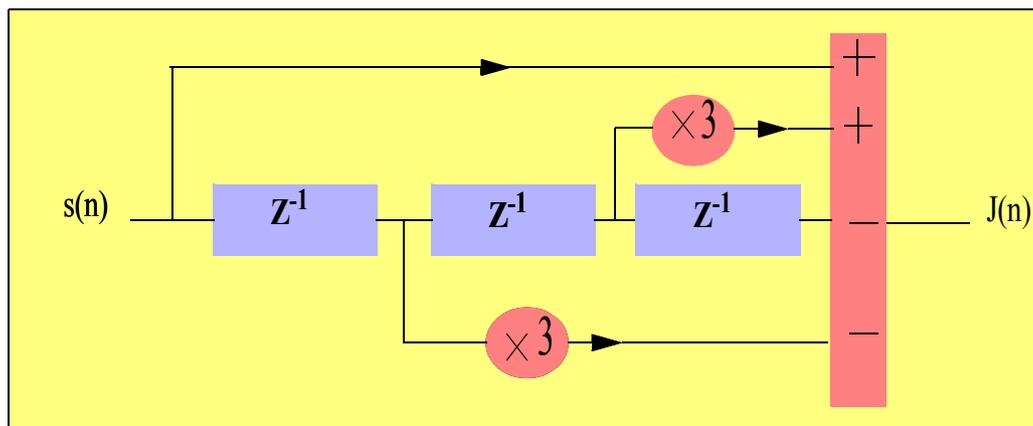


figure 3.10 : circuit permettant de calculer le jitter.

Comme pour le filtre calculant l'accélération, on se propose de calculer la fonction de transfert de ce filtre:

en passant par la transformée en Z on obtient:

$$J(z) = S(z) - 3 \times S(z) \times z^{-1} + 3 \times S(z) \times z^{-2} - S(z) \times z^{-3}$$

$$H(z) = \frac{J(z)}{S(z)} = 1 - 3 \times z^{-1} + 3 \times z^{-2} - z^{-3} \quad (\text{EQ 28})$$

on en déduit la fonction de transfert isochrone:

$$\text{Re}[H(w)] = 1 - 3 \times \cos wT_e + 3 \times \cos 2wT_e - \cos 3wT_e$$

$$\text{Im}[H(w)] = 3 \times \sin wT_e - 3 \times \sin 2wT_e + \sin 3wT_e \quad (\text{EQ 29})$$

On en prend enfin le module:

$$|H(w)| = \sqrt{(\text{Re}[H(w)])^2 + (\text{Im}[H(w)])^2} \quad (\text{EQ 30})$$

L'allure du gain du filtre est représentée sur la figure suivante:

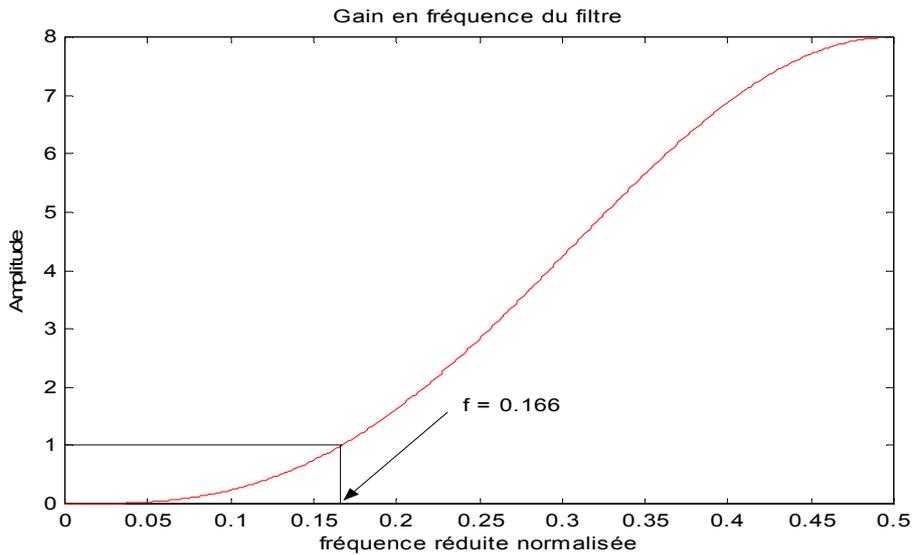


figure 3.11 : gain en fréquence du filtre permettant de calculer le jitter.

On s'aperçoit que ce filtre amplifie le spectre du signal d'entrée pour toutes les fréquences supérieures à $f > 166$ Hz. Cette fréquence est solution de l'équation : $|H(w)| = 1$. Comme pour le filtre permettant de calculer la vitesse, il diminue l'amplitude de la partie basse fréquence du spectre. On note également que pour les hautes fréquences le gain n'est plus de 4 comme pour l'accélération, mais de 8. De plus l'atténuation de l'amplitude des basses fréquences est encore plus importante que l'atténuation subie lors du calcul de l'accélération. Or le spectre du bruit de quantification est situé dans les hautes fréquences.

Le signal de sortie du filtre qui correspond au jitter (dérivée tierce du signal) est alors très fortement bruité:

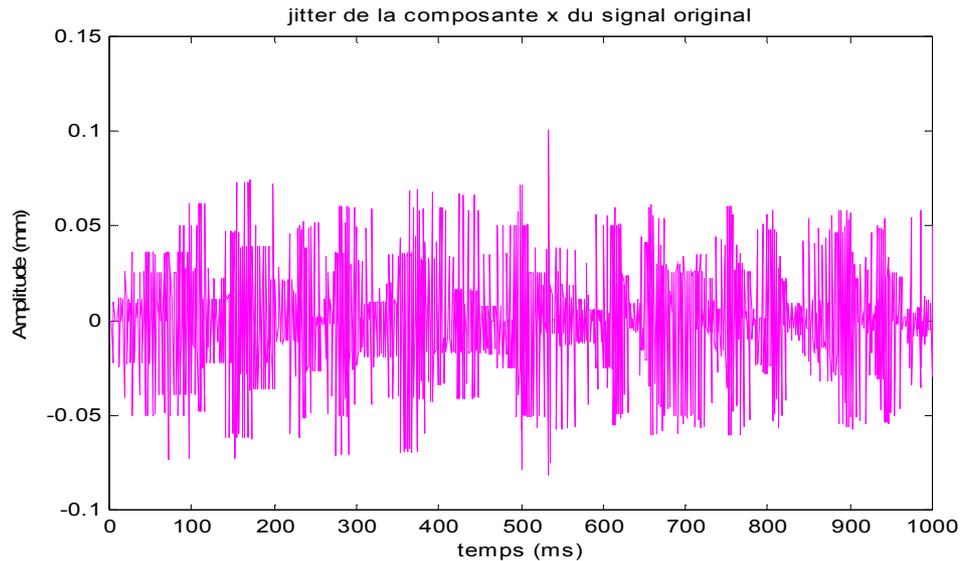


figure 3.12 : jitter de la composante x du signal original

On s'aperçoit que le signal représenté ci-dessus est très chahuté, or le signal prédit est une fonction du jitter (EQ 25). De plus cette dérivée tierce est multipliée par le facteur δ^3 ce qui amplifie énormément l'instabilité du jitter. La contribution du jitter au signal prédit doit être filtrée afin d'obtenir un signal lisse à la sortie. En fait comme pour la vitesse et l'accélération, nous sommes intéressés par la composante basse fréquence du jitter $J(n)$. En utilisant la méthode du périodogramme moyenné, on évalue le spectre du signal représentant le jitter, puis on détermine par l'analyse de ce spectre le meilleur filtre passe-bas qui nous permettra d'obtenir une composante lissée du jitter et donc par conséquent on réalisera une contribution lissée du jitter au signal prédit (Section 4.2.5, page 68).

3.3 Conclusion

Les signaux prédits à l'aide des prédicteurs issus du modèle de l'algorithme personnalisé sont calculés à partir des dérivées successives du signal que l'on désire prédire. Or, nous avons pu constater dans ce chapitre que ces dérivées évaluées par différences finies sont entachées d'un bruit de quantification qui s'amplifie avec le degré de la dérivation. De plus ces signaux dérivés sont multipliés par un coefficient directement dépendant du délai (Section 3.1.3, page 39). Par conséquent le bruit présent dans ces signaux dérivés sera également multiplié par les mêmes coefficients, il en résulte que les signaux prédits à partir des prédicteurs dérivés du modèle de l'algorithme personnalisé seront très bruités. Ils seront donc inexploitable pour calculer la force de retour lors d'un toucher distant. Il est alors impératif de filtrer les dérivées successives du signal que l'on désire prédire.

Chapitre 4

Filtrage du bruit de quantification

Introduction. Comme nous l'avons vu dans le chapitre précédent, les dérivées successives du signal d'entrée sur lesquelles s'appuient nos prédicteurs sont des signaux très bruités (Figure 3.6, page 44 et Figure 3.9, page 47 et Figure 3.12, page 50). Ce bruit est à l'origine un bruit (haute fréquence) de quantification du système à retour d'effort, il est amplifié par les dérivations successives effectuées par différences finies.

Nous sommes confrontés à plusieurs problèmes. Il nous faut tout d'abord décider de la méthode d'estimation du spectre [JEN 68] d'un signal aléatoire entaché d'un bruit de quantification [GUI 77]. De plus il nous faut faire un choix parmi les différents types de filtres existant dans la littérature. Ce choix effectué, il nous reste encore à déterminer le gabarit du filtre afin de respecter certaines contraintes. La première contrainte est celle du traitement des données en temps réel, on cherchera à obtenir un déphasage minimal résultant du temps de passage du signal à traiter dans le filtre. Les autres contraintes sont directement liées au gabarit du filtre tels que la fréquence de coupure, l'amplitude des oscillations en

bande passante, le coefficient d'atténuation dans la bande de coupure et la largeur de la bande de transition.

Nous présentons dans un premier temps, une étude du bruit de quantification ainsi que le choix d'un estimateur spectral pertinent de ce bruit [PRI 81]. Puis nous nous attachons à décrire le choix d'un type de filtrage passe bas, sur un critère de déphasage minimal. Enfin nous présentons les caractéristiques du filtre permettant d'affaiblir au maximum l'amplitude du bruit présent dans la dérivée première du signal à prédire, ensuite nous étudions le filtre passe bas permettant de filtrer le bruit présent dans la dérivée seconde du signal, enfin nous présentons le filtre permettant d'éliminer le bruit présent dans la dérivée tierce du signal.

4.1 Le bruit de quantification

4.1.1 Evaluation du spectre du bruit

L'estimateur spectral simple

Le bruit de quantification résulte de la numérisation d'un signal analogique, c'est un processus que l'on qualifie, en traitement du signal, d'aléatoire. La densité spectrale de puissance d'un processus aléatoire stationnaire au second ordre est la transformée de Fourier de sa fonction d'autocorrélation [Van 68].

La fonction d'autocorrélation est donnée par la relation suivante:

$$R_{xx}(\tau) = E\{x(t) \times x^*(t - \tau)\} \quad (\text{EQ 1})$$

on en tire donc la densité spectrale de puissance suivante:

$$G_{xx}(v) = \int_{-\infty}^{+\infty} R_{xx}(\tau) \times e^{-2\pi jv\tau} \times d\tau \quad (\text{EQ 2})$$

Nous obtenons donc pour les signaux discrets la relation suivante:

$$S_x(f) = \frac{1}{K} \times |X_K(f)|^2 \quad (\text{EQ 3})$$

avec
$$X_K(f) = \sum_{k=0}^{K-1} x(k) \times e^{-2\pi jfk}$$

La forme de l'estimateur spectral simple est donc représentée par (EQ 3). On l'appelle également dans une terminologie ancienne le périodogramme. Lors du calcul de l'espérance mathématique de l'estimateur spectral simple (ESS), on s'aperçoit que cet estimateur est biaisé:

$$E(S_x(f)) = \sum_{k=-(K-1)}^{K-1} \left(1 - \frac{|k|}{K}\right) \times R_{xx}(k) \times e^{-2\pi jfk} \quad (\text{EQ 4})$$

Par un argument de convergence dominée [DEL 93], on déduit que, si $R_{xx}(k)$ est de module sommable, le second membre tend vers $S_x(f)$ quand K tend vers

l'infini. Par conséquent, le périodogramme est un estimateur asymptotiquement sans biais de la densité spectrale de puissance : pour K suffisamment grand, $S_x(f)$ fluctue autour de la véritable valeur de la densité spectrale de puissance. Par contre, l'amplitude des fluctuations, qui est donnée par la variance de $S_x(f)$, ne tend pas vers 0 quand K tend vers l'infini. Plus précisément, on montre que cette variance est de l'ordre de la grandeur de la valeur à estimer. On conclut donc que l'estimateur spectral simple n'est pas un estimateur consistant du spectre : sa variance ne tend pas vers 0 quand la taille de l'échantillon tend vers l'infini.

L'estimateur spectral moyenné

La méthode la plus directe pour réduire la variance d'un estimateur est de calculer une moyenne sur plusieurs estimateurs indépendants [ROS 85]. Pour l'appliquer à l'estimateur spectral simple, il faut diviser le signal observé sur une durée K en L sections $x_l(k)$ de durée M chacune. On a :

$$x_l(k) = x(k + (l-1)M) \quad (\text{EQ 5})$$

avec $k = 0, \dots, (M-1)$, $K = M \times L$ et $l = 1, \dots, L$

On évalue ensuite L estimateurs simples du type :

$$S_{x_l}(f) = \frac{1}{M} \times \left| \sum_{k=0}^{M-1} x_l(k) \times e^{-2\pi j k k} \right|^2 \quad (\text{EQ 6})$$

avec $l = 1, \dots, L$

L'estimateur spectral moyenné est alors donné par :

$$\bar{S}_x(f) = \frac{1}{L} \times \sum_{l=1}^L S_{x_l}(f) \quad (\text{EQ 7})$$

Le biais de l'estimateur spectral moyenné est plus grand que le biais de l'estimateur spectral simple.

Cela aboutit à un algorithme qui fut proposé par P. Welch [WEL 67] :

1. On fragmente l'échantillon $\{x(k)\}$ de longueur K en L sous-intervalles de longueur M , avec ou sans chevauchement.

2. On applique éventuellement une fenêtre de pondération sur chacun des L sous-intervalles. On obtient la suite $\{\tilde{x}(k)\}$.

3. On calcule les L périodogrammes.

4. On effectue alors la moyenne suivante :

$$\widehat{S}_x(f) = \frac{1}{L} \times \sum_{l=0}^{L-1} \left(\frac{1}{M} \times \left| \sum_{k=0}^{M-1} \tilde{x}(l \times M + k) \times e^{-2\pi j f k} \right|^2 \right) \quad (\text{EQ 8})$$

Cette procédure appelle un certain nombre de remarques:

1. On montre que, sous des conditions assez générales, $\widehat{S}_x(f)$ est un estimateur du spectre dont la variance tend vers 0 quand K tend vers l'infini. En particulier, une condition est que L tende vers l'infini, mais moins vite que K.

2. En pratique, on réduit la variance en prenant L suffisamment grand. D'un autre côté, à K fixé, le fait d'augmenter le nombre L de périodogrammes revient à réduire la taille M de chaque fenêtre d'analyse et, par la même occasion, de réduire la résolution en fréquence. On sait en effet que, lorsqu'on observe un signal sur une durée M, on ne peut pas "voir" des écarts en fréquence inférieurs à $1/M$.

M est donc choisi de manière à réaliser un compromis entre résolution et variance.

3. L'absence de pondération dans l'expression (EQ 8) revient à multiplier les échantillons $x(k)$ du signal par la fenêtre rectangulaire:

$$w_r(k) = \frac{1}{M} \times 1_{\{0, \dots, (M-1)\}}(k), \text{ d'énergie } \sum_k w_r^2(k) = 1.$$

Cela fait apparaître des ondulations parasites liées aux lobes secondaires de sa transformée de Fourier discrète, d'où l'idée d'utiliser une fenêtre différente pour réduire ces ondulations. La conséquence est bien évidemment une réduction de la résolution liée à la largeur du lobe principal du spectre de la fonction fenêtre. De façon générale, on retrouve les résultats lors de la synthèse de filtres:

- plus la largeur du lobe principal est grande, plus les détails du spectre sont gommés, nous avons donc une perte de résolution en fréquence.

- plus la hauteur des lobes secondaires est importante, plus les ondulations provoquées sont grandes. Cela est particulièrement sensible dans les zones où le spectre présente peu de variations.

Il existe plusieurs types de fonctions fenêtre, rectangulaire, Bartlett, Hanning, Hamming, et Blackman en sont des exemples. Une fenêtre souvent utilisée est la fenêtre de Hamming :

$$w_h(k) = \alpha \times (0,54 - 0,46 \times \cos((2\pi k)/M)) \times 1_{\{0, \dots, (M-1)\}}(k)$$

où α est choisi de façon à ce que l'énergie de $w_h(k)$ soit égale à l'unité.

4. Le fait de multiplier certains échantillons par des coefficients de pondération très petits (environ 0.08 pour les valeurs extrêmes de chaque sous-intervalle), fait que ces échantillons jouent un rôle négligeable dans le calcul. D'où l'idée de Welch de faire chevaucher les sous-intervalles. Le taux de recouvrement le plus couramment utilisé est de 50%. Voici un exemple pour mieux visualiser le principe exposé ci-dessus :

On prend un signal de K=1000 échantillons, on fixe M=200 longueur de chaque sous-intervalles, on prend un taux de recouvrement de 50%, ce qui nous conduit à un nombre L=9 sous-intervalles distribués de la façon suivante:

x_1, \dots, x_{200} x_{201}, \dots, x_{400} x_{401}, \dots, x_{600} x_{601}, \dots, x_{800} x_{801}, \dots, x_{1000} x_{101}, \dots, x_{300} x_{301}, \dots, x_{500} x_{501}, \dots, x_{700} x_{701}, \dots, x_{900}

On pondère ensuite chacun de ces sous-intervalles par la fenêtre de Hamming adéquate. Puis on calcule les 9 périodogrammes et on en fait la moyenne.

5. Le fait de moyennner plusieurs périodogrammes ne fait pas disparaître complètement les "fluctuations". Plus précisément la référence [KAY 88] donne un intervalle de confiance à $\beta\%$ faisant intervenir la loi dite du χ^2 . En approximant celle-ci par une loi gaussienne, on en déduit que le spectre possède $\beta\%$ de chance de se trouver dans l'intervalle:

$$\left[\frac{\widehat{S}_w(f)}{1 + \gamma} \quad \frac{\widehat{S}_w(f)}{1 - \gamma} \right] \text{ avec } \gamma = \frac{\sqrt{2} \times \text{erfinv}(\beta)}{\sqrt{L}}$$

où $\widehat{S}_w(f)$ désigne le périodogramme moyenné avec la méthode de Welch, utilisant un recouvrement de 50%. La fonction $\text{erfinv}(\beta)$ est la fonction inverse de la fonction erreur.

Présentation des différentes fonctions fenêtre

1. La fenêtre rectangulaire

La fonction fenêtre est donnée par la relation suivante:

$$w_n = 1 \text{ avec } 0 < n < N - 1$$

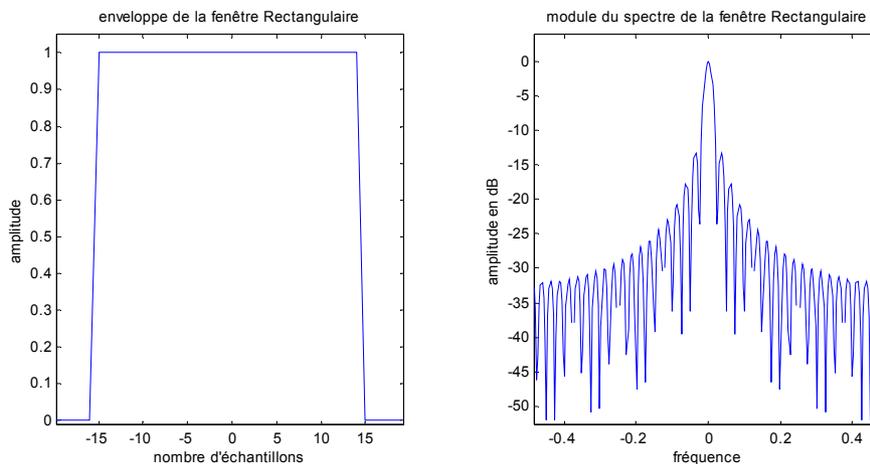


figure 4.1 : enveloppe et spectre de la fenêtre Rectangulaire.

On peut observer que le lobe central du spectre est assez étroit ceci qui se traduit sur le spectre estimé par une très bonne résolution en fréquence, par contre les lobes secondaires ont une amplitude élevée (environ -15dB pour les premiers lobes secondaires). ceci se traduira dans le spectre estimé par l'apparition d'oscillations dont l'amplitude sera assez importante (environ un sixième de l'amplitude du spectre) car elle est directement liée à l'amplitude des lobes secondaires.

2. La fenêtre de Bartlett

La fonction fenêtre [BAR 46] est donnée par la relation suivante:

$$w_n = \frac{n}{N-1} \text{ avec } 0 < n < \frac{N-2}{2}$$

$$\text{et } w_n = 2 - \frac{2 \times n}{N-1} \text{ avec } \frac{N-1}{2} < n < N-1$$

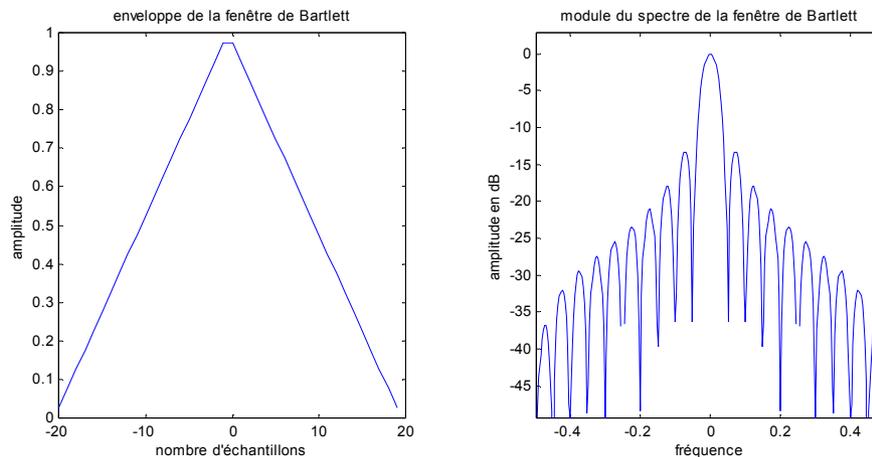


figure 4.2 : enveloppe et spectre de la fenêtre de Bartlett.

On observe un élargissement du lobe principal du spectre, on perd donc en terme de résolution en fréquence, par contre l'amplitude des lobes secondaires (environ -15 dB pour les deux premiers lobes secondaires) est moins importante. On aura alors diminué l'amplitude des oscillations apparues dans le spectre.

3. La fenêtre de Hanning

La fonction fenêtre est donnée par la relation suivante:

$$w_n = \frac{1}{2} \times \left[1 - \cos\left(\frac{2\pi n}{N-1}\right) \right] \text{ avec } 0 < n < N-1$$

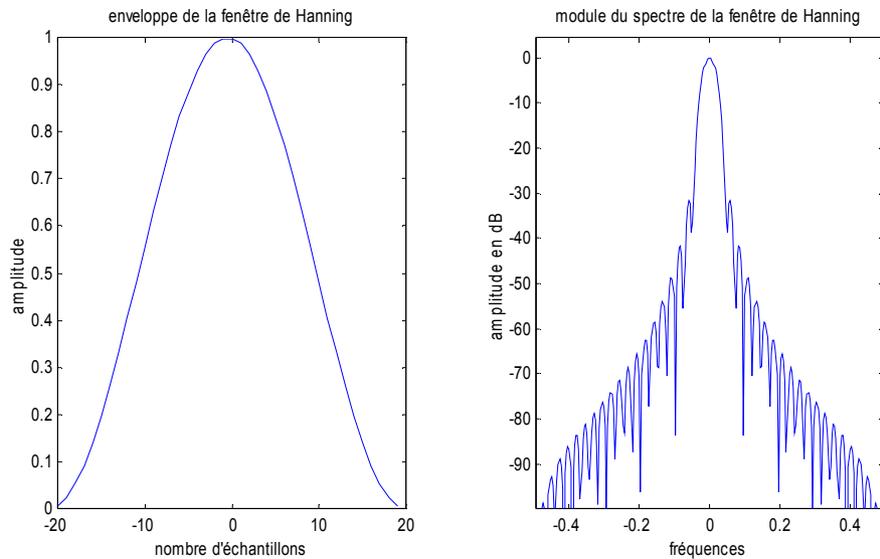


figure 4.3 : enveloppe et spectre de la fenêtre de Hanning.

On remarque sur le spectre de la fenêtre de hamming que la largeur du lobe central est la même que la largeur du lobe central de la fenêtre de Bartlett, elles auront donc une résolution en fréquence très proche. De plus, on observe également une diminution de l'amplitude des lobes secondaires (même si les premiers lobes secondaires ont une amplitude d'environ -30 dB). Il en résulte une atténuation plus importante des oscillations apparues dans le spectre.

4. La fenêtre de Hamming

La fonction fenêtre [CHEN 79] est donnée par la relation suivante:

$$w_n = 0,54 - 0,46 \times \cos\left(\frac{2\pi n}{N-1}\right) \text{ avec } 0 < n < N-1$$

c'est une fonction dite de cosinus surélevé.

C'est tout d'abord une fenêtre très souvent utilisée car la largeur du lobe central de son spectre reste inchangée par rapport aux fenêtres de Hanning et Bartlett. Par contre ses lobes secondaires ont une amplitude de l'ordre de -50 dB, ce qui entraîne la quasi-disparition des oscillations parasites dues à ces lobes.

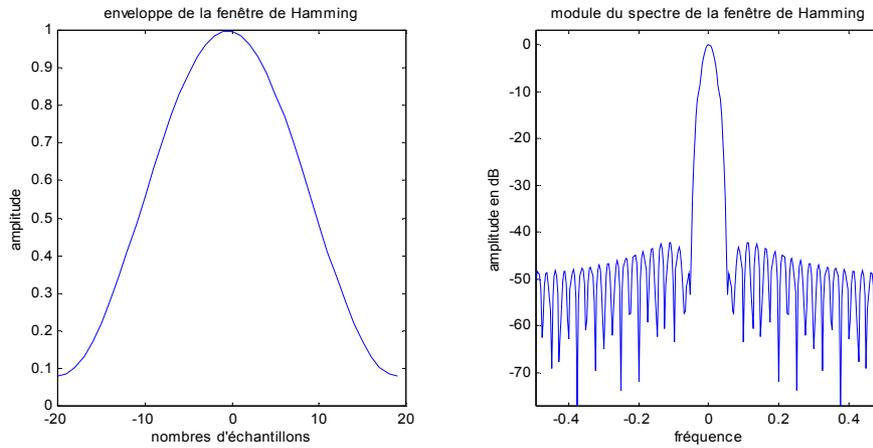


figure 4.4 : enveloppe et spectre de la fenêtre de Hamming.

5. La fenêtre de Blackman

La fonction fenêtre [BLA 58] est donnée par la relation suivante:

$$w_n = 0,42 - 0,5 \times \cos\left(\frac{2\pi n}{N-1}\right) + 0,08 \times \cos\left(\frac{4\pi n}{N-1}\right) \text{ avec } 0 < n < N-1$$

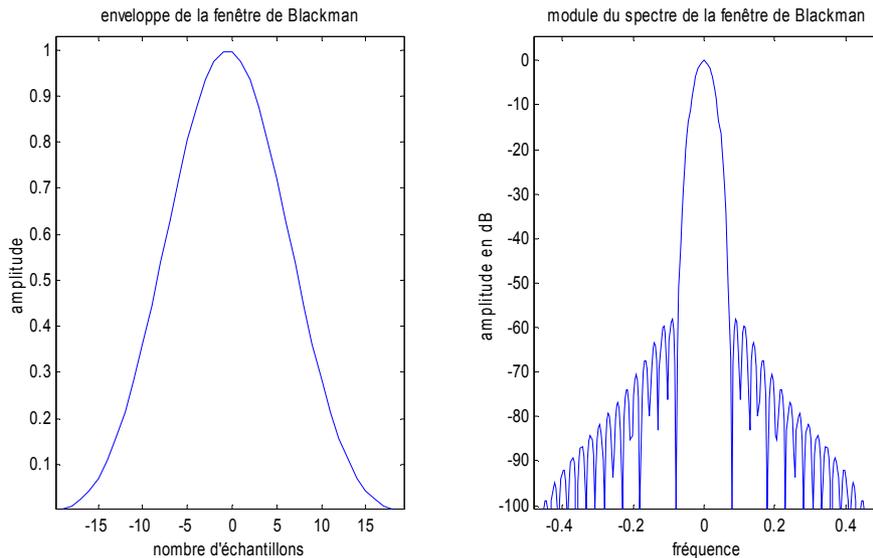


figure 4.5 : enveloppe et spectre de la fenêtre de Blackman.

On peut observer sur le module du spectre que les lobes secondaires ont une amplitude inférieure à -60 dB, c'est à dire que les oscillations parasites dues à ces lobes auront une amplitude mille fois plus petite que l'amplitude du signal traité à l'instant n. Autrement dit, elles seront pratiquement indétectables dans le périodogramme moyenné, par contre la largeur du lobe central a encore augmenté par rapport à la fenêtre de Hamming, il en résulte une perte supplémentaire en terme de résolution en fréquence.

Pour conclure, la fenêtre de Hamming est dans le cas général un bon compromis. Sa résolution en fréquence est environ deux fois moins bonne que celle de la fenêtre rectangulaire, mais l'amplitude des oscillations parasites dues à la présence des lobes secondaires est atténuée d'un facteur 30. Il existe vraiment une multitude de fenêtres, faisant partie des trois sous-groupes suivants : les fenêtres paraboliques, triangulaires et cosinusoïdales. Les fenêtres de Kaiser [KAI 66] et de Chebychev sont des fenêtres paramétrables, c'est à dire que l'on peut spécifier dans le domaine des fréquences le compromis entre la largeur du lobe central et l'amplitude des lobes secondaires.

4.1.2 Les résultats de l'estimation spectrale

Nous nous sommes contentés d'utiliser uniquement pour l'estimation de la densité spectrale de puissance (DSP) deux types de fenêtres : la fenêtre rectangulaire et la fenêtre de Hamming.

Spectre de la dérivée première du signal (vitesse)

On s'aperçoit que la dérivée première du signal d'entrée (composante x d'un signal gestuel) possède un spectre dont l'amplitude basse fréquence est bien supérieure à l'amplitude des hautes fréquences et ceci jusqu'à environ 25 Hz. Il sera donc assez aisé d'isoler cette partie du signal, par un filtrage passe bas.

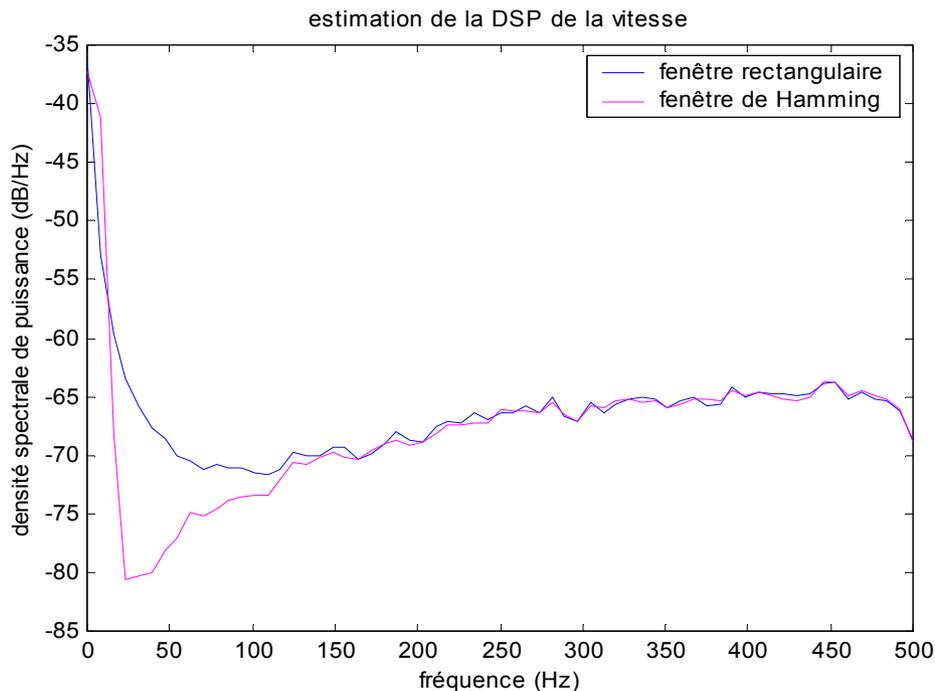


figure 4.6 : densité spectrale de puissance de la vitesse.

Spectre de la dérivée seconde du signal (accélération)

Pour la dérivée seconde du signal d'entrée, la partie basse fréquence du spectre est largement inférieure en amplitude à la partie haute fréquence (Figure 4.6, page 59), or comme nous l'avons répété jusqu'ici, c'est la partie basse

fréquence de l'accélération qui nous intéresse. On note également que la partie haute fréquence du spectre possède une amplitude bien plus importante que la partie haute fréquence du spectre estimé de la vitesse. Ceci provient directement de l'allure du gain du filtre permettant de calculer la vitesse (Figure 3.5, page 44 et Figure 3.8, page 47). Pour les hautes fréquences, ce gain en amplitude du spectre atteint un maximum de 2 pour la vitesse, alors que pour l'accélération ce gain atteint une valeur maximale de 4.

Inversement, l'atténuation d'amplitude pour les basses fréquences ($f < 166$ Hz) est plus importante que pour le filtre permettant de calculer la dérivée première du signal. Pour ces raisons, il sera alors plus difficile d'isoler la partie basse fréquence de l'accélération que d'isoler la partie basse fréquence de la vitesse.

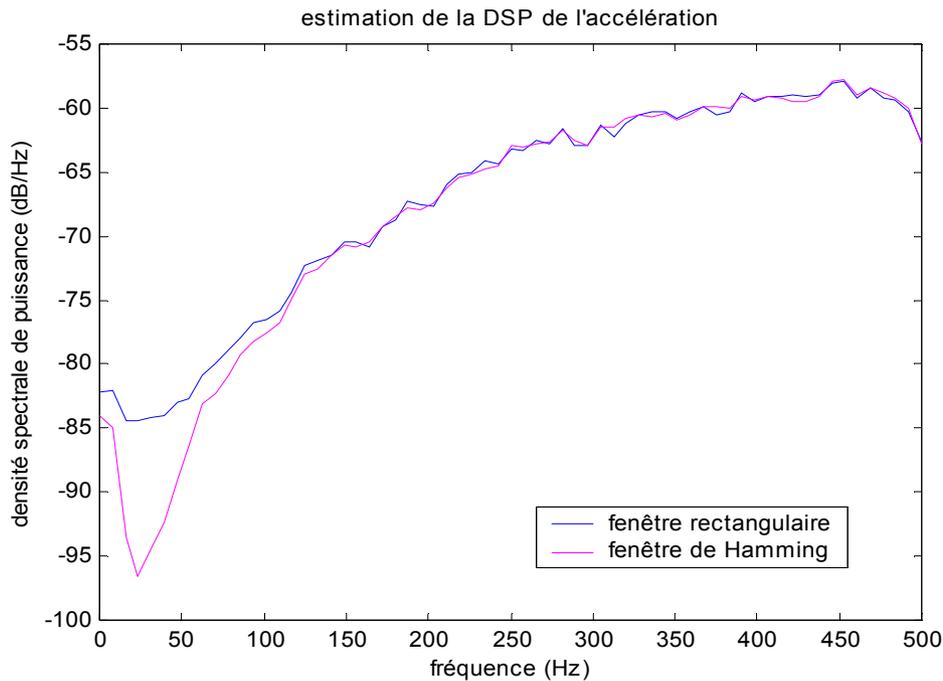


figure 4.7 : densité spectrale de puissance de l'accélération.

Spectre de la dérivée tierce du signal (jitter)

Pour la dérivée tierce du signal d'entrée, la partie basse fréquence du spectre est largement inférieure en amplitude à la partie haute fréquence (Figure 4.7, page 60), or comme nous l'avons répété jusqu'ici, c'est la partie basse fréquence des dérivées successives du signal qui nous intéresse. On note également que la partie haute fréquence du spectre possède une amplitude bien plus importante que la partie haute fréquence du spectre estimé du jitter. Ceci provient directement de l'allure du gain du filtre permettant de calculer le jitter. Pour les hautes fréquences le gain en amplitude du spectre atteint un maximum de 2 pour la vitesse, alors que pour le jitter, ce gain atteint une valeur maximale de 8.

Inversement, l'atténuation d'amplitude pour les basses fréquences ($f < 166$ Hz) du filtre permettant de calculer le jitter, est bien plus importante que pour le filtre permettant de calculer la dérivée première du signal, notamment pour les très

basses fréquences. Le gain du filtre permettant de calculer la dérivée tierce est pratiquement nul pour les fréquences inférieures à 50 Hz, il en résulte donc une très forte atténuation du spectre pour les très basses fréquences (Figure 3.5, page 44 et Figure 3.11, page 49). Pour ces raisons, il sera alors beaucoup plus difficile d'isoler la partie basse fréquence du jitter que d'isoler la partie basse fréquence de la vitesse.

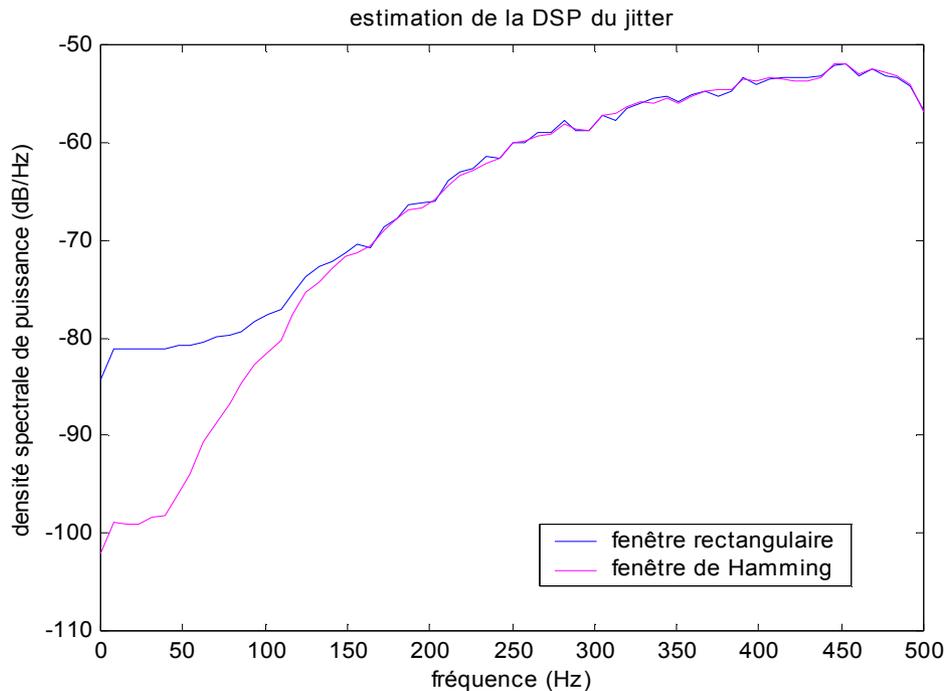


figure 4.8 : densité spectrale de puissance du jitter.

4.2 Le filtrage du bruit

4.2.1 Généralités

Le système linéaire numérique

Les systèmes linéaires discrets invariants dans le temps (LIT) constituent un domaine très important du traitement numérique du signal. Il s'agit des filtres numériques à caractéristiques fixes. Ces systèmes se caractérisent par le fait que leur fonctionnement est régi par une équation de convolution. L'analyse de leurs propriétés se fait à l'aide de la Transformation en Z [CAR 61], qui joue pour les

systemes discrets le même rôle que la transformée de Laplace ou de Fourier [PAL 34] pour les systemes continus.

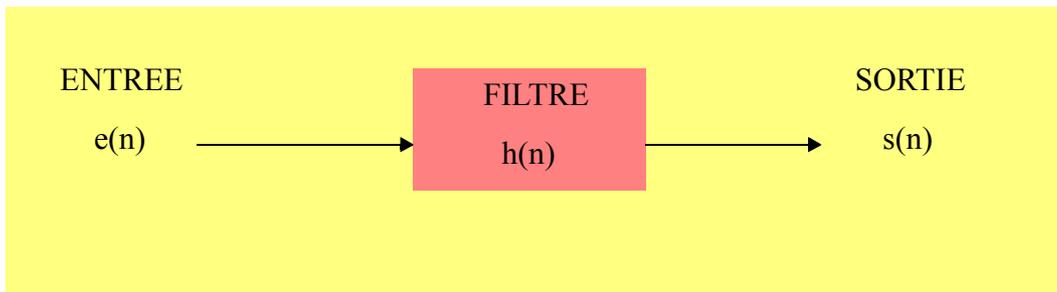


figure 4.9 : synopsis d'un système linéaire.

L'équation de convolution qui caractérise un système linéaire invariant dans le temps est formalisée par la relation suivante :

$$s(n) = \sum_{m=0}^{\infty} h(m) \times e(n - m) \quad (\text{EQ 9})$$

Un tel système est donc complètement défini par la donnée de la suite $h(n)$, qui est appelée réponse impulsionnelle du système [KAT 66].

Le filtre passe bas

Le filtre passe bas est un système linéaire, il a pour fonction de laisser passer les basses fréquences contenues dans le signal (son allure générale) et de supprimer ou d'atténuer les hautes fréquences du spectre de ce signal [KUN 86]. On le définit par son gabarit qui est fonction de plusieurs paramètres telles que l'ondulation en bande passante, la largeur de la bande passante, la fréquence de coupure, la largeur de la bande de transition ainsi que l'atténuation dans la bande de coupure (Figure 4.10, page 63).

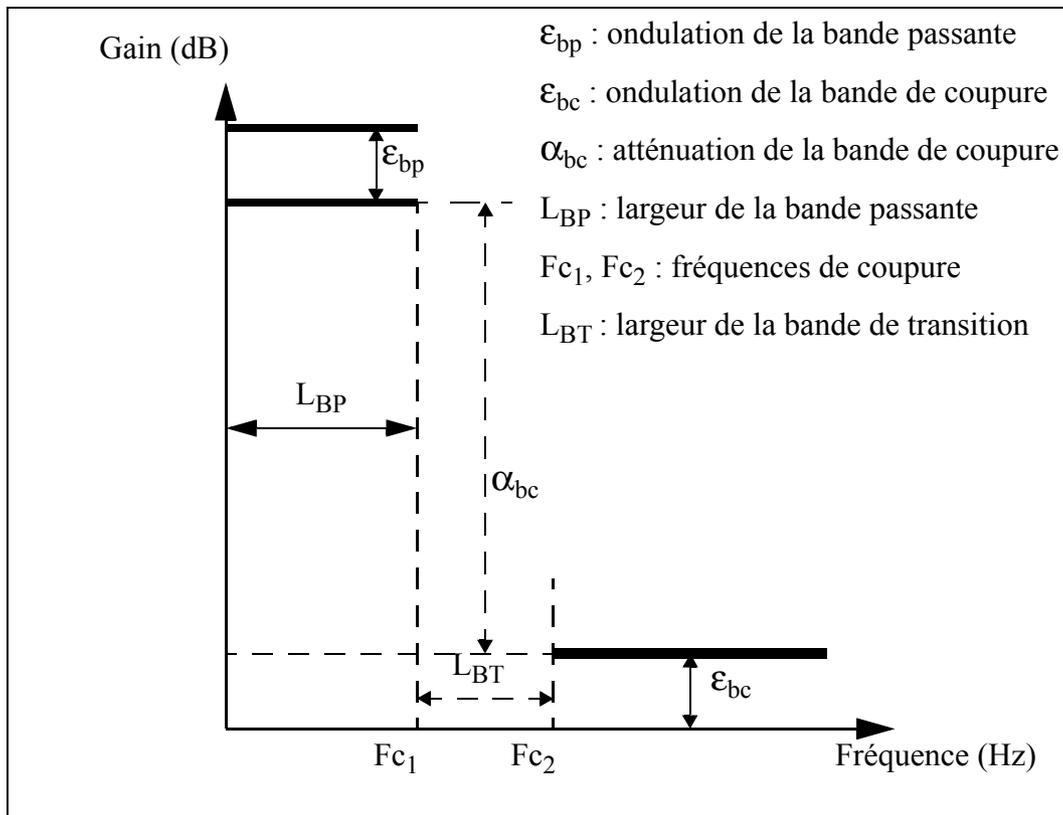


figure 4.10 : Gabarit d'un filtre passe bas.

4.2.2 Le choix du type de filtre

Le filtre passe bas à réponse impulsionnelle finie (RIF)

Les filtres numériques à réponse impulsionnelle finie sont des systèmes linéaires discrets invariants dans le temps, définis par une équation selon laquelle un nombre de sortie, représentant un échantillon filtré, est obtenu par une sommation pondérée d'un ensemble fini de nombres d'entrée, représentant les échantillons du signal à filtrer [OPP 89]. Les coefficients de la sommation pondérée constituent la réponse impulsionnelle du filtre et un ensemble fini d'entre eux seulement prennent des valeurs non nulles. Ce filtre est du type "à mémoire finie", c'est à dire qu'il détermine sa sortie en fonction d'informations d'entrée d'ancienneté limitée. Il est fréquemment désigné par un filtre non récursif, en raison de sa structure, car il ne nécessite pas de boucle de réaction dans sa réalisation, comme c'est notamment le cas pour une autre catégorie de filtres, celle des filtres à réponse impulsionnelle infinie.

La relation entrée-sortie de ces filtres peut être exprimée par l'expression suivante :

$$s(n) = b(1) + b(2) \times e(n-1) + \dots + b(i) \times e(n-i+1) \quad (\text{EQ 10})$$

L'entier i représente l'ordre de ce filtre, c'est aussi le nombre de coefficients du filtre.

Pour un filtre RIF passe bas, le gabarit est donné par l'ondulation en bande passante et en bande affaiblie ϵ_{bp} et ϵ_{bc} , la fréquence marquant la fin de la bande passante F_{c1} , et par la largeur de la bande de transition L_{BT} . En analysant les résultats du calcul d'un grand nombre de filtres avec des spécifications très variées, on constate qu'en première approximation le nombre de coefficients est proportionnel au logarithme de $\frac{1}{\epsilon_{bc}}$ et de $\frac{1}{\epsilon_{bp}}$ ainsi qu'au rapport de la fréquence d'échantillonnage F_e à la bande de transition L_{BT} (Figure 4.10, page 63).

Par ajustage des paramètres, on obtient alors l'estimation N_{coeff} suivante pour le nombre de coefficients d'un filtre à réponse impulsionnelle finie :

$$N_{coeff} = \frac{2}{3} \times \log \left(\frac{1}{10 \times \frac{\epsilon_{bp}}{2} \times \epsilon_{bc}} \right) \times \frac{F_e}{L_{BT}} \quad (EQ 11)$$

avec $L_{BT} = F_{c2} - F_{c1}$

Le filtre passe bas à réponse impulsionnelle infinie (RII)

Les filtres numériques à réponse impulsionnelle infinie sont des systèmes linéaires discrets invariants dans le temps dont le fonctionnement est régi par une équation de convolution portant sur une infinité de termes. En principe, ils conservent une trace des signaux qui leur ont été appliqués pendant une durée infinie, ils sont à mémoire infinie. Une telle mémoire est réalisée par une boucle de réaction de la sortie sur l'entrée, d'où la dénomination courante de filtre récursif. Chaque élément de la suite des nombres de sortie est calculé par la sommation pondérée d'un certain nombre d'éléments de la suite d'entrée et d'un certain nombre d'éléments de la suite de sortie [BLA 98].

Le fait d'avoir cette réponse impulsionnelle infinie permet d'obtenir en général des fonctions de filtrages beaucoup plus sélectives que celles des filtres RIF à quantités équivalentes de calculs .

La relation entrée-sortie de ces filtres peut être exprimée par l'expression suivante :

$$\begin{aligned} s(n) = & b(1) + b(2) \times e(n-1) + \dots + b(j) \times e(n-j+1) \\ & -(a(1) + a(2) \times s(n-1) + \dots + a(j) \times s(n-j+1)) \end{aligned} \quad (EQ 12)$$

L'entier j représente l'ordre du filtre.

Le filtre passe bas (RII) elliptique

Le filtre elliptique présente des ondulations en bande passante et en bande affaiblie. Il est optimal en ce sens que pour un ordre n donné et des amplitudes d'ondulations fixées, il présente la bande de transition la plus faible.

Une estimation du nombre de coefficients pour une bande transition fine, et une largeur de bande passante étroite est donnée par la relation suivante :

$$N_{\text{coeff}} \cong 1,076 \times \log \left(\frac{2}{\sqrt{\frac{\epsilon_{\text{bp}}}{2}} \times \epsilon_{\text{bc}}} \right) \times \log \left(\frac{F_{c1}}{L_{\text{BT}}} \right) \quad (\text{EQ 13})$$

avec $L_{\text{BT}} = F_{c2} - F_{c1}$

Ainsi pour les filtres à réponse impulsionnelle infinie de type elliptique, l'ordre est proportionnel au logarithme de l'inverse de la bande de transition normalisée, ce qui conduit à des valeurs beaucoup plus faibles que dans le cas des filtres à réponse impulsionnelle finie [BEL 90]. On obtient alors un temps de passage dans le filtre beaucoup plus faible. De plus, le fait d'avoir moins de coefficients permet de diminuer grandement le temps de calcul de la filtrée, ce qui nous permet de satisfaire à la contrainte de calcul en temps réel (tous les calculs doivent être effectués en moins de 1ms, ce qui correspond à la période de la boucle haptique).

4.2.3 Le filtrage du bruit contenu dans la vitesse

Nous avons utilisé un filtre elliptique [PRA 89]. Le gabarit que nous proposons d'utiliser possède les caractéristiques suivantes :

ordre du filtre	$n = 6$
ondulation dans la bande passante	$\epsilon_{\text{bp}} = 0.02 \text{ dB}$
facteur d'atténuation	$\alpha_{\text{bc}} = 30 \text{ dB}$
fréquence de coupure	$F_{c1} = 41 \text{ Hz}$
largeur de la bande de transition	$L_{\text{BT}} = 6 \text{ Hz}$

Le tableau des coefficients correspondants aux caractéristiques ci-dessus est le suivant :

$a(1)=1.0$	$b(1)=0.029228833596732$
$a(2)=-5.340292817183203$	$b(2)=-0.150610504796137$
$a(3)=12.008014502129118$	$b(3)=0.343709765658651$
$a(4)=-14.539561938571605$	$b(4)=-0.444446944354783$
$a(5)=9.992409631383369$	$b(5)=0.343709765658651$
$a(6)=-3.694162877459526$	$b(6)=-0.150610504796137$
$a(7)=0.573803226624091$	$b(7)=0.029228833596732$

Le spectre de la vitesse filtrée représenté Figure 4.11, page 66, est confondu avec le spectre du signal représentant la vitesse jusqu'à une fréquence d'environ 50 Hz, au delà de ce point on note une forte atténuation du module du spectre de la filtrée, cette atténuation atteint les 30 dB fixés lors de la détermination du gabarit du filtre elliptique aux environs de 175 Hz, avec une atténuation de 20 dB atteint dès les 75 Hz.

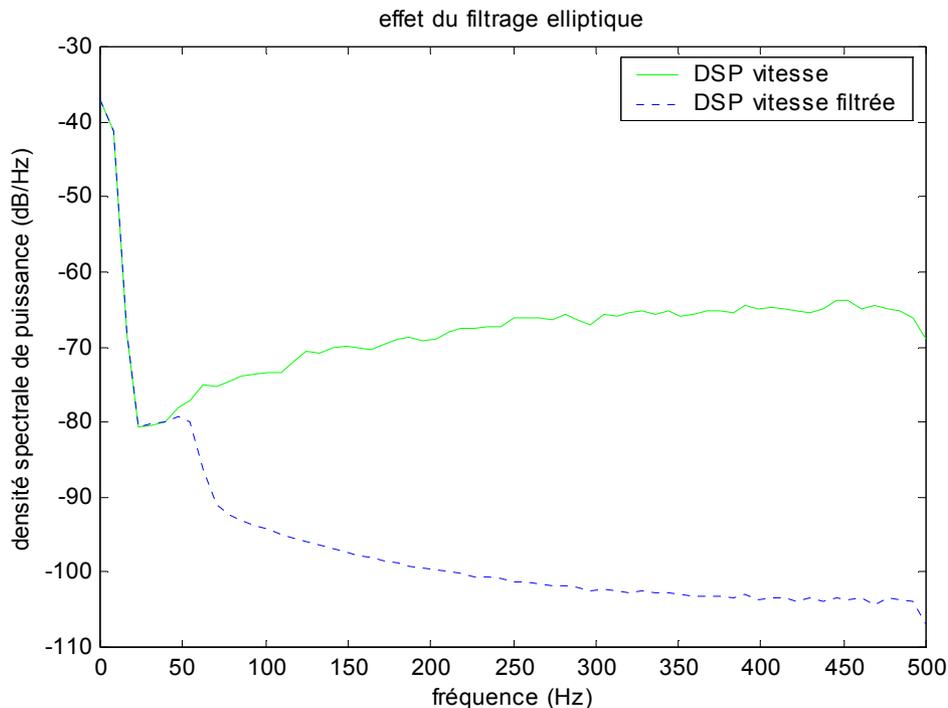


figure 4.11 : filtrage elliptique du spectre de la vitesse.

4.2.4 Le filtrage du bruit contenu dans l'accélération

Nous avons utilisé un filtre elliptique. Le gabarit que nous proposons d'utiliser possède les caractéristiques suivantes :

ordre du filtre	$n = 4$
ondulation dans la bande passante	$\epsilon_{bp} = 0.04 \text{ dB}$
facteur d'atténuation	$\alpha_{bc} = 80 \text{ dB}$
fréquence de coupure	$F_{c1} = 80 \text{ Hz}$
largeur de la bande de transition	$L_{BT} = 300 \text{ Hz}$

Le tableau des coefficients correspondants aux caractéristiques ci-dessus est le suivant :

a(1)=1.0	b(1)=0.000216083163017
a(2)=-3.711876275508533	b(2)=-0.000276041761253
a(3)=5.190010477018813	b(3)=0.000406477172726
a(4)=-3.238749720588917	b(4)=-0.000276041761253
a(5)=0.760903401755644	b(5)=0.000216083163017

Le spectre de l'accélération filtrée représenté Figure 4.12, page 67, est confondu avec le spectre du signal représentant l'accélération jusqu'à une fréquence d'environ 75 Hz, au delà de ce point on note une forte atténuation du module du spectre de la filtrée, cette atténuation atteint les 80 dB fixés lors de la détermination du gabarit du filtre elliptique aux environs de 350 Hz, avec une atténuation de 20 dB atteint dès les 100 Hz.

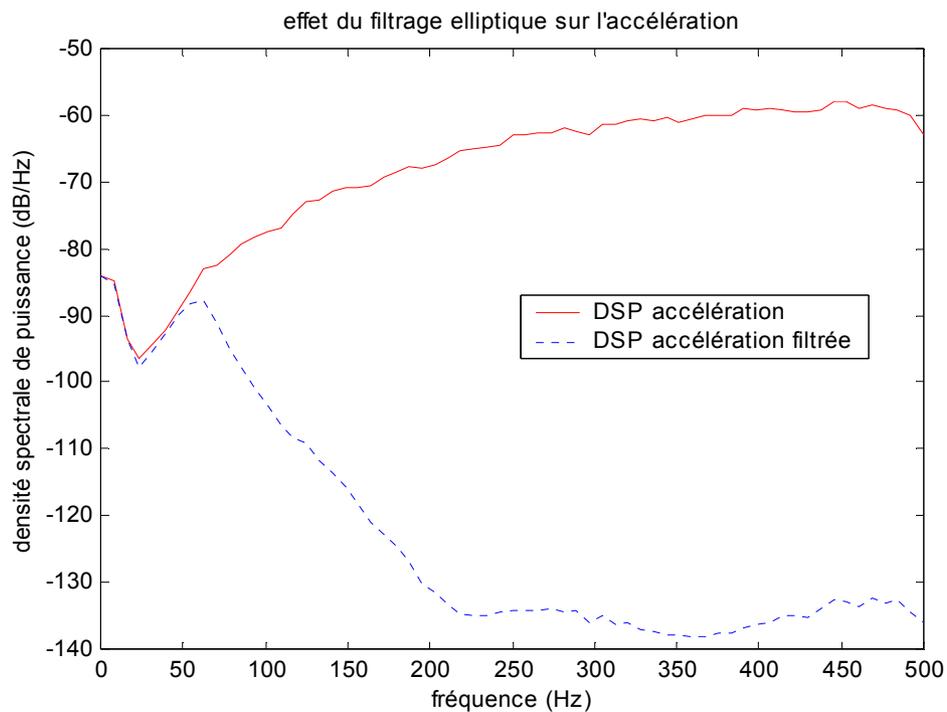


figure 4.12 : filtrage elliptique du spectre de l'accélération.

4.2.5 Le filtrage du bruit contenu dans le jitter

Nous avons utilisé un filtre elliptique. Le gabarit que nous proposons d'utiliser possède les caractéristiques suivantes :

ordre du filtre	$n = 3$
ondulation dans la bande passante	$\epsilon_{bp} = 0.2 \text{ dB}$
facteur d'atténuation	$\alpha_{bc} = 140 \text{ dB}$
fréquence de coupure	$F_{c1} = 70\text{Hz}$
largeur de la bande de transition	$L_{BT} = 380\text{Hz}$

Le tableau des coefficients correspondants aux caractéristiques ci-dessus est le suivant :

a(1)=1.0	b(1)=0.001289413955435
a(2)=-2.610253893050856	b(2)=0.003847346253766
a(3)=2.319842682216394	b(3)=0.003847346253766
a(4)=-0.699315268747137	b(4)=0.001289413955435

Contrairement au cas de la vitesse et de l'accélération, le spectre du jitter filtré représenté Figure 4.13, page 69, n'est pas confondu avec le spectre du signal représentant le jitter pour les basses fréquences. On note une atténuation des très basses fréquences de l'ordre de 15 dB. Par la suite, cette atténuation tend vers 0 dB sur une plage étroite de fréquences comprise entre 60 et 80 Hz, au delà de ce point on note une forte atténuation du module du spectre de la filtrée. Cette atténuation n'atteint jamais les 140 dB fixés lors de la détermination du gabarit du filtre elliptique, mais on détecte une atténuation de 20 dB aux alentours des 125 Hz.

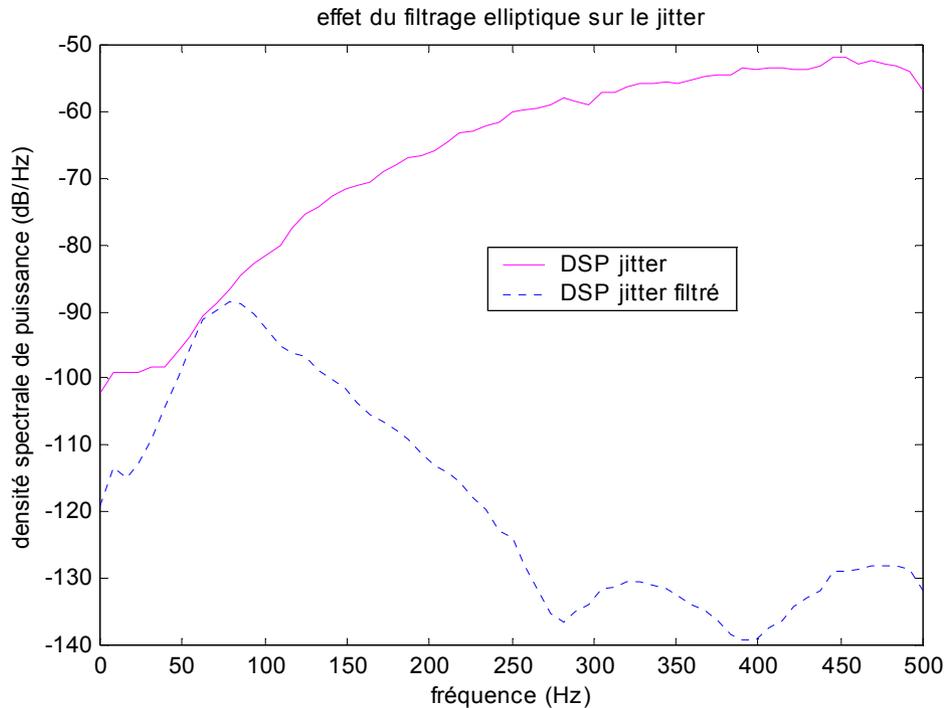


figure 4.13 : filtrage elliptique du spectre du jitter.

4.3 Conclusion

Afin de réduire l'effet perturbateur du bruit présent dans la vitesse, dans l'accélération et dans le jitter, nous avons choisi d'utiliser un filtre passe bas de réponse impulsionnelle infinie de type elliptique. Le nombre de coefficients lié au gabarit du filtre elliptique est bien inférieur au nombre de coefficients d'un filtre à réponse impulsionnelle finie qui aurait un gabarit identique. Il en résulte un déphasage bien moins important, ce qui conduit à un temps de passage du signal dans la structure beaucoup plus faible. On obtient des résultats de filtrage très intéressants pour la vitesse et pour l'accélération, par contre il sera nécessaire de pondérer la composante issue du filtrage de la dérivée tierce du signal que l'on nomme jitter. Ces filtres seront présents dans l'architecture finale des codeurs prédictifs (Section 8, page 117).

Chapitre 5

La prédiction : un problème de filtrage

Introduction. Les problèmes d'estimation et plus particulièrement les procédures de prédiction s'apparentent généralement à un problème de filtrage. En fait, il est intéressant de considérer les différents prédicteurs comme des filtres que l'on applique au signal à transmettre. Nous allons calculer pour chacun des prédicteurs le gain et la phase du filtre qui leur est associé, on pourra ainsi mieux appréhender les résultats de chacun des prédicteurs. De plus le gain et la phase de chacun des filtres associés aux prédicteurs ont une caractéristique commune car ils ne sont fonction que de la fréquence, du pas d'échantillonnage et du délai introduit par le réseau.

Nous définissons dans un premier temps les limites du cadre de notre étude, puis nous présentons les caractéristiques intrinsèques de gain et de phase des filtres (FP1, FP2, et FP3) associés aux prédicteurs d'ordre 1, 2 et 3. Enfin, nous présentons des résultats récapitulatifs, sur les filtres associés aux prédicteurs proposés.

5.1 Définition des limites du cadre de l'étude

Le pas d'échantillonnage sera en général constant puisqu'il est égal à l'inverse de la fréquence de la boucle haptique qui gère le système à retour d'effort Phantom. Dans notre étude, nous avons décidé d'examiner les performances des prédicteurs pour trois délais différents:

$$\delta_1 = 10 \text{ ms} \quad \delta_2 = 30 \text{ ms} \quad \delta_3 = 100 \text{ ms} .$$

Le signal d'entrée des prédicteurs correspond à la suite de positions (x,y,z) qui forme la trajectoire de la main pendant le geste. Une étude du geste nous a permis de constater que la fréquence maximale sur chaque composante (x,y,z) détectée dans un geste quelconque est toujours inférieure à 10 Hz, et dans le cas particulier du geste d'écriture cette valeur est ramenée à 3 Hz. On respecte donc largement la condition de Shannon [SHA 49] dans l'échantillonnage du signal d'entrée car la fréquence d'échantillonnage est fixée à 1000 Hz. La condition de Shannon permettant la reconstruction du signal analogique à partir du signal numérisé s'exprime par la relation:

$$F_{\max} \leq \frac{F_e}{2} \quad (\text{EQ 1})$$

Dans notre cas, la fréquence maximale admise (i.e respectant Shannon) contenue dans l'une des trois composantes du signal d'entrée est alors de 500 Hz [NYQ 28]. Avec des fréquences maximales, enregistrées sur chaque composante pour un geste quelconque, inférieures à 10 Hz, on est donc bien loin des limites de Shannon ($F_{\text{shannon max}} \leq 500 \text{ Hz}$). On se limitera dans notre étude à deux cas : $F_{\max} \leq 3 \text{ Hz}$ $F_{\max} \leq 10 \text{ Hz}$. Ces limites étant fixées par le système à retour d'effort mais aussi par les limites humaines dans la rapidité de changement de direction de la main [SRI 97].

5.2 Le prédicteur d'ordre 0

Pour la prédiction d'ordre 0, on utilise directement les données issues du réseau, et dans le cas d'un réseau idéal qui ne ferait que retarder le signal d'entrée, on obtient en sortie le signal suivant:

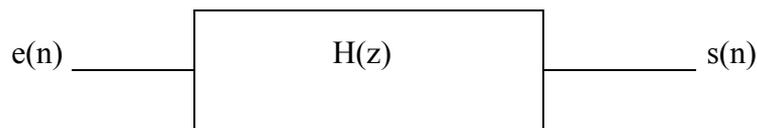


figure 5.1 : synopsis du filtrage.

ceci se traduit donc par l'équation suivante:

$$s(n) = e(n - \delta) \quad (\text{EQ 2})$$

on passe en transformée en Z: $S(z) = E(z) \times z^{-\delta}$

on en tire alors la fonction de transfert en z du filtre:

$$H(z) = \frac{S(z)}{E(z)} = z^{-\delta}$$

ce qui traduit effectivement un retard, afin d'exploiter les résultats il nous faut passer par la transmittance isochrone en utilisant un changement de variables $z = e^{j\omega T_e}$ dans lequel T_e correspond au pas d'échantillonnage du signal d'entrée. La transmittance isochrone correspond à la fonction de transfert d'un filtre analogique qui aurait les mêmes propriétés que le filtre numérique caractérisé par $H(z)$. On obtient alors:

$$\begin{aligned} H(\omega) &= e^{-j\omega\delta T_e} = \cos(\omega\delta T_e) - j \times \sin(\omega\delta T_e) \\ |H(\omega)| &= 1 \\ \phi(\omega) &= -\omega\delta T_e \end{aligned} \quad (\text{EQ 3})$$

On arrive donc bien au résultat escompté, car le gain du filtre est unitaire et le déphasage introduit entre l'entrée et la sortie est une fonction de la fréquence ω , du délai δ et du pas d'échantillonnage T_e . On comprend également que le déphasage augmentera non seulement avec le délai mais également avec la fréquence du signal d'entrée. Le déphasage augmentera linéairement avec la fréquence pour un délai fixé. Ces données nous permettront d'évaluer par comparaison la performance des autres prédicteurs.

5.3 Le prédicteur d'ordre 1

Dans ce cas le signal prédit est une fonction du signal d'entrée retardé et de sa dérivée première. Le filtre associé à la prédiction d'ordre 1 sera dorénavant nommé FP1.

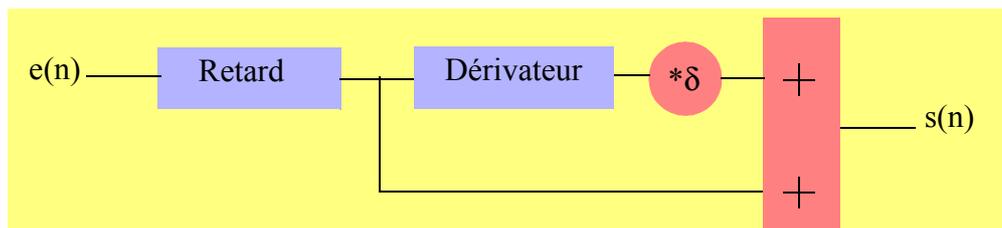


figure 5.2 : circuit permettant de réaliser la prédiction d'ordre 1.

5.3.1 Calcul du module et de la phase du filtre FP1

Le signal de sortie s'exprime par la relation suivante :

$$s(n) = e(n - \delta) + \delta \times \frac{d}{dn} e(n - \delta) \quad (\text{EQ 4})$$

on passe alors en transformée en z :

$$S(z) = E(z) \times z^{-\delta} + \delta \times z^{-\delta} \times (1 - z^{-1}) \times E(z)$$

on peut alors évaluer la fonction de transfert du filtre associé à la prédiction d'ordre 1:

$$H(z) = \frac{S(z)}{E(z)} = z^{-\delta} \times [1 + \delta \times (1 - z^{-1})] \quad (\text{EQ 5})$$

Afin de mieux appréhender les résultats, il nous faut calculer la transmittance isochrone, en effectuant le changement de variables suivant : $z = e^{jwT_e}$, l'équation (EQ 5) devient alors:

$$H(w) = e^{-jw\delta T_e} \times [1 + \delta \times (1 - e^{-jwT_e})] \quad (\text{EQ 6})$$

$$\text{on pose: } A = w\delta T_e \quad B = wT_e$$

on obtient alors:

$$\text{Re}[H(w)] = (\delta + 1) \times \cos A - \delta \times \cos(A + B)$$

$$\text{Im}[H(w)] = \delta \times \sin(A + B) - (\delta + 1) \times \sin A$$

On peut donc maintenant calculer le module et la phase du filtre correspondant à la prédiction d'ordre 1 :

$$\begin{aligned} |H(w)| &= \sqrt{\text{Re}^2[H(w)] + \text{Im}^2[H(w)]} \\ &= \sqrt{2\delta \times (1 - \cos(wT_e)) \times (\delta + 1) + 1} \end{aligned}$$

$$\begin{aligned} \phi(w) &= \text{atan}\left(\frac{\text{Im}[H(w)]}{\text{Re}[H(w)]}\right) \\ &= \text{atan}\left(\frac{\delta \times \sin(w\delta T_e + wT_e) - (\delta + 1) \times \sin(w\delta T_e)}{(\delta + 1) \times \cos(w\delta T_e) - \delta \times \cos(w\delta T_e + wT_e)}\right) \end{aligned}$$

Le module et la phase du filtre d'ordre 1 sont donc fonction de trois paramètres : la fréquence w , le délai δ introduit par le réseau et le pas d'échantillonnage T_e .

5.3.2 Représentation graphique du module et de la phase du filtre FP1

En respectant le cadre de notre étude, la fréquence w sera comprise entre 0 et 10 Hz, le pas d'échantillonnage T_e sera égal à 0.001 s (i.e 1 ms), et le délai δ sera fixé à trois valeurs différentes 10 ms, 30 ms et 100 ms.

Généralités

Si le gain du filtre (ou module) est différent de 1 alors le signal de sortie du filtre sera déformé. Pour un gain supérieur à 1, le signal d'entrée sera amplifié, à contrario un gain inférieur à 1 entraînera une atténuation du signal d'entrée. Si la phase du filtre est différente de 0 alors le signal de sortie sera déphasé par rapport au signal d'entrée. Une phase positive correspond à une avance tandis qu'une phase négative correspond à un retard. Plus le gain sera proche de 1 et plus la phase sera proche de 0 et donc meilleur sera le prédicteur.

Caractéristiques de FP1 pour un délai de 10 ms

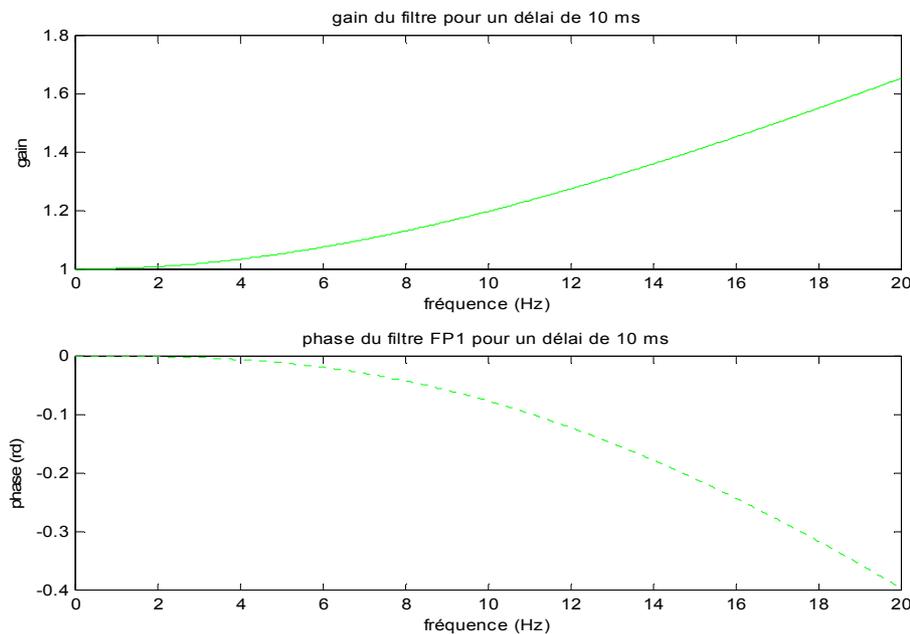


figure 5.3 : gain et phase du filtre FP1 pour un δ de 10 ms.

On observe que le gain est supérieur à 1, nous aurons alors une amplification du signal. Le gain est également une fonction croissante de la fréquence, le signal sera d'autant plus amplifié qu'il possède des hautes fréquences. La phase est une fonction décroissante (dans cette plage de fréquences) de la fréquence, de plus elle est négative, donc le signal de sortie du filtre sera retardé par rapport au signal d'entrée.

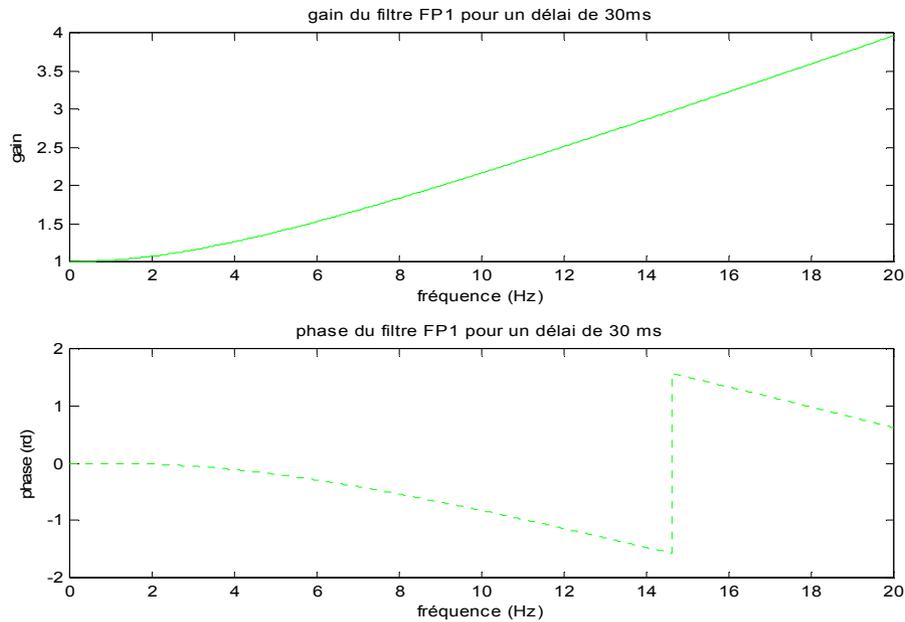
Caractéristiques de FP1 pour un délai de 30 ms

figure 5.4 : gain et phase du filtre FP1 pour un δ de 30 ms.

On peut observer que le gain est supérieur au gain de FP1 pour un délai de 10 ms, ceci signifie donc que plus le délai augmente et plus le signal de sortie du filtre sera déformé par rapport au signal d'entrée. On détecte également une discontinuité dans la phase pour une fréquence d'environ 15 Hz, ceci correspond aux pôles de la phase du filtre.

Caractéristiques de FP1 pour un délai de 100 ms

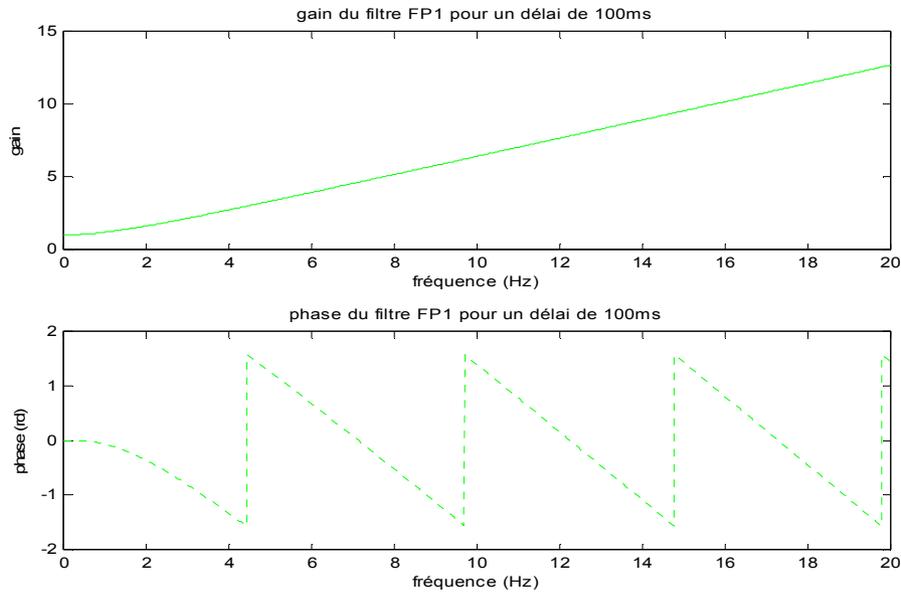


figure 5.5 : gain et phase du filtre FP1 pour un δ de 100 ms.

On peut voir que le gain continue d'augmenter avec la fréquence, on détecte également une périodicité de la phase, qui est une fonction de deux fonctions périodiques cosinus et sinus.

5.4 Le prédicteur d'ordre 2

Dans ce cas le signal prédit est une fonction du signal d'entrée retardé et de sa dérivée première et de sa dérivée seconde. Le filtre associé à la prédiction d'ordre 2 sera dorénavant nommé FP2.

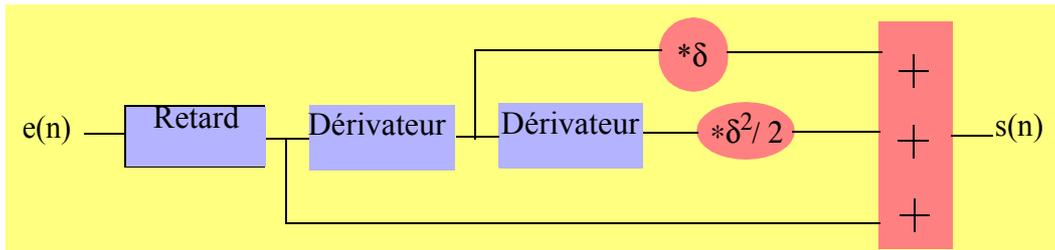


figure 5.6 : circuit permettant de réaliser la prédiction d'ordre 2

5.4.1 Calcul du module et de la phase du filtre FP2

Le signal de sortie s'exprime par la relation suivante:

(EQ 7)

$$s(n) = e(n - \delta) + \delta \times \frac{d}{dn} e(n - \delta) + \frac{\delta^2}{2} \times \frac{d^2}{dn^2} e(n - \delta)$$

on passe alors en transformée en Z:

$$S(z) = E(z) \times z^{-\delta} + \delta \times z^{-\delta} \times (1 - z^{-1}) \times E(z) + \frac{\delta^2}{2} \times z^{-\delta} \times (1 - z^{-1})^2$$

On peut alors évaluer la fonction de transfert du filtre associé à la prédiction d'ordre 2:

$$H(z) = \frac{S(z)}{E(z)} = z^{-\delta} \times \left[1 + \delta \times (1 - z^{-1}) + \frac{1}{2} \times [\delta \times (1 - z^{-1}) \times \delta \times (1 - z^{-1})] \right]$$

(EQ 8)

Afin de mieux appréhender les résultats, il nous faut calculer la transmittance isochrone, en effectuant le changement de variables suivant: $z = e^{jwT_e}$, l'équation (EQ 8) devient alors:

$$H(w) = e^{-jw\delta T_e} \times [1 + \delta \times (1 - e^{-jwT_e})] + \frac{1}{2} e^{-jw\delta T_e} \times [\delta \times (1 - e^{-jwT_e}) \times \delta \times (1 - e^{-jwT_e})]$$

on pose : $A = w\delta T_e$ $B = wT_e$ on obtient alors:

$$\begin{aligned} \text{Re}[H(w)] = & \cos A \times \left(1 + \delta - \delta \cos B + \frac{\delta^2}{2} - \delta^2 \cos B + \frac{\delta^2}{2} \cos 2B \right) \\ & + \sin A \times (\delta \sin B + \delta^2 \sin B - \delta^2 \sin B \cos B) \end{aligned}$$

$$\begin{aligned} \text{Im}[H(w)] = & \cos A \times (\delta \sin B + \delta^2 \sin B - \delta^2 \sin B \cos B) \\ & - \sin A \times \left(1 + \delta - \delta \cos B + \frac{\delta^2}{2} - \delta^2 \cos B + \frac{\delta^2}{2} \cos 2B \right) \end{aligned}$$

On peut alors déterminer le gain et la phase du filtre FP2 à partir des formules suivantes:

$$|H(w)| = \sqrt{\text{Re}^2[H(w)] + \text{Im}^2[H(w)]}$$

$$\phi(w) = \text{atan}\left(\frac{\text{Im}[H(w)]}{\text{Re}[H(w)]}\right) \quad (\text{EQ 9})$$

5.4.2 Représentation graphique du module et de la phase du filtre FP2

En respectant le cadre de notre étude, la fréquence w sera comprise entre 0 et 10 Hz, le pas d'échantillonnage T_e sera égal à 0.001 s (i.e 1 ms), et le délai δ sera fixé à trois valeurs différentes 10 ms, 30 ms et 100 ms.

Caractéristiques de FP2 pour un délai de 10 ms

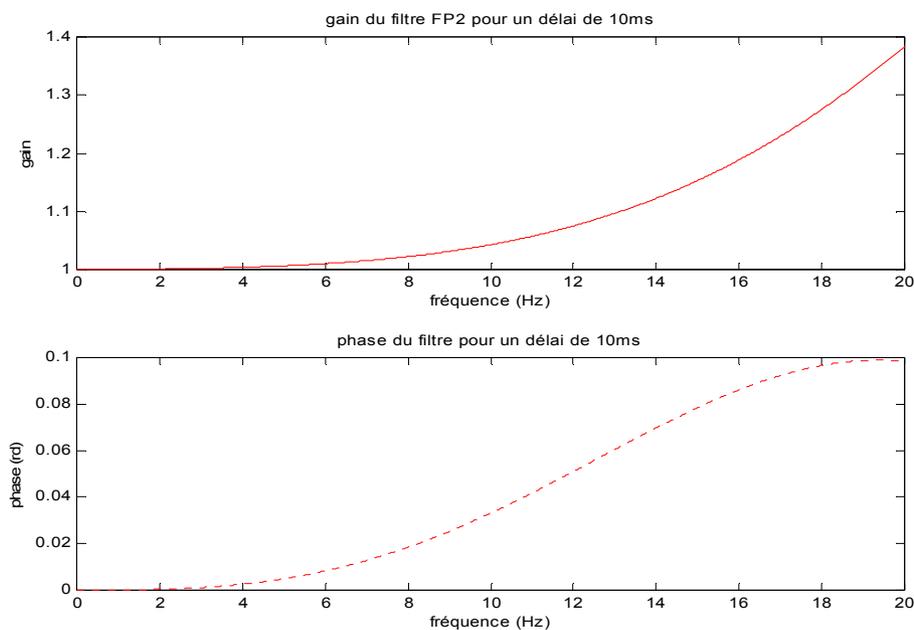


figure 5.7 : gain et phase du filtre FP2 pour un δ de 10 ms.

Ici on peut constater que le gain est supérieur à 1, nous sommes donc en présence d'un filtre amplificateur, la phase est pour sa part positive et croissante (dans cette plage de fréquences) ce qui entraînera une avance du signal de sortie du filtre par rapport au signal d'entrée.

Caractéristiques de FP2 pour un délai de 30 ms

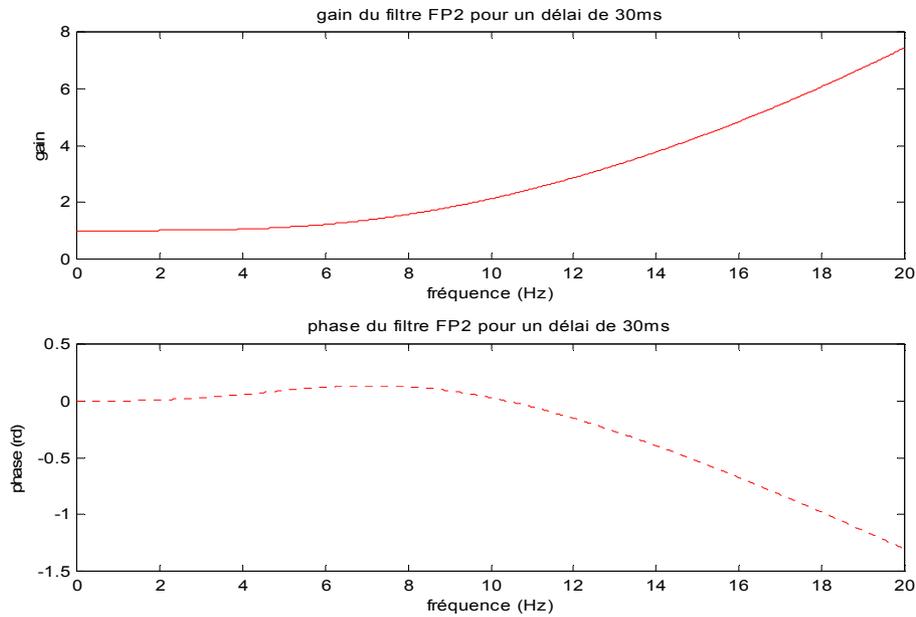


figure 5.8 : gain et phase du filtre FP2 pour un δ de 30 ms.

On constate pour le filtre FP2, une augmentation du gain en fonction du délai et de la fréquence. Ceci a aussi été observé pour le filtre FP1. La phase n'est plus dans cette plage de fréquences une fonction monotone.

Caractéristiques de FP2 pour un délai de 100 ms

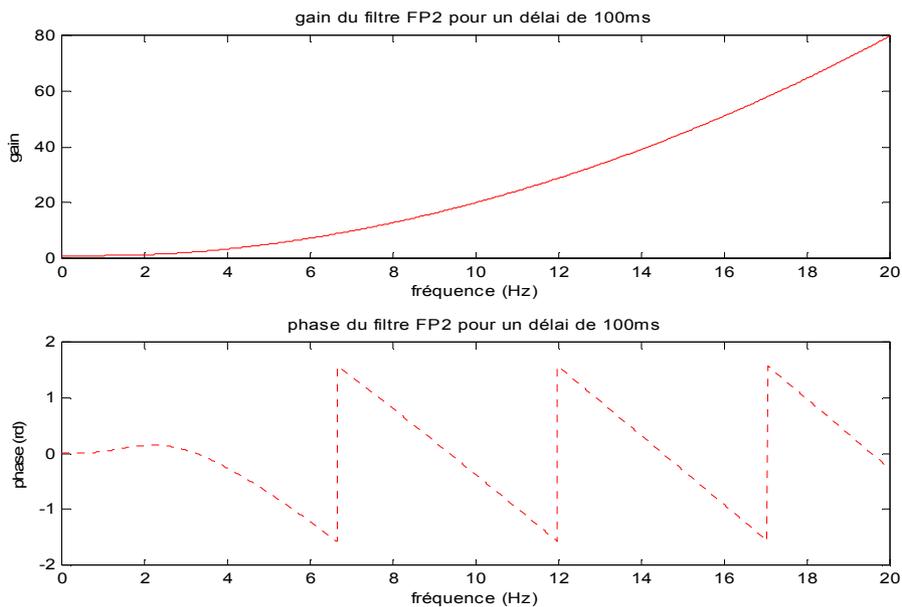


figure 5.9 : gain et phase du filtre FP2 pour un δ de 100 ms.

On observe que le gain du filtre FP2 continue d'augmenter avec le délai, de plus comme pour le filtre FP1, on voit apparaître la périodicité de la phase, ainsi que ses discontinuités liées aux pôles de la fonction $\phi(w)$.

5.5 Le prédicteur d'ordre 3

Dans ce cas le signal prédit est une fonction du signal d'entrée retardé et de sa dérivée première ainsi que de sa dérivée seconde et de sa dérivée tierce. Le filtre associé à la prédiction d'ordre 3 sera dorénavant nommé FP3.

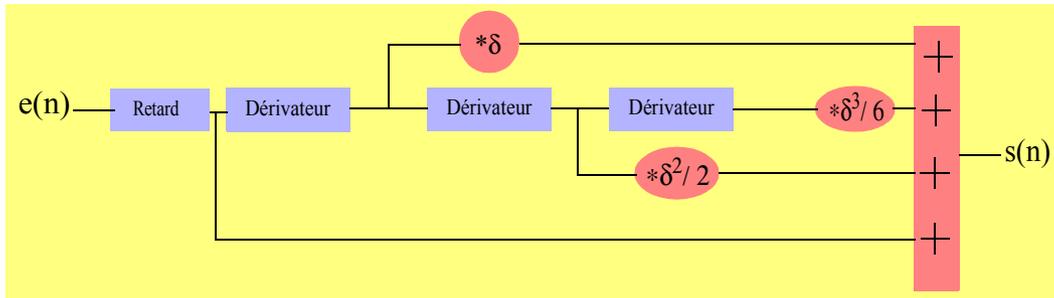


figure 5.10 : circuit permettant de réaliser la prédiction d'ordre 3.

5.5.1 Calcul du module et de la phase du filtre FP3

Le signal de sortie s'exprime par la relation suivante:

$$s(n) = e(n - \delta) + \delta \times \frac{d}{dn} e(n - \delta) + \frac{\delta^2}{2} \times \frac{d^2}{dn^2} e(n - \delta) + \frac{\delta^3}{6} \times \frac{d^3}{dn^3} e(n - \delta) \quad (\text{EQ 10})$$

on passe alors en transformée en Z:

$$S(z) = E(z) \times z^{-\delta} + \delta \times z^{-\delta} \times (1 - z^{-1}) \times E(z) + \frac{\delta^2}{2} \times z^{-\delta} \times (1 - z^{-1})^2 + \frac{\delta^3}{6} \times z^{-\delta} \times (1 - z^{-1})^3$$

on peut alors évaluer la fonction de transfert du filtre associé à la prédiction d'ordre 3 :

$$H(z) = \frac{S(z)}{E(z)} = z^{-\delta} \times \left[1 + \delta \times (1 - z^{-1}) + \frac{1}{2} \times [\delta \times (1 - z^{-1}) \times \delta \times (1 - z^{-1})] + z^{-\delta} \times \left[\frac{1}{6} \times [\delta \times (1 - z^{-1}) \times \delta \times (1 - z^{-1}) \times \delta \times (1 - z^{-1})] \right] \right] \quad (\text{EQ 11})$$

Afin de mieux appréhender les résultats, il nous faut calculer la transmittance isochrone, en effectuant le changement de variables suivant : $z = e^{j\omega T_e}$, l'équation (EQ 11) devient alors:

$$H(\omega) = e^{-j\omega\delta T_e} \times \left[1 + \delta \times (1 - e^{-j\omega T_e}) + \frac{1}{2} \times [\delta \times (1 - e^{-j\omega T_e}) \times \delta \times (1 - e^{-j\omega T_e})] \right. \\ \left. + e^{-j\omega\delta T_e} \times \left[\frac{1}{6} \times [\delta \times (1 - e^{-j\omega T_e}) \times \delta \times (1 - e^{-j\omega T_e}) \times \delta \times (1 - e^{-j\omega T_e})] \right] \right]$$

on pose : $A = \omega\delta T_e$ $B = \omega T_e$

on obtient alors:

$$\begin{aligned} \text{Re}[H(\omega)] &= \cos A \times \left(1 + \delta - \delta \cos B + \frac{\delta^2}{2} - \delta^2 \cos B + \frac{\delta^2}{2} \cos 2B \right) \\ &+ \cos A \times \left(\frac{\delta^3}{6} - \frac{\delta^3}{2} \cos B + \frac{\delta^3}{2} \cos 2B + \frac{\delta^3}{2} \cos B (\sin B)^2 - \frac{\delta^3}{2} \cos B \cos 2B \right) \\ &+ \sin A \times (\delta \sin B + \delta^2 \sin B - \delta^2 \sin B \cos B) \\ &+ \sin A \times \left(\frac{\delta^3}{2} \sin B - \delta^3 \cos B \sin B + \frac{\delta^3}{3} (\cos B)^2 \sin B + \frac{\delta^3}{6} \cos 2B \sin B \right) \end{aligned}$$

(EQ 12)

$$\begin{aligned} \text{Im}[H(\omega)] &= \cos A \times (\delta \sin B + \delta^2 \sin B - \delta^2 \sin B \cos B) \\ &+ \cos A \times \left(\frac{\delta^3}{2} \sin B - \delta^3 \cos B \sin B + \frac{\delta^3}{3} (\cos B)^2 \sin B + \frac{\delta^3}{6} \cos 2B \sin B \right) \\ &- \sin A \times \left(1 + \delta - \delta \cos B + \frac{\delta^2}{2} - \delta^2 \cos B + \frac{\delta^2}{2} \cos 2B \right) \\ &- \sin A \times \left(\frac{\delta^3}{6} - \frac{\delta^3}{2} \cos B + \frac{\delta^3}{2} \cos 2B + \frac{\delta^3}{2} \cos B (\sin B)^2 - \frac{\delta^3}{2} \cos B \cos 2B \right) \end{aligned}$$

(EQ 13)

On peut alors déterminer le gain et la phase du filtre FP3 à partir des formules (EQ 12) et (EQ 13).

5.5.2 Représentation graphique du module et de la phase du filtre FP3

En respectant le cadre de notre étude, la fréquence w sera comprise entre 0 et 10 Hz, le pas d'échantillonnage T_e sera égal à 0.001 s (i.e 1 ms), et le délai δ sera fixé à trois valeurs différentes 10 ms, 30 ms et 100 ms.

Caractéristiques de FP3 pour un délai de 10 ms

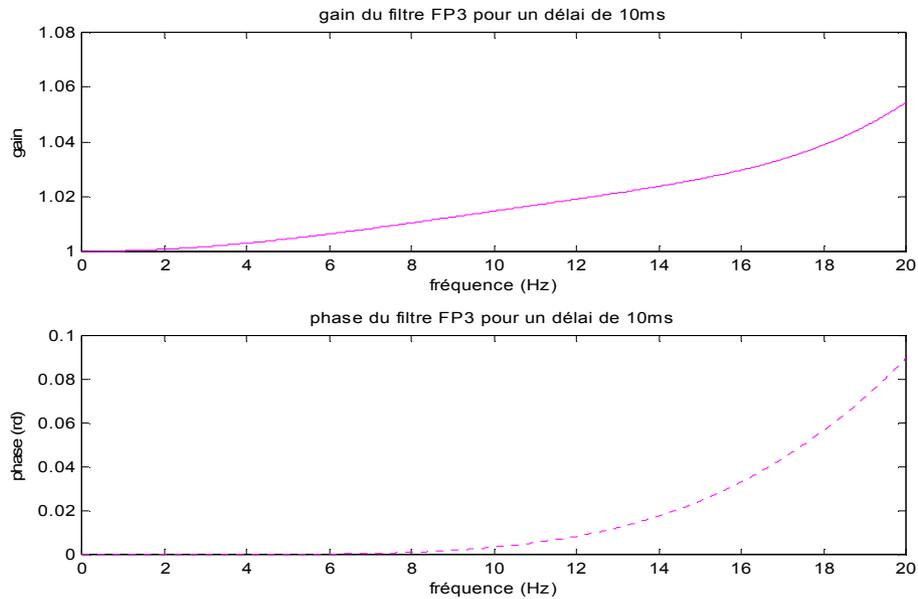


figure 5.11 : gain et phase du filtre FP3 pour un δ de 10 ms

On constate que le gain du filtre FP3 est légèrement supérieur à 1, nous sommes donc en présence d'un filtre amplificateur, ce qui se traduira par une légère augmentation de l'amplitude du signal de sortie par rapport à l'amplitude du signal d'entrée. De plus on observe une phase positive et croissante (dans cette plage de fréquences) ce qui entraînera un léger décalage sous la forme d'avance du signal de sortie par rapport à celui de l'entrée.

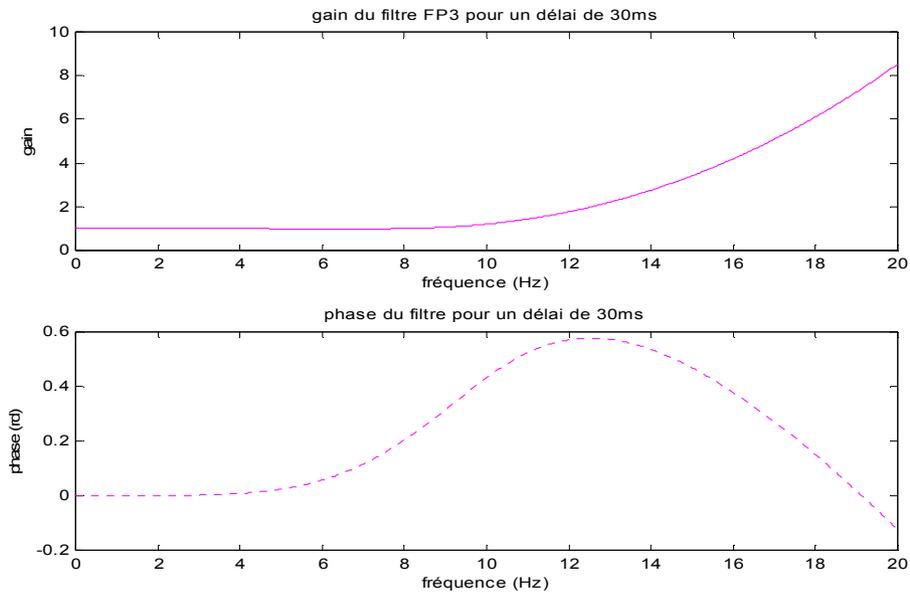
Caractéristiques de FP3 pour un délai de 30 ms

figure 5.12 : gain et phase du filtre FP3 pour un δ de 30 ms.

Comme pour les filtres FP1 et FP2, on détecte une augmentation du gain du filtre en fonction du délai et de la fréquence. La phase n'est plus une fonction monotone et croissante de la fréquence et l'on voit apparaître le premier pôle de la fonction $\phi(\omega)$ situé aux alentours de 19 Hz pour un délai de 30 ms.

Caractéristiques de FP3 pour un délai de 100 ms

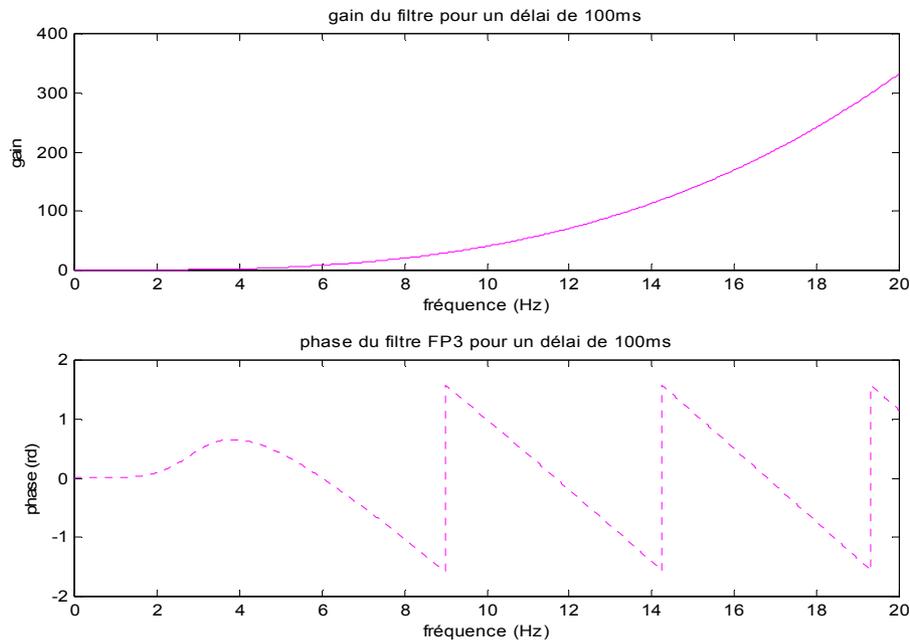


figure 5.13 : gain et phase du filtre FP3 pour un δ de 100 ms.

On observe que le gain du filtre FP3 continue d'augmenter avec le délai, de plus comme pour les filtres FP1 et FP2, on découvre l'apparition d'une périodicité dans la phase du filtre FP3. La périodicité observée sur la phase, provient du fait que la phase est une fonction de fonction en l'occurrence c'est une fonction de cosinus et de sinus qui sont deux fonctions périodiques. On peut également observer des discontinuités dans la phase, elles sont directement liées aux pôles de la fonction $\phi(w)$.

5.6 Les prédicteurs d'ordre plus élevé

D'après les résultats précédents, nous pourrions nous intéresser au développement des prédicteurs d'ordre plus élevé afin d'améliorer le gain du filtre associé, notamment pour les basses fréquences, mais ceci n'est malheureusement pas possible. Comme nous l'avons vu dans le chapitre précédent, le fait de calculer la vitesse, l'accélération et sa dérivée de manière instantanée, amplifie considérablement le bruit de quantification. A chaque dérivation, les hautes fréquences du signal sont amplifiées. C'est pourquoi on utilise un filtre passe bas afin d'extraire la partie basse fréquence qui nous intéresse. De plus la partie filtrée du signal va être multipliée selon l'ordre de la prédiction par une puissance de δ , ceci provoquera des oscillations perceptibles par l'utilisateur dans le signal de sortie du filtre associé à l'une des prédictions. En fait, la limitation dans l'utilisation d'ordres plus élevés est directement liée aux performances du filtre passe bas employé pour éliminer le bruit présent dans les signaux réels. On ne peut

réduire infiniment la fréquence de coupure du filtre passe bas tout en gardant une atténuation très forte et un faible déphasage.

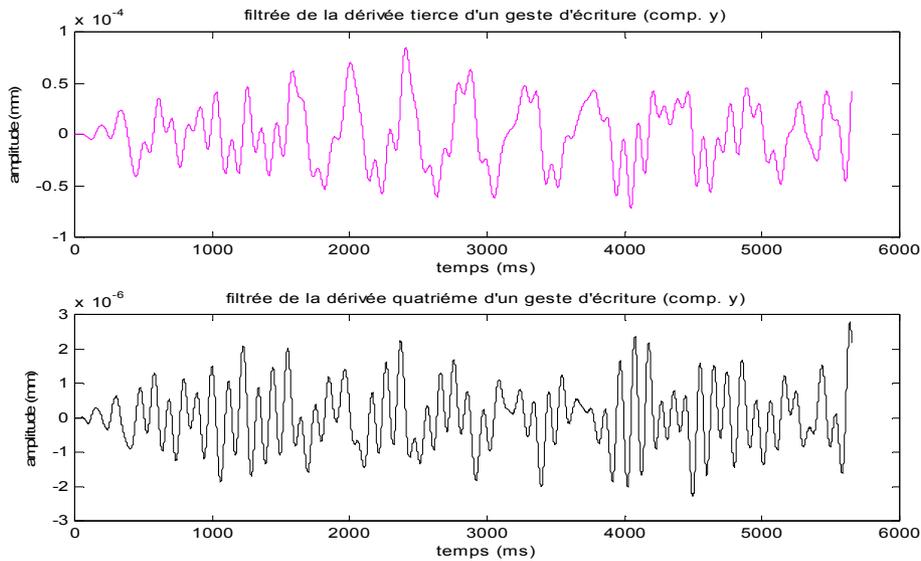


figure 5.14 : en haut, la filtrée de la dérivée tierce d'un geste d'écriture.

On peut observer la dérivée tierce filtrée passe bas de la composante y d'un signal d'écriture, ainsi que la dérivée quatrième également filtrée passe bas. On voit alors que le signal filtré issu de la dérivée quatrième est plus oscillant que le signal provenant du filtrage passe bas de la dérivée tierce. Ceci est d'autant plus préjudiciable pour le signal prédit que la contribution de la dérivée quatrième sera multipliée par $\delta^4/4!$ tandis que la contribution de la dérivée tierce sera, comme nous l'avons vu dans ce chapitre, multipliée par le facteur $\delta^3/3!$. Aussi pour limiter les oscillations de la force répercutée dans la main de l'utilisateur, on se contentera pour des délais avoisinant les 30 ms d'utiliser des prédicteurs d'ordre inférieur ou égal à trois.

5.7 Récapitulatif des résultats

5.7.1 Comparaison des performances des trois filtres pour un délai de 10 ms

Pour un geste quelconque

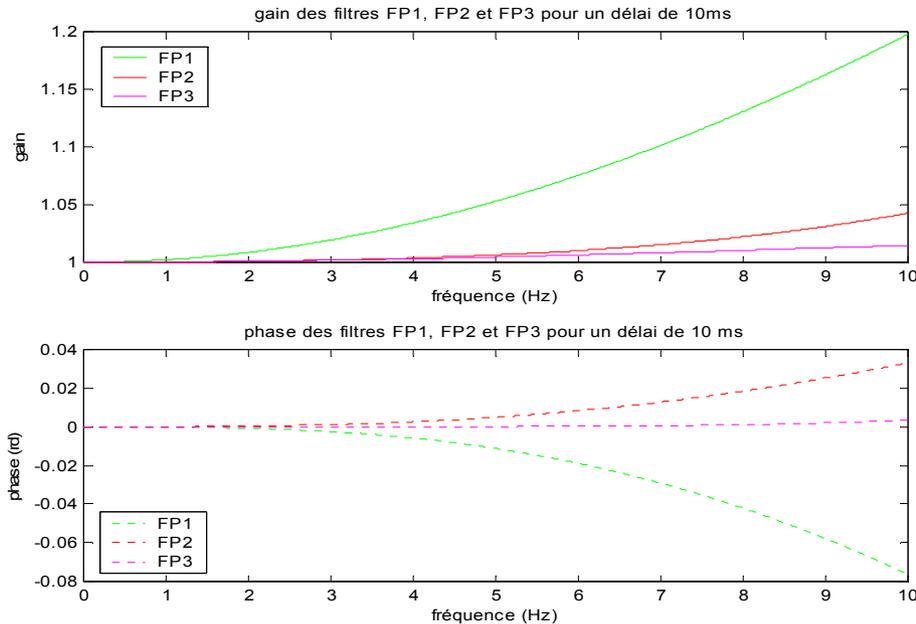


figure 5.15 : gain et phase des filtres FP1, FP2, FP3 avec un délai de 10 millisecondes pour un geste quelconque.

Dans le cas d'un geste quelconque la plage de fréquences utiles est comprise entre 0 et 10 Hz, on observe que le gain du filtre FP1 diverge beaucoup plus rapidement que le gain de FP2 et celui de FP3 (dans cette plage de fréquences). Rappelons que plus le gain est proche de 1 et meilleures seront les performances du prédicteur associé. On constate en effet que le filtre FP3 est le filtre qui amplifiera le moins le signal de sortie par rapport au signal d'entrée. L'amplification sera d'ailleurs inférieure à 2% dans toute la plage de fréquences. Rappelons également que plus la phase est proche de 0 et plus le déphasage "entrée-sortie" sera faible. On observe sur le schéma représentant la phase en fonction de la fréquence que le filtre FP3 entraînera le minimum de décalage "entrée-sortie" dans cette plage de fréquences, ce décalage dû à la phase positive sera donc une avance. Enfin dans les conditions préalablement citées, le classement s'effectuant du meilleur au moins bon sera le suivant, en premier FP3, puis FP2 et enfin le filtre FP1 en dernier.

Pour un geste d'écriture

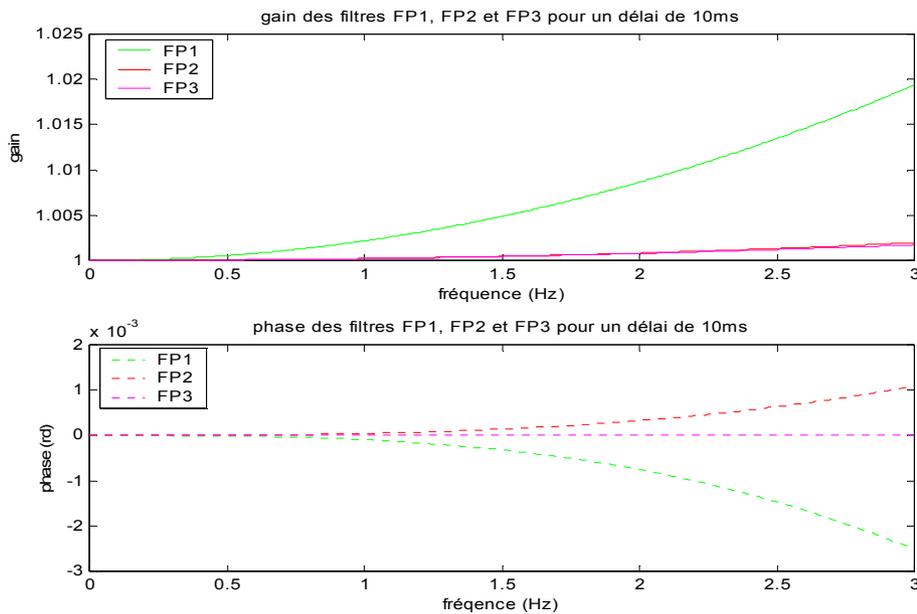


figure 5.16 : gain et phase des filtres FP1, FP2, FP3 avec un délai de 10 millisecondes pour un geste d'écriture.

Dans ce cas la plage utile des fréquences s'étale de 0 à 3 Hz. On note que le gain du filtre FP1 diverge des deux autres, l'amplification du filtre FP2 et FP3 est pratiquement la même, on doit donc s'attendre à des signaux de sortie très proches pour ces deux filtres. La phase du filtre FP3 reste néanmoins la plus proche de 0, on peut également établir le classement suivant: FP3 légèrement meilleur que le FP2 qui est lui-même supérieur au filtre FP1.

5.7.2 Comparaison des performances des trois filtres pour un délai de 30 ms

Pour un geste quelconque

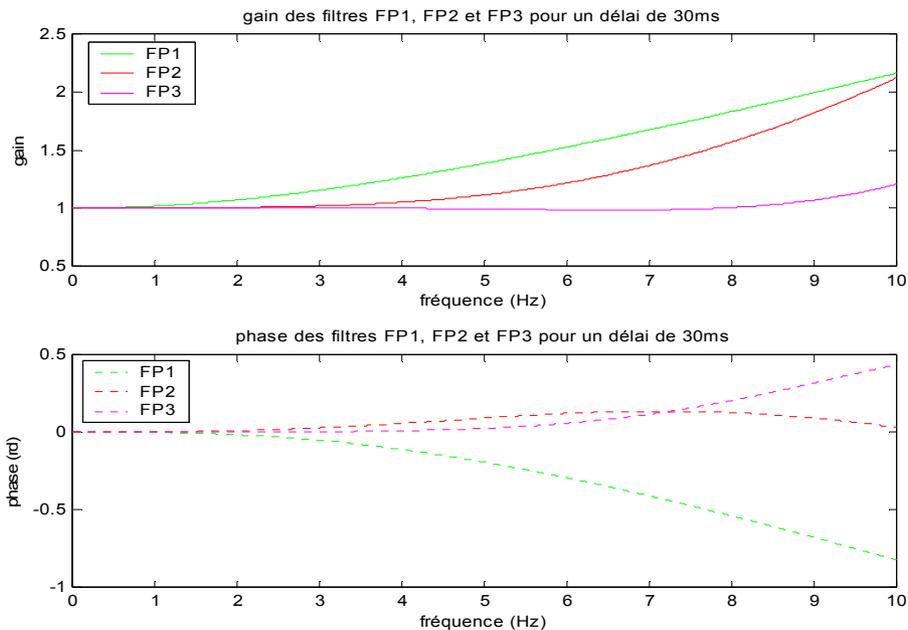


figure 5.17 : gain et phase des filtres FP1, FP2, FP3 avec un délai de 30 millisecondes pour un geste quelconque.

Dans ce cas la plage utile des fréquences est [0-10Hz], le gain du filtre FP3 reste le meilleur tandis que le gain du filtre FP2 tend vers le gain du filtre FP1 aux limites supérieures de la plage de fréquences. Pour un signal d'entrée possédant des fréquences de l'ordre de 8 Hz, l'amplification des filtres FP1 et FP2 est de l'ordre de 50%, ce qui est très préjudiciable pour le signal issu de la prédiction. La phase du filtre FP2 est la plus proche (en moyenne) de 0 sur la totalité de la plage des fréquences. Compte tenu de l'amplification du filtre FP2 enregistrée pour des fréquences supérieures à 6 Hz, on peut à nouveau établir le classement suivant : la meilleure prédiction sera établie par le filtre FP3, puis vient ensuite le filtre FP2 et enfin en dernière position le filtre FP1.

Pour un geste d'écriture

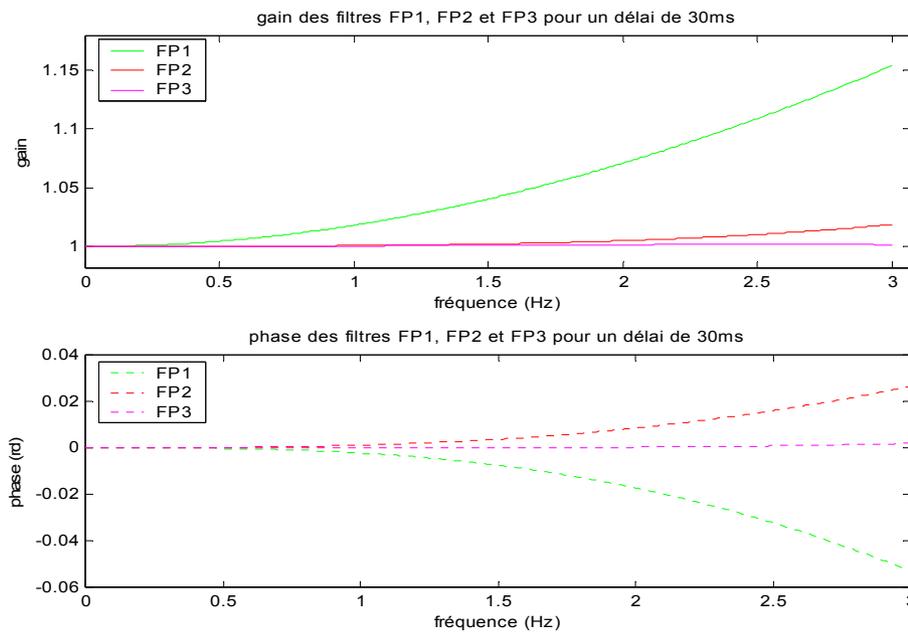


figure 5.18 : gain et phase des filtres FP1, FP2, FP3 avec un délai de 30 millisecondes pour un geste d'écriture.

Nous sommes de nouveau dans une plage utile des fréquences de [0-3Hz], on note une forte divergence du gain du filtre FP1 par rapport aux deux autres filtres. Le gain de FP3 reste très proche de 1 tout au long de la plage des fréquences. De plus, la phase de FP3 reste également très proche de 0 sur toute la plage, ce qui lève tous les doutes quant au choix du meilleur prédicteur dans les conditions présentes ($\delta = 30\text{ms}$ $f \in [0,3\text{Hz}]$). Le filtre FP3 sera donc plus performant que le filtre FP2 qui sera lui-même plus performant que le filtre FP1 associé à une prédiction d'ordre 1.

5.7.3 Comparaison des performances des trois filtres pour un délai de 100 ms

Pour un geste quelconque

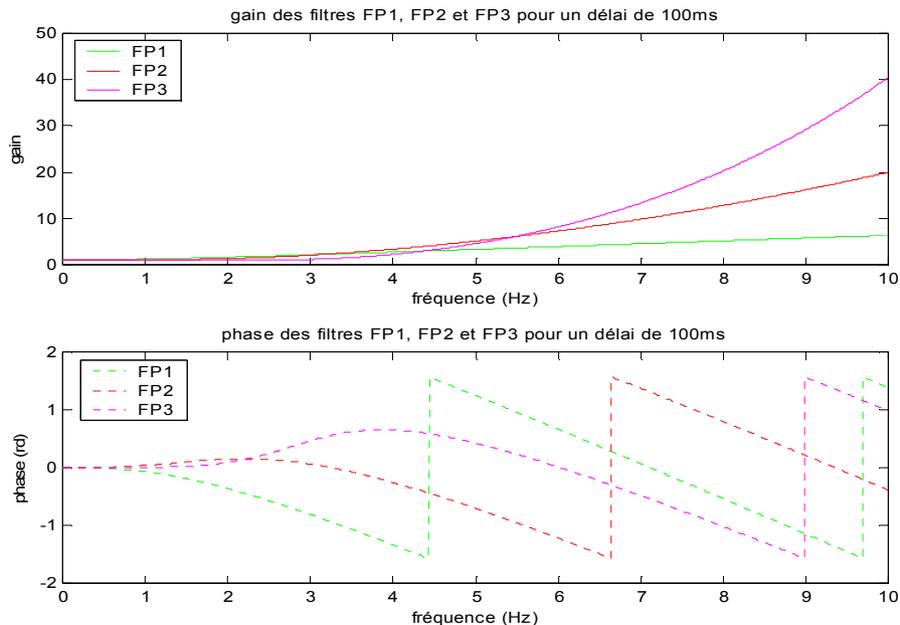


figure 5.19 : gain et phase des filtres FP1, FP2, FP3 avec un délai de 100 millisecondes pour un geste quelconque.

Dans les conditions suivantes ($\delta = 100\text{ms}$ $f \in [0,10\text{Hz}]$), on constate que le gain le plus proche de 1 est celui du filtre FP1, ce gain avoisine 5 pour une fréquence de 10 Hz ce qui aura pour conséquence une augmentation de 500% de l'amplitude du signal de sortie par rapport au signal d'entrée. La construction du signal prédit à partir du filtre FP1 sera alors très différente du signal d'entrée, et cette construction sera encore moins bonne avec des prédicteurs d'ordre plus élevé car leurs gains seront encore plus importants que celui du filtre FP1. De plus le déphasage dans cette plage de fréquence devient une fonction périodique de la fréquence. La phase des filtres prédicteurs sera tantôt positive et tantôt négative selon les fréquences présentes dans le signal d'entrée, ceci aura pour effet de déformer de façon importante le signal de sortie par rapport au signal d'entrée. Dans cette configuration, les prédicteurs donneront de très mauvais résultats qui seront pratiquement inexploitable car l'erreur de prédiction sera très importante, donc le signal prédit ne ressemblera plus du tout au signal que l'on désire prédire, on atteint ici les limites des prédicteurs FP1, FP2 et FP3.

Pour un geste d'écriture

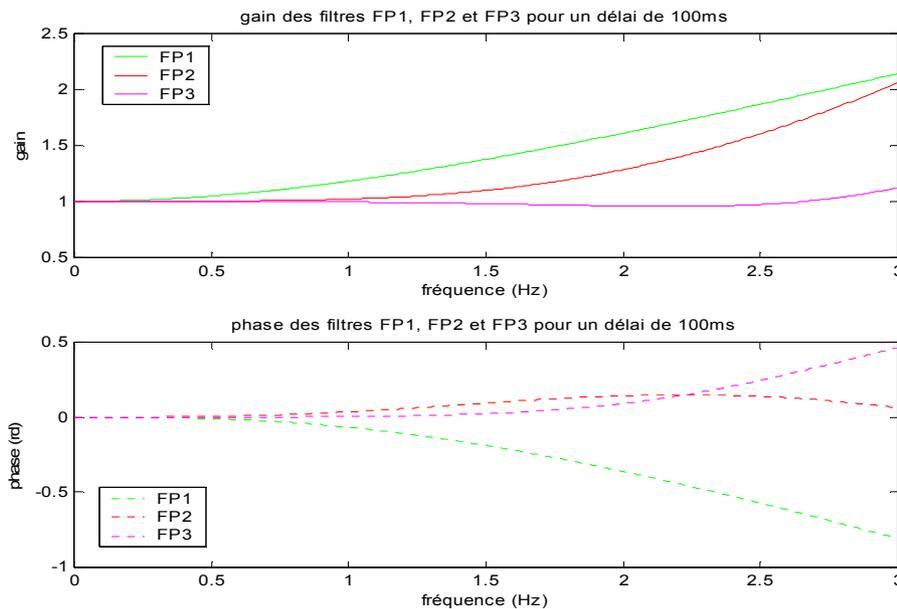


figure 5.20 : gain et phase des filtres FP1, FP2, FP3 avec un délai de 100 millisecondes pour un geste d'écriture.

Dans les conditions suivantes ($\delta = 100\text{ms}$ $f \in [0,3\text{Hz}]$), le gain du filtre FP3 reste le plus proche de 1, il s'en écarte d'ailleurs assez peu. Il en résultera que le signal de sortie du filtre FP3 sera très proche en terme d'amplitude du signal que l'on désire prédire. Les gains des filtres FP2 et FP3 dépassent assez rapidement la valeur 1.5 ce qui entraîne une amplification du signal de sortie de l'ordre de 50% par rapport au signal à prédire, et donc invalide complètement dans les conditions présentées ci-dessus, l'utilisation de ces filtres dans la prédiction. La phase du filtre FP2 reste néanmoins plus proche du 0 en moyenne, donc le signal prédit par le prédicteur d'ordre 2 sera le moins décalé par rapport au signal à prédire. Malgré ceci, le meilleur prédicteur reste le filtre FP3 devançant de loin le filtre FP2 et une nouvelle fois le filtre FP1 occupe la dernière place.

5.8 Conclusion

Dans cette étude théorique des filtres associés aux différentes prédictions, on relève un élément essentiel et commun à l'utilisation des filtres FP1, FP2 et FP3. Il s'agit des conditions initiales d'expérimentation. En effet, le gain et la phase de chacun des filtres prédicteurs sont fonction des fréquences présentes dans le signal d'entrée, du délai résultant de la latence du réseau, et enfin du pas d'échantillonnage du signal d'entrée. En fait, les performances de chacun des prédicteurs diminuent lorsque la fréquence maximale présente dans le signal d'entrée augmente. De la même façon plus le délai augmente et plus le signal issu des prédicteurs sera différent du signal d'entrée. On peut également conclure que le prédicteur d'ordre 3 possède les meilleures performances dans le cas d'un geste d'écriture dont la fréquence maximale ne sera pas supérieure à 3Hz pour un délai allant jusqu'à 100 ms. Pour un geste quelconque, possédant une fréquence

maximale (sur chaque composante x,y,z) inférieure à 10 Hz, le filtre prédicteur FP3 reste utilisable jusqu'à un délai de 27 ms pour laquelle l'amplification sera inférieure à 10%. Enfin le pas d'échantillonnage agit également sur les performances des prédicteurs, plus il est élevé et plus les performances baissent.

Chapitre 6

Les supports techniques du télégeste

Introduction. Ce chapitre aborde les différents réseaux de télécommunications qui ont été testés dans le cadre du télé geste. On présente les exigences de qualité de service de ces applications (haptique) *temps réel*. Nous exposons également les arguments qui ont amené à choisir le réseau Numéris dans un premier temps.

En premier lieu, nous faisons une présentation générale du modèle OSI suivie d'une description du RNIS (Réseau Numérique à Intégration de Services) [SER 97]. Puis nous présentons les divers protocoles (IP, TCP, UDP) utilisés dans les télécommunications [ROL 97]. Nous poursuivons par une présentation de l'architecture réseau mise en oeuvre pour la télé-échographie. Enfin, nous concluons par une énumération des problèmes rencontrés lors de la transmission d'un geste.

6.1 Le modèle OSI

Des efforts de normalisation effectués au niveau national et international permettent une approche logique et unifiée dans la conception d'une architecture de réseau [MAC 91]. Débutés en 1977, les travaux de l'I.S.O. (International Standards Organisation) ont abouti en 1984 à une norme internationale connue sous le nom de modèle de référence O.S.I. (Open System Inteconnexion).

Ce modèle a pour but d'établir une structuration permettant de décomposer l'architecture d'un réseau en sept couches auxquelles correspondent des protocoles.

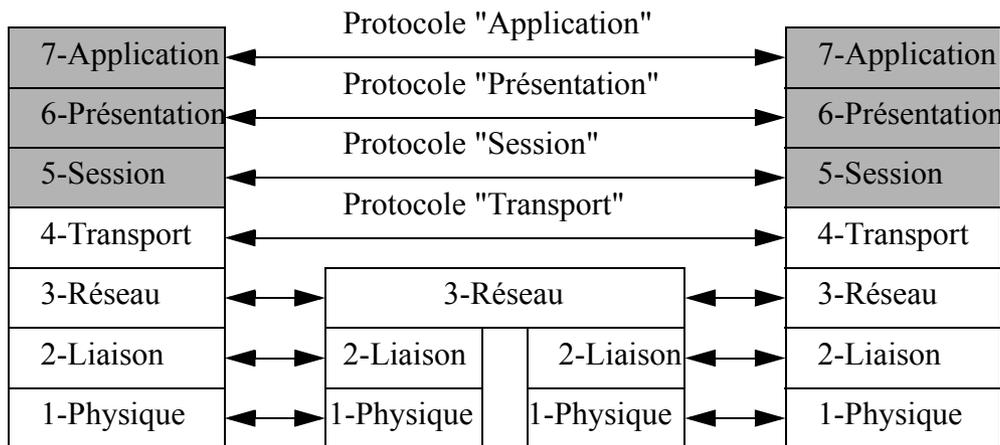


figure 6.1 : Modèle OSI - Norme ISO 7498 (Recommandation X200 du CCITT).

Un protocole est un ensemble de règles et de formats d'informations qui permettent aux couches de même niveau de communiquer. Chaque couche réalise un certain nombre de fonctions dont bénéficie la couche supérieure : la couche N offre des services à la couche N+1.

L'intérêt de la décomposition en couches est le nombre limité d'interfaces qu'un élément primaire doit connaître. En outre le modèle OSI constitue un cadre conceptuel, comportant un ensemble de règles formelles pour l'élaboration des architectures de réseaux de communication [HEN 91].

Le modèle OSI permet une bonne compréhension des différents éléments qui interviennent dans la conception d'un réseau. Toutes les couches ne sont pas nécessairement utilisées. Par exemple, une simple connexion entre deux ordinateurs identiques ne nécessite pas tous les niveaux. En revanche, un réseau d'ordinateurs hétérogènes nécessite la présence de toutes les couches afin d'obtenir des interfaces standards pour les stations.

Lorsqu'elles passent d'une couche à l'autre, dans le sens des numéros de couches décroissants, les données à transmettre sous la forme d'unités de données, sont parfois découpées pour former des entités plus petites. Des informations supplémentaires (entêtes de protocoles), qui traduisent le travail de chaque couche

sont ajoutées sur les unités de données (encapsulation des données). La démarche inverse est effectuée à la réception.

Fonctionnalités des différentes couches :

Couche physique : elle fournit l'ensemble des moyens nécessaires à la transmission de suites d'éléments binaires. Elle couvre tout ce qui concerne la liaison physique :

Caractéristiques physiques du support (type de media, topologie, distance entre deux abonnés ou entre deux répéteurs...)

Caractéristiques des signaux propagés (nature et caractéristiques du signal, mode de transmission...)

Couche liaison : elle est responsable de l'acheminement sans erreur de blocs d'informations sur la liaison physique, et assure au minimum un service de détection d'erreurs. Elle prend en charge l'établissement et la libération de liaisons de données sur le support physique offert par la première couche. Lorsque plusieurs équipements se partagent un même support (technologies multipoints), elle assure la gestion des accès.

Elle doit éventuellement effectuer (en fonction de l'architecture du réseau, c'est à dire des types de protocoles qui sont retenus pour les couches supérieures, et de la qualité de service globale attendue), un contrôle de flux, une fonction de séquençement ou la correction des erreurs de transmission.

A ce niveau, l'entité manipulée n'est plus un signal électrique mais un train de données binaires, formatées et organisées.

Pour résumer, la couche liaison assure les services de contrôle d'accès au canal de transmission et le contrôle du lien logique. Elle est responsable de la transmission des unités de données, avec au minimum un service de détection d'erreurs.

Couche réseau : Elle est responsable de l'acheminement des paquets de données qui peuvent traverser plusieurs noeuds intermédiaires, c'est le routage. Elle doit offrir un service d'adressage.

Elle réalise un contrôle de flux et assure la détection et la correction d'erreurs. Elle peut offrir, de façon facultative, le séquençement des messages, un service de transfert accéléré de paquets, une segmentation permettant d'assurer la compatibilité des longueurs de blocs entre sous-réseaux, ainsi qu'une fonction de multiplexage.

Couche transport : Elle constitue une frontière stable entre le monde des traitements applicatifs (couches 5-6-7) et celui des transmissions. Elle assure l'indépendance de l'équipement par rapport au réseau (c'est à dire indépendante des moyens de mise en oeuvre pour la transmission). Elle complète la qualité de services offerte par les couches inférieures. Les fonctions prises en charge par cette couche sont donc déterminées en fonction des couches inférieures et de la qualité globale de services attendue par les applications.

Les normes ISO définissent 5 classes de service, offrant une qualité plus ou moins complète :

- Classe 0 : qualité de service minimale, pas de multiplexage, pas de reprise sur erreur, la couche réseau est sensée lui fournir un taux d'erreurs et un taux de coupures de connexions acceptables. La couche transport assure alors la segmentation / réassemblage.
- Classe 1 : capacité réduite de détection et de reprise sur erreur (fermeture parasite d'une connexion, réception de données n'appartenant pas à une connexion connue). La couche transport assure alors la segmentation / réassemblage, et la reprise sur erreur signalée.
- Classe 2 : la couche transport classe 2 assure la segmentation / réassemblage, le multiplexage, et en option le contrôle de flux et la reprise sur erreur.
- Classe 3 : elle assure la segmentation / réassemblage, le multiplexage, le contrôle de flux la reprise sur erreur signalée.
- Classe 4 : c'est la classe la plus "complète", elle assure la segmentation / réassemblage, le multiplexage, le contrôle de flux et la reprise sur erreur.

Couche session : c'est la première couche, dans la hiérarchie OSI, qui ne concerne plus la transmission de données en tant que telle. Elle n'assure ni contrôle ni reprise en cas d'erreur. Elle fournit les moyens nécessaires à l'organisation et la synchronisation du dialogue entre les deux entités communicantes (initialisation et maintien d'une session, identification mutuelle, synchronisation du dialogue).

Couche présentation : cette couche est responsable de la présentation des données échangées par les applications. Le but est d'assurer la compatibilité entre les différents types de systèmes hétérogènes raccordés au réseau.

Couche application : cette couche peut contenir une importante quantité de services différents les uns des autres. Elle assure l'interface utilisateur et offre des services de communication dédiés à différents types d'applications.

6.2 Présentation du RNIS et de l'IP

6.2.1 Le RNIS

Le Réseau Numérique à Intégration de Services (RNIS) est l'évolution naturelle du réseau téléphonique [PUJ 89]. En effet, les opérateurs de tous les pays ont depuis plusieurs années modernisé leur réseau en remplaçant leurs centraux analogiques par des équipements basés entièrement sur la technologie numérique. Ainsi, aujourd'hui, le réseau véhicule des données (voix, images...) dans les canaux numériques à 64 Kb/s.

Numéris est le nom commercial du RNIS de France Télécom. C'est la prolongation jusque chez l'abonné de cette modernisation. Le RNIS, grâce à l'information de son réseau, intègre un certain nombre de services dont les principaux sont les suivants :

- discrimination des appels voix, données, télécopie (fax), etc...
- présentation du numéro appelant à l'appelé
- possibilité de sous-adressage et de sélection directe à l'arrivée
- pour la voix : possibilité de suspension, reprise, transfert d'appel, renvoi de terminal...

Outre les services intégrés, les atouts principaux sont les suivants :

- Rapidité : débit équivalent à 64 000 bps (450 kilo-octets par minute) sur un seul canal B, sans compression. Le débit est symétrique et donc l'envoi et la réception des données s'effectuent à la même vitesse. On note qu'il est possible d'augmenter la rapidité en utilisant la compression de données. On peut également regrouper plusieurs canaux en une seule voie logique.
- Fiabilité et garantie des débits : l'opérateur de télécommunication garantit la qualité de la transmission numérique (pas d'erreurs de transmission à corriger). Le débit réel est donc très proche du débit maximum théorique (64 000 bps sur un canal, 128 000 bps sur deux canaux), contrairement au réseau téléphonique commuté où le débit peut varier considérablement avec la qualité de la ligne (le débit peut chuter jusqu'à 9 600 bps).
- Mode synchrone : sur les liaisons numériques, on utilise généralement des profils de communications synchrones permettant de réduire à moins de 3% la partie de trafic consacrée au contrôle (contre 20% dans le cas des communications analogiques asynchrones du réseau téléphonique commuté).
- Temps de connexion court : environ 2 secondes pour le réseau français.
- Réseau commuté : on peut établir des connexions directes point à point entre les abonnés.
- Couverture : le RNIS offre une couverture internationale. En France la plupart du territoire est couvert : seuls les endroits les plus isolés ne sont pas desservis.

Deux types d'abonnement au réseau RNIS sont disponibles :

- *L'accès de base* : deux canaux B à 64 Kb/s pour le transport des informations en mode circuit et un canal D à 16 Kb/s pour la signalisation et le transport des données en mode paquet.

L'opérateur de télécommunication installe chez l'abonné une Terminaison Numérique de Réseau (TNR), de laquelle part un bus S0 comportant des prises permettant à l'abonné de connecter jusqu'à 8 équipements de communications différents. L'opérateur attribue à l'installation un seul numéro de téléphone. Avec le service de Sélection Directe à l'Arrivée, l'opérateur attribue à l'installation plusieurs numéros de téléphone. Le SDA permet une affectation sélective des appels entrant aux différents terminaux de l'installation.

- *L'accès primaire* : il comporte jusqu'à 30 canaux B à 64 Kb/s pour le transport des informations en mode circuit et d'un canal D à 64 Kb/s pour la signalisation et le transport des données en mode paquet.

L'opérateur de télécommunication attribue à l'installation un numéro de téléphone. Si l'abonné souscrit le service complémentaire de SDA, l'opérateur attribue à l'installation plusieurs numéros de téléphone.

Dans le cas d'un accès de base, il peut être utile d'utiliser un autocommutateur numérique privé (PABX) pour structurer le réseau numérique au sein des locaux. Du PABX partent différents câbles distribuant les prises sur lesquelles on raccorde les équipements de communication de l'installation :

- Équipements analogiques : modems, postes téléphoniques, télécopieurs...
- Bus S0 permettant de raccorder jusqu'à 8 équipements numériques : micro-ordinateurs, télécopieurs et téléphones RNIS...

A l'intérieur des locaux, les équipements sont distingués par des numéros de poste. Si l'abonné a souscrit le service de SDA, le PABX gère une table de correspondance entre les numéros de postes internes et les numéros du plan de numérotation téléphonique national attribués à l'installation.

L'utilisation d'un PABX est indispensable pour toute souscription à un accès primaire...

Un abonné ne disposant pas du service de SDA ou voulant distinguer plusieurs terminaux auxquels est affecté le même numéro de SDA peut recourir à la fonction de sous-adressage RNIS. La sous-adresse est une extension du numéro RNIS qui comporte jusqu'à 4 chiffres (0001 à 9999). Elle est choisie librement par l'abonné. Lors d'un appel, elle doit suivre le numéro et être séparée de celui-ci par le caractère *.

Autres services offerts par le RNIS :

- Distinction des appels voix et données : les terminaux en attente d'appel sont capables de trier les appels qui leur sont destinés grâce à l'information de Service Support (donnée, voix...) véhiculée par les messages de signalisation du canal D. Ainsi, par exemple, si une application de téléphonie et un logiciel de transfert de fichiers sont en attente d'appels sur un même accès de base (pas de SDA ni de sous-adresse), l'affectation de l'appel se fera correctement.
- Accès au réseau public X.25 par le canal D : le RNIS intègre dans un certain nombre de pays l'accès direct au réseau public X.25 en mode paquet (transpac pour la France). Ce service est ouvert par le canal D et offre un débit logique de 9 600 bps. Il est réalisé par un couplage entre le central téléphonique local dont dépend l'abonné et un autocommutateur X.25 du réseau public. Ce service nécessite bien entendu un abonnement spécifique contracté auprès de l'opérateur du réseau X.25.

6.2.2 Le TCP/IP

Les logiciels TCP/IP sont structurés en quatre couches de protocoles qui s'appuient sur une couche matérielle :

- *La couche de liens* est l'interface avec le réseau et est constituée d'un driver du système d'exploitation et d'une carte d'interface de l'ordinateur avec le réseau.
- *La couche réseau* ou couche IP (*Internet Protocol*) gère la circulation des paquets à travers le réseau en assurant leur routage. Elle comprend aussi les protocoles ICMP (*Internet Control Message Protocol*) et IGMP (*Internet Group Management Protocol*).
- *La couche transport* assure tout d'abord une communication de bout à bout en faisant abstraction des machines intermédiaires entre l'émetteur et le destinataire. Elle s'occupe de réguler le flux de données et assure un transport fiable (données transmises sans erreur et reçues dans l'ordre de leur émission) dans le cas de TCP (*Transmission Control Protocol*) ou non fiable dans le cas de l'UDP (*User Datagram Protocol*). Pour UDP, il n'est pas garanti qu'un paquet (appelé dans ce cas datagramme) arrive à bon port, c'est à la couche application de s'en assurer.

- *La couche application* est celle des programmes utilisateurs comme *telnet* (connexion à un ordinateur distant), *FTP (File Transfert Protocol)*, *SMTP (Simple Mail Transfert Protocol)* etc...

Cette architecture et ces différents protocoles permettent de faire fonctionner un réseau local, par exemple sur un bus Ethernet reliant un ordinateur *client A* à un *serveur FTP B*.

Le protocole IP est au coeur du fonctionnement d'internet [DOU 96]. Il assure sans connexion un service non fiable de délivrance de datagramme IP. Le service est non fiable car il n'existe aucune garantie pour que les datagrammes IP arrivent à destination. Certains peuvent être perdus, dupliqués, retardés, altérés ou remis dans le désordre. On parle de remise au mieux (*best effort delivery*) et ni l'émetteur ni le récepteur ne sont informés directement par IP des problèmes rencontrés. Le mode de transmission est non connecté car IP traite chaque datagramme indépendamment de ceux qui le précèdent et le suivent. Ainsi en théorie, au moins, deux datagrammes IP issus de la même machine ayant la même destination peuvent ne pas suivre obligatoirement le même chemin. Le rôle du protocole IP est centré autour des trois fonctionnalités suivantes :

- Définir le format du datagramme IP qui est l'unité de base des données circulant sur Internet.
- Définir le routage.
- Définir la gestion de la remise non fiable des datagrammes.

Le protocole UDP

Le protocole UDP utilise IP pour acheminer, d'un ordinateur à un autre, en mode non fiable des datagrammes qui lui sont transmis par une application. UDP n'utilise pas d'accusé de réception et ne peut donc pas garantir que les données ont bien été reçues. Il ne réordonne pas les messages si ceux-ci n'arrivent pas dans l'ordre dans lequel ils ont été émis et il n'assure pas non plus de contrôle de flux. Il se peut donc que le récepteur ne soit pas apte à faire face au flux de datagrammes qui lui arrivent. C'est donc à l'application qui utilise UDP de gérer les problèmes de perte de messages, duplications, retards, déséquencement,...

Cependant, UDP fournit un service supplémentaire par rapport à l'IP, il permet de distinguer plusieurs applications destinataires sur la même machine par l'intermédiaire des ports. Un port est une destination abstraite sur une machine identifiée par un numéro qui sert d'interface à l'application pour recevoir et émettre des données.

Le protocole TCP

Contrairement à UDP, TCP est un protocole qui procure un service de flux d'octets orienté connexion et fiable. Le terme *orienté connexion* signifie que les applications dialoguant à travers TCP sont considérées l'une comme serveur, l'autre comme client, et qu'elles doivent établir une connexion avant de pouvoir dialoguer (comme dans le cas de l'utilisation du téléphone). Les ordinateurs vérifient donc préalablement que le transfert est autorisé, que les deux machines sont prêtes en s'échangeant des messages spécifiques. Une fois que tous les détails ont été précisés, les applications sont informées qu'une connexion a été établie et qu'elles peuvent commencer leurs échanges d'informations. Il y a donc

exactement deux extrémités communiquant l'une avec l'autre sur une connexion TCP. Cette connexion est bi-directionnelle simultanée ("*full duplex*") et composée de deux flots de données indépendants et de sens contraire. Il est cependant possible d'inclure, dans l'en-tête de segments TCP d'une communication de A vers B des informations relatives à la communication de B vers A. Cette technique de superposition ("*piggybacking*") permet de réduire le trafic sur le réseau.

Tout au long de la connexion, TCP échange un flux d'octets sans qu'il soit possible de séparer par une marque quelconque certaines données. Le contenu des octets n'est pas du tout interprété par TCP, c'est donc aux applications d'extrémité de savoir gérer la structure du flot de données.

Si elles sont trop volumineuses, les données à transmettre pour une application sont fractionnées en fragments dont la taille est jugée optimale par TCP. A l'inverse, TCP peut regrouper des données d'une application pour ne former qu'un seul datagramme de taille convenable de manière à ne pas charger inutilement le réseau. Cette unité d'information émise est appelée segment. Certaines applications demandent que les données soient émises immédiatement, même si le tampon n'est pas plein. Pour cela, elles utilisent le principe du "*push*" pour forcer le transfert. Les données sont alors émises avec un bit marquant cela pour que la couche TCP réceptrice du segment remette immédiatement les données à l'application concernée.

La fiabilité fournie par TCP consiste à remettre des datagrammes, sans perte, ni duplication, alors même qu'il utilise IP qui lui, est un protocole de remise non fiable. Ceci est réalisé à l'aide de la technique de l'accusé de réception (ACK).

Chaque segment est remis avec un numéro qui va servir au récepteur pour envoyer un accusé de réception. Ainsi l'émetteur sait si l'information qu'il voulait transmettre est bien parvenue à destination. De plus, à chaque envoi de segment, l'émetteur arme une temporisation qui lui sert de délai d'attente de l'accusé de réception correspondant à ce segment. Lorsque la temporisation expire sans qu'il n'ait reçu de ACK, l'émetteur considère que le segment s'est perdu et il le réexpédie. Mais il se peut que la temporisation expire alors que le segment a été transmis sans problème, par exemple suite à un engorgement de réseau ou à une perte de l'accusé de réception correspondant. Dans ce cas, l'émetteur réémet un segment alors que c'est inutile. Mais le récepteur garde trace des numéros de segments reçus, donc il est apte à faire la distinction et peut éliminer les doublons.

6.3 Exemple : la télé-échographie

6.3.1 L'architecture réseau

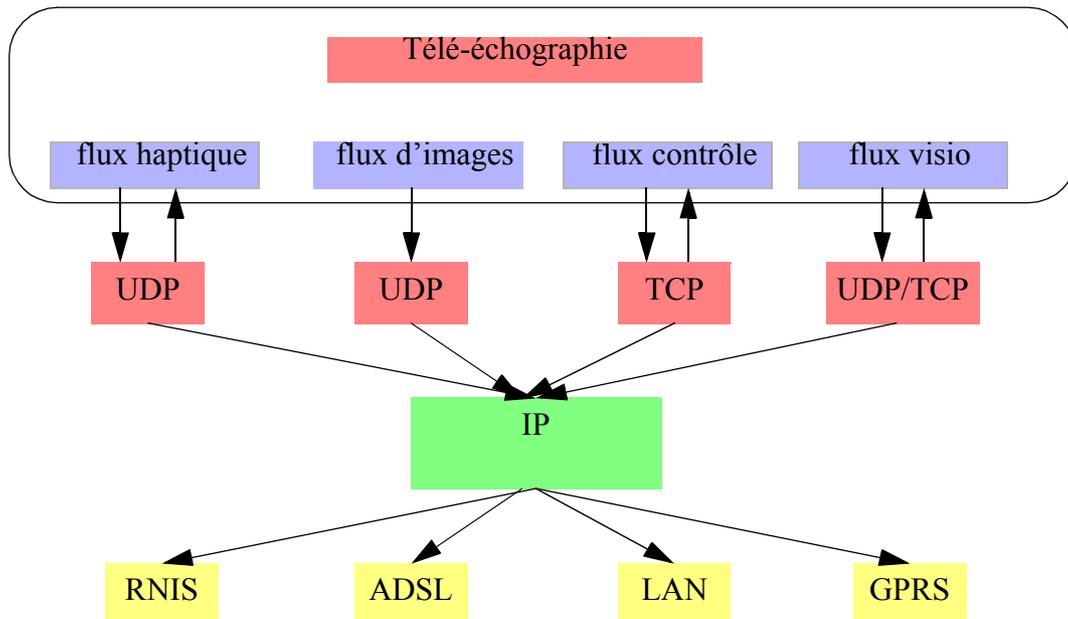


figure 6.2 : Architecture réseau pour le poste de télé-échographie (poste serveur car le flux d'images est sortant).

La télé-échographie a été testée sur deux types de support, le RNIS et le LAN (Local Area Network). On peut voir (Figure 6.2, page 103) que les protocoles TCP et UDP sont utilisés selon le flux de données à transmettre. Ainsi on utilise le protocole UDP pour les flux de données servant à alimenter la boucle haptique, mais également pour le retour visuel, ce protocole est en effet préférable au TCP pour les applications *"temps réel"*. Par contre, pour le flux de contrôle des services ainsi que pour une part de la visio on utilise le protocole TCP. Le RNIS est préférable pour l'instant à l'ADSL non seulement pour son débit assuré et constant mais également pour sa synchronisation. Malgré un fort débit, l'ADSL entraîne des retards dûs aux passages dans les correcteurs d'erreurs, aux routeurs encombrés. On ne peut donc ni assurer un débit constant, ni disposer d'une latence variant peu. Pourtant ceci est nécessaire dans le cas d'une application haptique *temps réel*.

6.3.2 Evaluation du flux haptique

Les données sont encapsulées dans les messages sur le réseau. En général, les messages contiennent : les positions X, Y, Z, les angles α, β, γ et la date d'envoi ou estampille. La taille des messages dépend de la zone de travail qui doit être encodée. La zone de travail est de 160x130x130 (en mm) et la précision est de 0.02 mm, pour un appareil de bureau de type Sensable Phantom.

Le nombre de points à encoder est donné par la formule suivante :

$$\text{nombre de points} = \frac{\text{ambitus maximale}}{\text{precision}} \quad (\text{EQ 1})$$

En appliquant la formule (EQ 1), on obtient le nombre maximum de points à encoder, il est égal à 8000. Les positions peuvent être codées sur 13 bits comme $2^{12}=4096$ et $2^{13}=8192$.

La période d'échantillonnage des messages est un des paramètres qui règle l'expérimentation. Le débit du réseau et les limites de taille des messages sont directement corrélés. Par exemple, le débit sur une ligne RNIS est de 64 Kbits par seconde. Si la période d'échantillonnage est de 10 ms, la taille maximale d'un message est alors de 640 bits, et proportionnellement si la période d'échantillonnage est la milliseconde, on aura des messages d'une taille de 64 bits, soit huit octets.

6.4 Problèmes rencontrés dans la transmission du geste

Il existe dans la littérature un grand nombre de problèmes et de solutions appropriées dans les transmissions réseaux. Notre but n'est pas de tous les présenter mais de proposer une liste non exhaustive des problèmes les plus fréquemment rencontrés. L'ordre de présentation n'est nullement lié à l'importance des problèmes ou à leur fréquence d'apparition.

• Les pertes de données

C'est un problème que l'on rencontre dans la transmission sur IP. La solution réside en deux phases, il faut en premier lieu détecter ces pertes, et dans un deuxième temps il faut interpoler le ou les morceaux du signal qui ont été perdus.

• Les paquets désordonnés

La détection se fait simplement car les messages sont estampillés. Il faut par contre définir un critère permettant d'évaluer le désordre. Dans le cas d'un désordre faible, on pourra créer une boucle de tri à l'arrivée des messages permettant ainsi de rétablir l'ordre d'envoi des messages.

• Les erreurs de transmission

Ce problème a été couramment traité dans la littérature, il existe des algorithmes de codeurs permettant la détection et la correction de ces erreurs.

• Coupure de liaison

Deux types de causes peuvent provoquer ce genre de problèmes, ce peut être une défaillance du "soft" ou une déficience du "hard". Il faudra donc arrêter le système, le recalibrer puis le relancer.

• Délais trop important

Il faut pouvoir détecter que le délai de réception de ce message est trop important. Pour cela, il faut déterminer un seuil au delà duquel, on considère le message comme perdu.

- Limite de la fréquence d'échantillonnage dans l'envoi des messages

Il est possible que la fréquence d'envoi des messages soit inférieure à la fréquence de la boucle haptique, ceci aura pour effet d'accroître le délai réseau.

- Délai variable

La durée de la traversée des différentes couches réseaux est une donnée qui varie avec les différents types de réseaux. De plus, sur un même trajet, en transportant les mêmes données, il existe toujours une différence de durée de trajet. On appelle ce phénomène : la gigue. Dans notre cas, les prédicteurs proposés sont directement liés au délai, puisqu'ils en sont fonction. La résolution de ce problème passe par la synchronisation des deux machines (émetteur - récepteur) à la milliseconde. En effet on transporte des messages qui contiennent leur propre date d'envoi, sur la machine distante il leur sera attribué une date d'arrivée. Il suffit donc de faire la soustraction de ces deux dates, afin d'évaluer le délai qui sera utilisé dans la formule de prédiction.

6.5 Conclusion

Pour clore ce chapitre, nous pouvons dire que les applications haptiques *temps réel* nécessitent des réseaux non seulement rapides mais présentant également des caractéristiques de synchronisation et de sûreté. Pour le flux de données haptiques, il est impératif d'utiliser le protocole UDP bien plus rapide que le protocole TCP. De plus la boucle haptique évoluant à une fréquence de 1 KHz, la *couche application* doit définir des "timers" permettant l'envoi et la réception de messages pratiquement à la même cadence. Le support RNIS reste, de part sa synchronisation et sa sûreté, le meilleur dès lors que la distance de connexion entre le poste client et le serveur dépasse le km. Toutes ces expériences menées par Agnès Guerraz et Frank Pellissier au centre FT R&D de Meylan montrent que le geste peut être transporté à l'aide de filtrage ad-hoc, par le réseau Numéris, ou sur IP.

Chapitre 7

Validation théorique du modèle

Introduction. Il existe trois types de signaux de base, qui sont couramment utilisés comme signal de test dans le cadre du traitement du signal : le signal triangulaire, le signal carré et enfin le signal sinusoïdal. Les deux premiers sont des signaux qui ne sont pas continûment dérivables, or nos prédicteurs sont basés sur les dérivées successives par différences finies du signal à prédire.

On préférera pour ces raisons choisir le signal sinusoïdal qui est continûment et indéfiniment dérivable, ce qui signifie que la dérivée première, la dérivée seconde et la dérivée tierce seront des signaux continus.

Nous présentons dans un premier temps, l'outil de comparaison, qui nous permettra de mesurer les performances des différents prédicteurs proposés. Par la suite, nous évaluons successivement les performances des trois prédicteurs proposés (Section 3.2, page 40), en utilisant deux signaux sinusoïdaux ayant des fréquences différentes afin de mettre en évidence la dépendance des performances

des prédicteurs et de la fréquence maximale contenue dans le signal que l'on désire prédire (Section 5.7, page 87).

7.1 L'outil de comparaison

Afin de mesurer les performances des différents prédicteurs, nous avons besoin d'un critère de comparaison. Nous avons choisi d'utiliser l'un des critères le plus souvent utilisé : le critère d'erreur quadratique moyenne.

L'erreur quadratique moyenne est calculée de la façon suivante :

(EQ 1)

$$Eqm = \frac{\sum_{i=1}^N (\hat{s}(i) - s(i))^2}{N}$$

$\hat{s}(n)$ correspond au signal prédit.

De plus, il est nécessaire de disposer d'un étalon qui permettra d'évaluer l'apport du prédicteur en terme de minimisation de l'erreur de prédiction. Cet étalon sera représenté par le signal issu du prédicteur d'ordre 0 (Section 3.2.1, page 40).

Nous utiliserons deux types de sinus : un sinus de fréquence 3 Hz qui correspond à la fréquence maximale (sur chaque composante) contenue dans un geste d'écriture, et un sinus de fréquence 8 Hz correspondant à la fréquence maximale, toujours sur chaque composante, contenue dans un geste quelconque.

7.2 Evaluation des performances du prédicteur d'ordre 1

7.2.1 Le signal sinusoïdal de fréquence 3 Hz

Les résultats du calcul des erreurs en fonction du délai sont présentés dans le tableau ci-dessous :

Eqm	FP0	FP1	FP0/FP1
$\delta=10$ ms	0.017	$2.35 \cdot 10^{-4}$	74
$\delta=30$ ms	0.15	0.014	10
$\delta=80$ ms	0.9	0.58	1.5

On peut noter que l'erreur de prédiction est sensible à l'augmentation du délai, en effet lorsque le délai grandit, l'erreur de prédiction grandit également. On

observe aussi une diminution des performances du prédicteur d'ordre 1 lorsque le délai augmente, le rapport des erreurs quadratiques moyennes entre le prédicteur d'ordre 0 (désigné comme étalon) et le prédicteur d'ordre 1 diminue avec la croissance du délai. On constate (voir tableau ci-dessus) que le prédicteur d'ordre 1 (FP1), pour un délai de 10 ms, divise l'erreur de prédiction par un facteur 74 par rapport à l'erreur de prédiction provoquée par le prédicteur d'ordre 0 (FP0), ce facteur est réduit à 10 pour un délai de 30 ms et enfin il avoisine 1.5 pour un délai de 80 ms.

On note également que le signal prédit par FP1 pour un délai de 10 ms (Figure 7.1, page 109) est très proche en amplitude du signal à prédire. De plus ces deux signaux sont en phase, donc pour un délai de 10 ms, le prédicteur FP1 est un bon prédicteur du sinus 3 Hz. A contrario, pour un délai de 80 ms, non seulement le signal prédit est amplifié par rapport au sinus 3 Hz mais en plus il n'est plus en phase avec celui-ci. Pour ce délai le prédicteur FP1 ne présente pas de bonnes performances. Ces dernières sont également mitigées pour un délai de 30 ms.

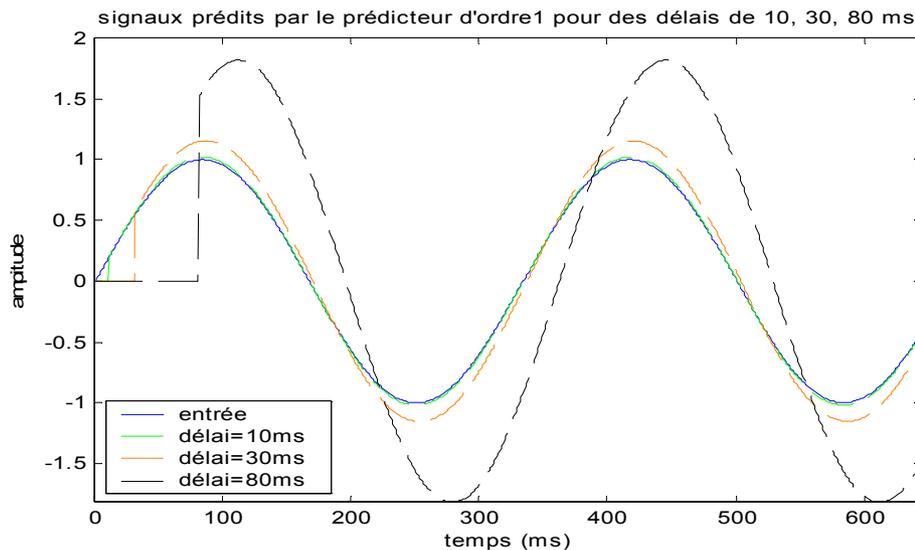


figure 7.1 : signal initial (sinus de fréquence 3 Hz) et ses prédictions par le prédicteur FP1 pour des délais de 10, 30 et 80 ms.

7.2.2 Le signal sinusoïdal de fréquence 8 Hz

Les résultats du calcul des erreurs en fonction du délai sont présentés dans le tableau ci-dessous :

Eqm	FP0	FP1	FP0/FP1
$\delta=10$ ms	0.12	0.0098	12
$\delta=30$ ms	0.92	0.607	1.5

Eqm	FP0	FP1	FP0/FP1
$\delta=80$ ms	1.61	12.64	0.12

De même que dans le cas précédent, on observe une augmentation de l'erreur liée à l'augmentation du délai. Ici aussi, les performances du prédicteur FP1 diminuent lorsque le délai augmente. Pour ce signal sinus 8 Hz, on s'aperçoit que les rapports des Eqm entre le prédicteur d'ordre 0 (FP0) et le prédicteur d'ordre 1 (FP1), sont respectivement bien inférieurs aux rapports des Eqm calculés pour le sinus 3 Hz. Ceci vient du fait que les performances des prédicteurs sont non seulement liées au délai mais aussi à la fréquence maximale contenue dans chaque composante du signal (Section 5.8, page 92) si ce dernier n'est pas un signal à une dimension.

Le prédicteur d'ordre 1 (FP1) nous donne des résultats acceptables pour la prédiction du sinus 8 Hz quand le délai réseau ne dépasse pas les 10 ms (Figure 7.2, page 110). On peut en effet voir que pour un délai de 30 ms, le signal prédit est non seulement très amplifié (pratiquement un facteur 2) mais il est également décalé dans le temps par rapport au sinus 8 Hz. Pour un délai de 80 ms, le phénomène s'amplifie, le signal prédit est amplifié par un facteur 4, de plus il est en opposition de phase avec le signal à prédire. Pour le sinus de fréquence 8 Hz, le prédicteur FP1 donne un résultat acceptable pour des délais réseaux inférieurs à 10 ms, mais pour des délais supérieurs il n'est pas intéressant de retenir le prédicteur d'ordre 1.

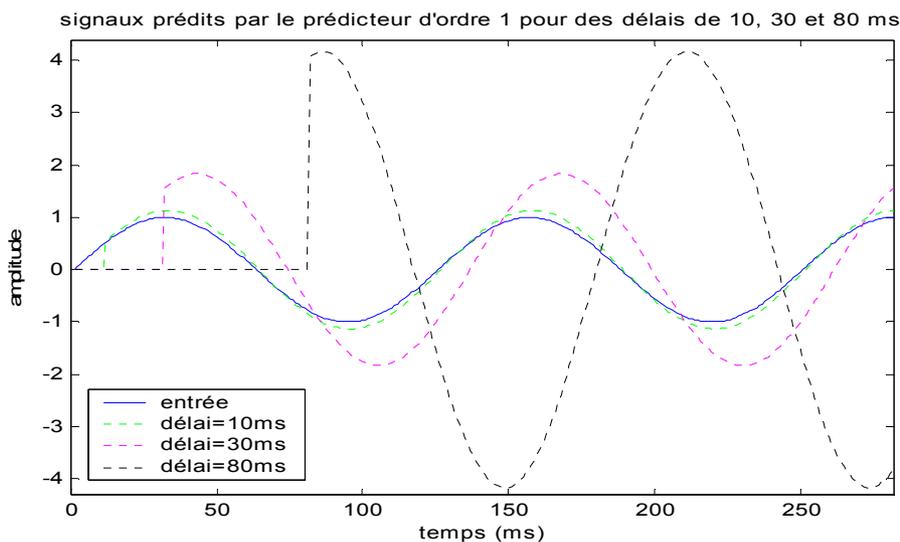


figure 7.2 : signal initial (sinus de fréquence 8 Hz) et ses prédictions par le prédicteur FP1 pour des délais de 10, 30 et 80 ms.

7.3 Evaluation des performances du prédicteur d'ordre 2

7.3.1 Le signal sinusoïdal de fréquence 3 Hz

Les résultats du calcul des erreurs en fonction du délai sont présentés dans le tableau ci-dessous :

Eqm	FP0	FP2	FP0/FP2
$\delta=10$ ms	0.017	$4.77 \cdot 10^{-5}$	369
$\delta=30$ ms	0.15	0.0015	97
$\delta=80$ ms	0.9	0.165	5.5

On constate que le prédicteur d'ordre 2 (FP2), pour un délai de 10 ms, divise l'erreur de prédiction par un facteur 369 par rapport à l'erreur de prédiction provoquée par le prédicteur d'ordre 0, ce facteur passe à 97 pour un délai de 30 ms et il finit à 5.5 pour un délai de 80 ms. En comparant ces résultats à ceux obtenus avec le filtre FP1 dans les mêmes conditions, on s'aperçoit que le filtre FP2 possède de bien meilleures performances que le filtre FP1.

On observe également que les signaux prédits par le filtre FP2 pour un délai de 10 ms et un délai de 30 ms (Figure 7.3, page 111) sont très proches en amplitude du signal à prédire (sinus 3 Hz). De plus, ils ne sont pratiquement pas décalés en temps avec le sinus 3Hz. Le filtre FP2 est donc un très bon prédicteur pour les signaux continus ne possédant pas de fréquences supérieures à 3 Hz, pour un délai s'élevant jusqu'à 30 ms.

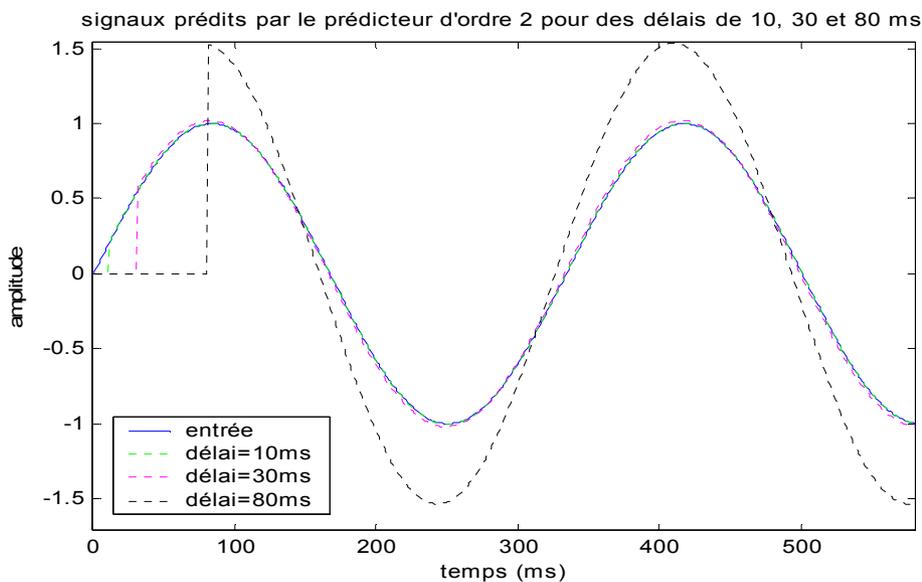


figure 7.3 : signal initial (sinus de fréquence 3 Hz) et ses prédictions par le prédicteur FP2 pour des délais de 10, 30 et 80 ms.

7.3.2 Le signal sinusoïdal de fréquence 8 Hz

Les résultats du calcul des erreurs en fonction du délai sont présentés dans le tableau ci-dessous :

Eqm	FP0	FP2	FP0/FP2
$\delta=10$ ms	0.12	$7.21 \cdot 10^{-4}$	166
$\delta=30$ ms	0.92	0.17	5.41
$\delta=80$ ms	1.61	32.56	0.05

De même que dans les cas précédents, on observe une augmentation de l'erreur liée à l'augmentation du délai. Pour le sinus 8 Hz, on note que les rapports des Eqm entre le prédicteur d'ordre 0 (FP0) et le prédicteur d'ordre 2 (FP2), sont respectivement bien inférieurs aux rapports des Eqm calculés pour le sinus de fréquence 3 Hz. Ce phénomène a également été observé pour le prédicteur d'ordre 1.

Les performances du prédicteur FP2 sont tout à fait acceptables dans la prédiction du sinus 8 Hz tant que le délai ne dépasse pas les 10 ms (Figure 7.4, page 112). Le signal prédit par le filtre FP2 pour un délai de 30 ms est légèrement amplifié par rapport au sinus à prédire, il est également très légèrement retardé. La prédiction pour un délai de 80 ms est non seulement beaucoup trop amplifiée (facteur 8) mais elle est en plus déphasée. Ces résultats montrent que le prédicteur FP2 pour des signaux possédant des fréquences allant jusqu'à 8 Hz fonctionnera pour des délais réseaux compris entre 0 et 20 ms.

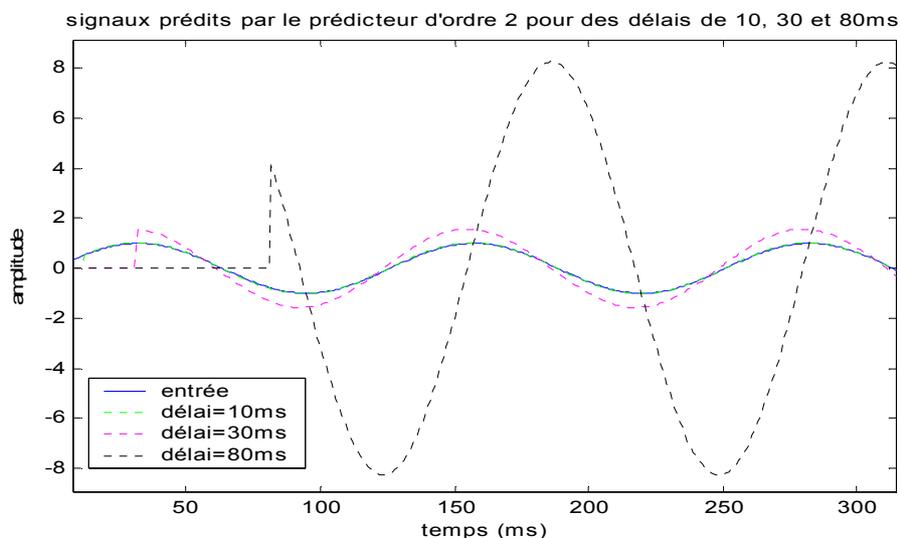


figure 7.4 : sinus initial (sinus de fréquence 8 Hz) et ses prédictions par le prédicteur FP2 pour des délais de 10, 30 et 80 ms.

7.4 Evaluation des performances du prédicteur d'ordre 3

7.4.1 Le signal sinusoïdal de fréquence 3 Hz

Les résultats du calcul des erreurs en fonction du délai sont présentés dans le tableau ci-dessous :

Eqm	FP0	FP3	FP0/FP3
$\delta=10$ ms	0.017	$4.67 \cdot 10^{-5}$	377
$\delta=30$ ms	0.15	0.010	146
$\delta=80$ ms	0.9	0.033	27

On peut noter que le prédicteur d'ordre 3 (FP3), pour un délai de 10 ms, divise l'erreur de prédiction par un facteur 377 par rapport à l'erreur de prédiction produite par le prédicteur d'ordre 0, ce facteur passe à 146 pour un délai de 30 ms et il finit à 27 pour un délai de 80 ms. En comparant ces résultats aux résultats obtenus par le filtre FP2, on s'aperçoit que les performances du prédicteur d'ordre 3 sont similaires aux performances du prédicteur FP2, pour des délais inférieurs à 10 ms, mais le prédicteur d'ordre 3 affiche de meilleurs résultats en terme de minimisation de l'erreur pour des délais de l'ordre de 30 ms et supérieurs.

On constate également que les signaux prédits par le filtre FP3 pour un délai de 10 ms et un délai de 30 ms (Figure 7.5, page 114) se confondent pratiquement avec le signal à prédire (sinus 3 Hz), ils ont pratiquement la même amplitude, et ils ne paraissent pas décalés en temps avec le sinus 3 Hz. Pour un délai de 80 ms, le signal prédit garde à peu près la même amplitude que le signal à prédire, mais on peut observer un décalage en temps de l'ordre d'une quinzaine de millisecondes. Le filtre FP3 est le meilleur des trois filtres, c'est un excellent prédicteur pour les signaux continus ne possédant pas de fréquences supérieures à 3 Hz, pour des délais s'élevant jusqu'à 60 ms.

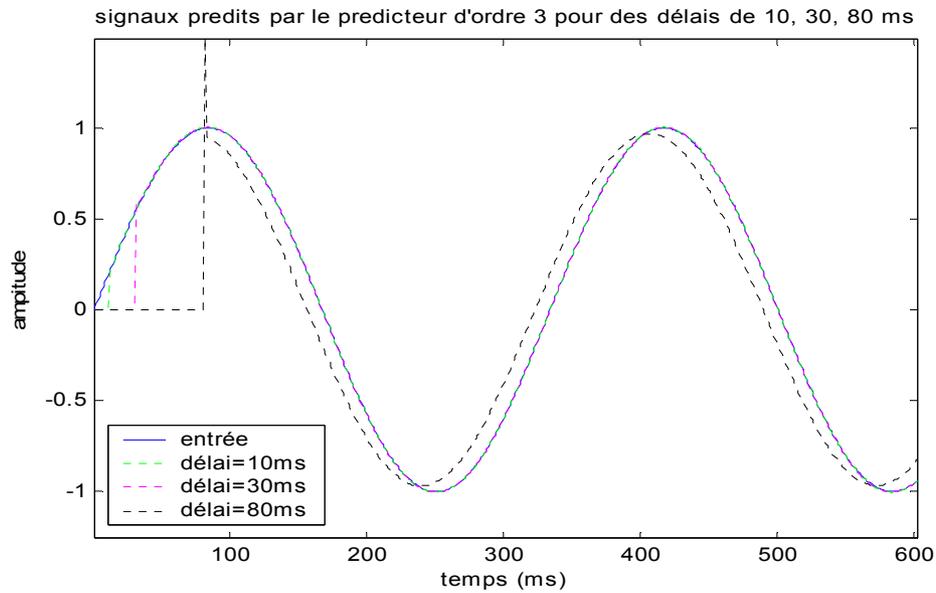


figure 7.5 : signal initial (sinus de fréquence 3Hz) et ses prédictions par le prédicteur FP3 pour des délais de 10, 30 et 80 ms.

7.4.2 Le signal sinusoïdal de fréquence 8 Hz

Les résultats du calcul des erreurs en fonction du délai sont présentés dans le tableau ci-dessous :

Eqm	FP0	FP3	FP0/FP3
$\delta=10$ ms	0.12	$3.61 \cdot 10^{-4}$	332
$\delta=30$ ms	0.92	0.025	36.8
$\delta=80$ ms	1.61	40.36	0.04

Le phénomène de l'augmentation de l'erreur avec le délai est ici à son paroxysme. En effet l'erreur de prédiction est multipliée par un facteur supérieur à 10^5 entre le cas où le délai vaut 10 ms et le cas où le délai vaut 80 ms. Comme pour les autres prédicteurs, on note une augmentation des erreurs de prédiction en passant du sinus 3 Hz au sinus 8 Hz.

Les performances du prédicteur FP3 sont très bonnes dans la prédiction du sinus 8 Hz tant que le délai ne dépasse pas 10 ms (Figure 7.6, page 115). Le signal prédit par le prédicteur FP3 pour un délai de 30 ms est très proche du signal à prédire, ils ont pratiquement la même amplitude, mais le signal prédit est légèrement décalé en temps. Comme pour le filtre FP2, la prédiction du sinus 8 Hz pour un délai de 80 ms est très mauvaise, son amplitude est 10 fois supérieure à l'amplitude du signal à prédire. Ces résultats montrent que le prédicteur d'ordre 3

utilisé sur des signaux continus possédant des fréquences allant jusqu'à 8 Hz fonctionnera pour des délais réseaux compris entre 0 et 30 ms.

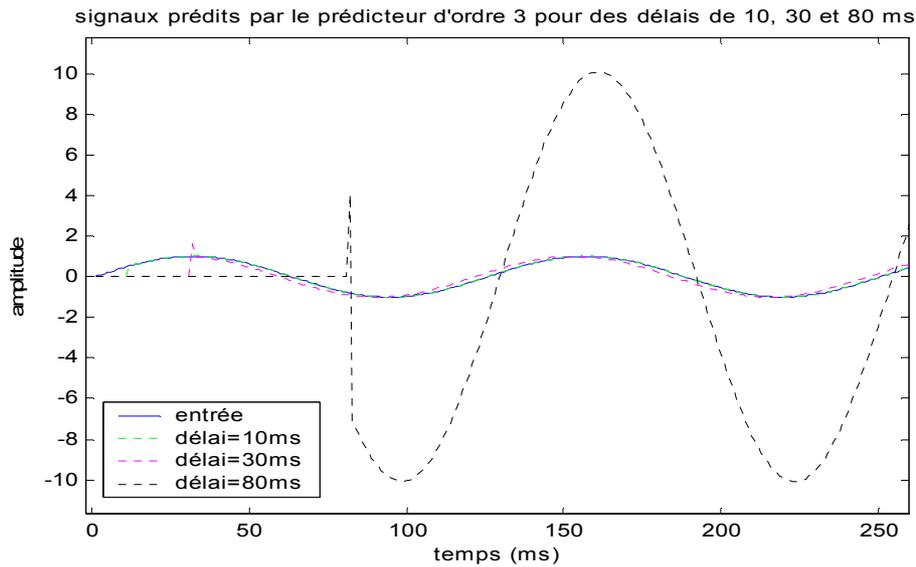


figure 7.6 : signal initial (sinus de fréquence 8 Hz) et ses prédictions par le prédicteur FP3 pour des délais de 10, 30 et 80 ms.

7.5 Conclusion

Ces résultats renforcent les conclusions du chapitre précédent. On observe une forte dépendance des performances des trois prédicteurs vis-à-vis du délai mais aussi de la fréquence maximale contenue dans chaque composante du signal.

Pour des signaux continus, contenant des fréquences inférieures à 3 Hz, les prédicteurs FP2 et FP3 possèdent de très bonnes performances pour un délai de l'ordre de 10 ms. Ces performances sont d'ailleurs meilleures que les performances affichées par le prédicteur FP1 dans les mêmes conditions. Ainsi, on peut effectuer un classement qui s'établit comme suit :

- pour un délai de l'ordre de 30 ms, le prédicteur d'ordre 3 possède des performances supérieures au prédicteur d'ordre 2 qui lui-même est meilleur que le prédicteur FP1.
- pour un délai de l'ordre de 80 ms, le prédicteur d'ordre 3 possède des performances supérieures au prédicteur d'ordre 2 qui lui-même est meilleur que le prédicteur FP1.

Pour des signaux continus, contenant des fréquences allant jusqu'à 8 Hz, le prédicteur d'ordre 3 est meilleur que le prédicteur d'ordre 2, et le prédicteur FP2 possède des performances supérieures au prédicteur FP1. Lorsque le signal que l'on désire prédire possède des fréquences de l'ordre de 8 Hz et que le délai réseau dépasse les 30 ms on atteint les limites des prédicteurs proposés dans cette thèse.

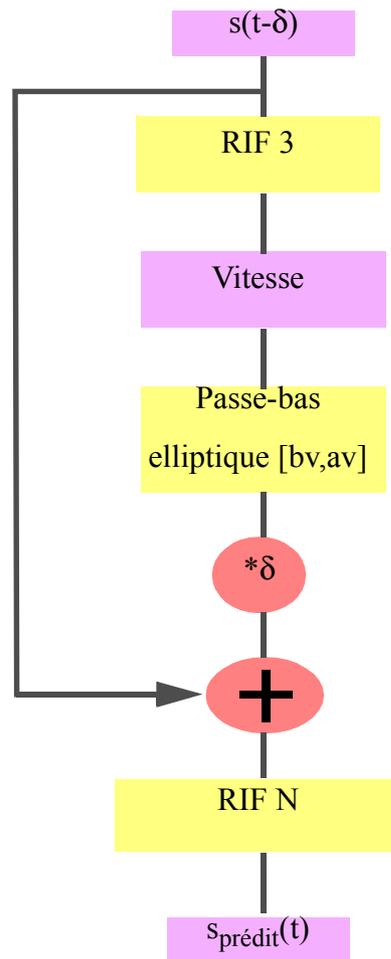
Chapitre 8

Evaluation pratique des modèles proposés

Introduction. Dans le chapitre précédent, nous nous sommes intéressés à un cas d'école, qui consistait à l'évaluation des performances des différents prédicteurs proposés, dans le cas d'un signal sinusoïdal non bruité. Nous allons dans ce chapitre, présenter et évaluer les performances de codeurs prédicteurs ayant pour base les prédicteurs précédemment étudiés, et ce dans le cadre d'un signal gestuel, numérisé par l'intermédiaire du système à retour d'effort Phantom.

Nous présentons dans un premier temps l'architecture des trois codeurs prédicteurs que l'on se propose d'étudier : le codeur prédicteur d'ordre 1 (CP1), le codeur prédicteur d'ordre 2 (CP2) et enfin le codeur prédicteur d'ordre 3 (CP3). Ensuite nous aborderons une discussion sur les choix qui nous ont conduit à préférer ce type d'architecture. Enfin, nous commenterons les résultats expérimentaux de la prédiction d'un geste d'écriture.

8.1 Architecture du codeur prédicteur d'ordre 1



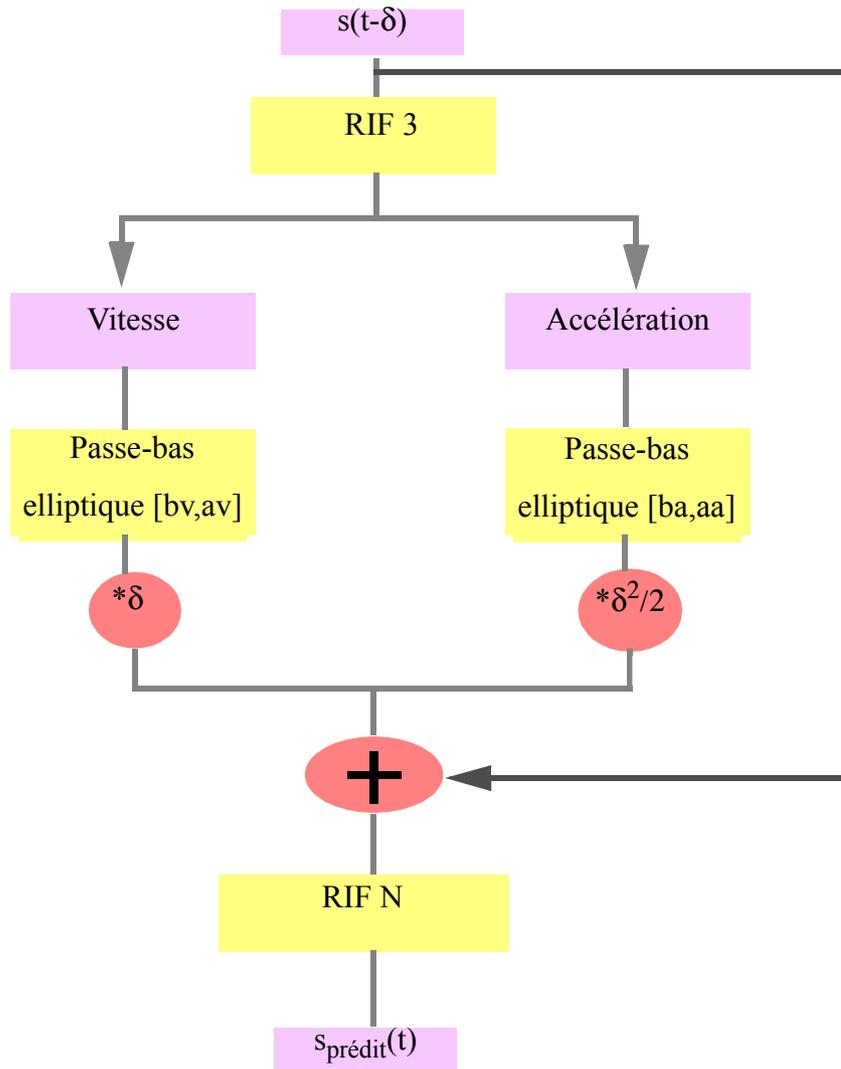
- le signal $s(t-\delta)$ correspond à l'historique du signal dont on dispose pour prédire ce même signal jusqu'à la date t .
- RIF 3 correspond à un filtre lisseur moyennneur représenté mathématiquement par la fonction suivante : $s_{\text{lisse}}(n) = \frac{s(n) + s(n-1) + s(n-2)}{3}$
- Vitesse correspond à la dérivée première du signal, elle est exprimée par la relation suivante :

$$v(n) = s(n) - s(n-1) \quad (\text{Section 3.2.2, page 41}).$$

- Passe-bas elliptique correspond à un filtre à réponse impulsionnelle infinie permettant de débruiter le signal de vitesse. Il est totalement représenté par deux vecteurs de coefficients b_v et a_v (Figure 4.2.3, page 65).
- RIF N est une extension du filtre lisseur moyennneur RIF 3, il est représenté par la fonction suivante :

$$s_{\text{lisse}}(n) = \frac{s(n) + s(n-1) + s(n-2) + \dots + s(n-N+1)}{N}$$

8.2 Architecture du codeur prédicteur d'ordre 2

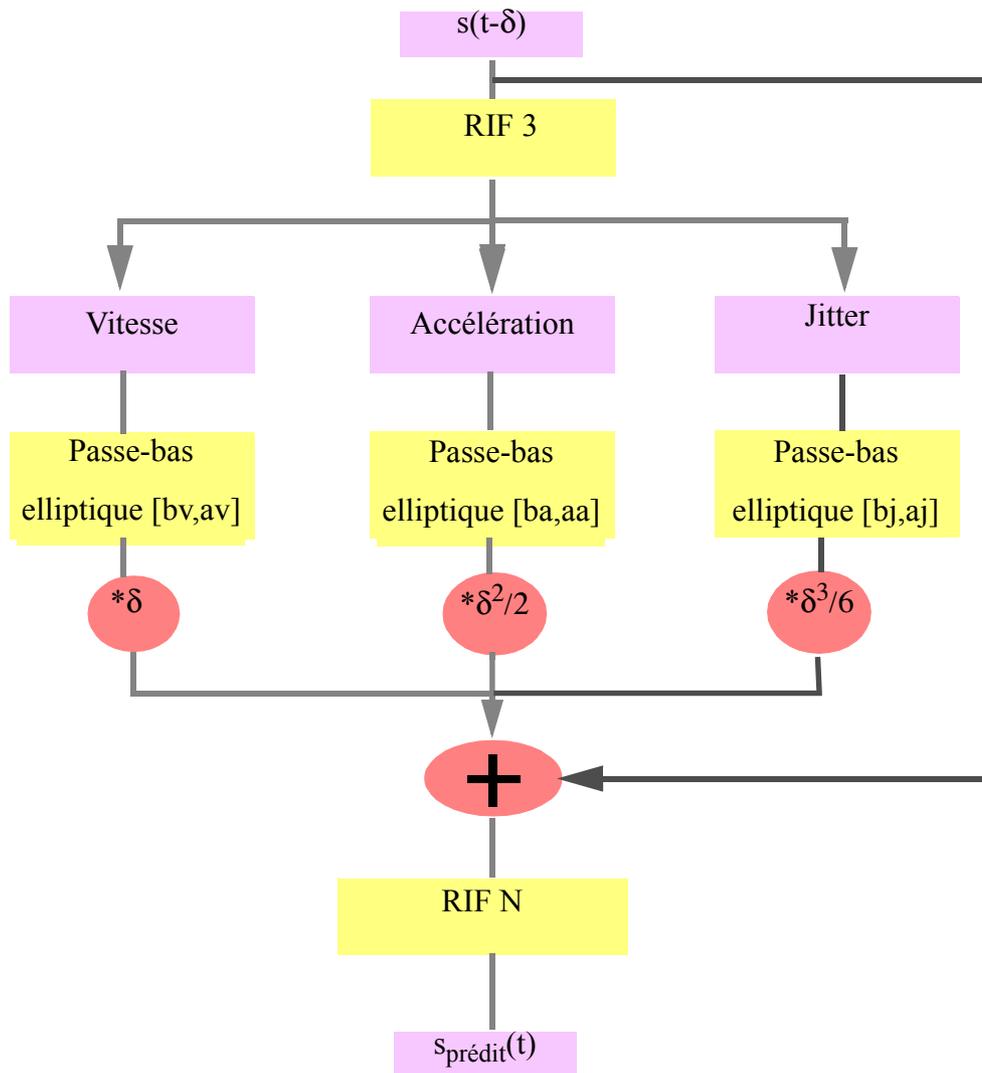


- Accélération correspond à la dérivée seconde du signal, elle est exprimée par la relation suivante :

$$\Gamma(n) = s(n) - 2 \times s(n-1) + s(n-2) \quad (\text{Section 3.2.3, page 45}).$$

- Passe-bas elliptique correspond à un filtre à réponse impulsionnelle infinie permettant de débruiter le signal de vitesse. Il est totalement représenté par deux vecteurs de coefficients ba et aa (Section 4.2.4, page 66).

8.3 Architecture du codeur prédicteur d'ordre 3



- Jitter correspond à la dérivée tierce du signal, il est exprimé par la relation suivante :

$$\Gamma(n) = s(n) - 2 \times s(n-1) + s(n-2)$$
 (Section 3.2.4, page 48).
- Passe-bas elliptique correspond à un filtre à réponse impulsionnelle infinie permettant de débruiter le signal de vitesse. Il est totalement représenté par deux vecteurs de coefficients b_j et a_j (Section 4.2.5, page 68).

8.4 Discussion sur le choix des architectures

8.4.1 L'architecture du codeur prédicteur d'ordre 1 (CP1)

Le premier élément de filtrage de ce codeur est un filtre très simple. C'est un filtre à réponse impulsionnelle finie (Section 4.2.2, page 63), c'est un filtre non récursif car il ne nécessite pas de boucle de réaction dans sa réalisation. Il est calculé sur peu de coefficients (trois en l'occurrence) ce qui induit un retard très léger entre le signal original et sa filtrée. C'est un filtre dont la mise en oeuvre est très simple. Il est essentiellement utile afin d'atténuer les sauts présents dans le signal d'origine. Ces sauts sont très préjudiciables lors de l'évaluation des dérivées successives, par différences finies, du signal d'origine, car ils provoquent des discontinuités importantes. La question se pose donc sur le choix du nombre de coefficients du filtre. Pourquoi avoir utilisé un RIF 3 plutôt qu'un RIF 2 ou qu'un RIF 4. Le RIF 2 ne filtrait pas suffisamment l'amplitude des sauts, tandis que le RIF 4 entraîne un retard trop important de la filtrée par rapport au signal d'origine. Le RIF 3 est un compromis optimal entre le filtrage des discontinuités présentes dans le signal d'origine et le retard qu'il induit.

Le second élément de filtrage est constitué d'un filtre elliptique. Le choix du filtre elliptique comme filtre passe bas afin de lisser les dérivées successives du signal à prédire est expliqué dans le chapitre 4 (Figure 4.2.2, page 63). L'optimisation du filtre elliptique permettant de filtrer le bruit contenu dans la dérivée première du signal original s'est faite à l'aide d'une procédure d'optimisation basée sur la minimisation de l'erreur quadratique :

(EQ 1)

$$E_{qm} = \frac{\sum_{i=1}^N (\hat{s}(i) - s(i))^2}{N}$$

$\hat{s}(n)$ correspond au signal prédit.

Le filtre passe bas elliptique est déterminé par deux vecteurs de coefficients, $a(i)$ et $b(i)$ (voir Matlab Toolbox), eux mêmes sont calculés à partir d'une série de quatre paramètres :

- n correspond à la dimension des vecteurs $a(i)$ et $b(i)$.
- R_p correspond au coefficient d'ondulation de la bande passante.
- R_s correspond au coefficient d'ondulation de la bande atténuée, il est aussi lié à l'atténuation de la bande de coupure.
- W_n correspond à la fréquence de coupure.

Procédure d'optimisation (sous Matlab) permettant de choisir le meilleur ordre de filtre elliptique pour lisser la composante vitesse.

```

%Procédure permettant de choisir le meilleur ordre de filtre elliptique
afin de filtrer la vitesse
%le critère est un critère de minimisation de l'erreur quadratique moyenne
function [xp1,xp2,xp3,xp4,xp5,xp6,E,D]=mon_optimasation_elliptique(x,delt
a)%delta orrespond au délai
xr=the_rifn(x,3); %calcul la filtrée du signal d'entrée x par un filtre
rif d'ordre 3
vit=vitesse(xr); %calcul la dérivée première du signal d'entrée filtré
nommé xr
acc=acceleration(xr); %calcul la dérivée seconde du signal d'entrée
filtrée nommé xr
jitt=jitter(xr); %calcul la dérivée tierce du signal d'entrée filtré
nommé xr

%sur la base [b,a] = ellip(n,Rp,Rs,Wn)

[bv1,av1]=ellip(2,0.02,30,0.08);%calcul les coefficients du filtre ellip-
tique 2 utilisé pour filtrer la vitesse
[bv2,av2]=ellip(3,0.02,25,15/500);%calcul les coefficients du filtre
elliptique 3 utilisé pour filtrer la vitesse
[bv3,av3]=ellip(4,0.02,30,0.08);%calcul les coefficients du filtre ellip-
tique 4 utilisé pour filtrer la vitesse
[bv4,av4]=ellip(5,0.02,30,0.08);%calcul les coefficients du filtre ellip-
tique 5 utilisé pour filtrer la vitesse
[bv5,av5]=ellip(6,0.02,30,0.08);%calcul les coefficients du filtre ellip-
tique 6 utilisé pour filtrer la vitesse
[bv6,av6]=ellip(7,0.02,30,0.08);%calcul les coefficients du filtre ellip-
tique 7 utilisé pour filtrer la vitesse

vx11=filter(bv1,av1,vit);% calcul de la filtrée de la vitesse par le fil-
tre elliptique d'ordre 2
vx12=filter(bv2,av2,vit);% calcul de la filtrée de la vitesse par le fil-
tre elliptique d'ordre 2
vx13=filter(bv3,av3,vit);% calcul de la filtrée de la vitesse par le fil-
tre elliptique d'ordre 2
vx14=filter(bv4,av4,vit);% calcul de la filtrée de la vitesse par le fil-
tre elliptique d'ordre 2
vx15=filter(bv5,av5,vit);% calcul de la filtrée de la vitesse par le fil-
tre elliptique d'ordre 2
vx16=filter(bv6,av6,vit);% calcul de la filtrée de la vitesse par le fil-
tre elliptique d'ordre 2

for i=1:delta, %initialisation des signaux prédits
    xp1(i)=xr(1);
    xp2(i)=xr(1);
    xp3(i)=xr(1);
    xp4(i)=xr(1);
    xp5(i)=xr(1);
    xp6(i)=xr(1);
end

for i=delta+1:(length(x)),

    xp1(i)=xr(i-delta)+vx11(i-delta)*(delta);%formule de prédiction
    xp2(i)=xr(i-delta)+vx12(i-delta)*(delta);
    xp3(i)=xr(i-delta)+vx13(i-delta)*(delta);
    xp4(i)=xr(i-delta)+vx14(i-delta)*(delta);
    xp5(i)=xr(i-delta)+vx15(i-delta)*(delta);
    xp6(i)=xr(i-delta)+vx16(i-delta)*(delta);

end
figure

```

```

plot(x);
hold on;
plot(xp1, 'g');
plot(xp2, 'r');
plot(xp3, 'm');
plot(xp4, 'k');
plot(xp5, 'y');
plot(xp6, 'r:');
[d1]=erq(x,xp1);%calcul l'erreur quadratique moyenne entre le signal x
d'entrée et le signal prédit xp1
[d2]=erq(x,xp2);%calcul l'erreur quadratique moyenne entre le signal x
d'entrée et le signal prédit xp2
[d3]=erq(x,xp3);%calcul l'erreur quadratique moyenne entre le signal x
d'entrée et le signal prédit xp3
[d4]=erq(x,xp4);%calcul l'erreur quadratique moyenne entre le signal x
d'entrée et le signal prédit xp4
[d5]=erq(x,xp5);%calcul l'erreur quadratique moyenne entre le signal x
d'entrée et le signal prédit xp5
[d6]=erq(x,xp6);%calcul l'erreur quadratique moyenne entre le signal x
d'entrée et le signal prédit xp6

D=[d1 d2 d3 d4 d5 d6];
B=[2 3 4 5 6 7];
figure;
plot(B,D);
%fin de la procédure d'optimisation

```

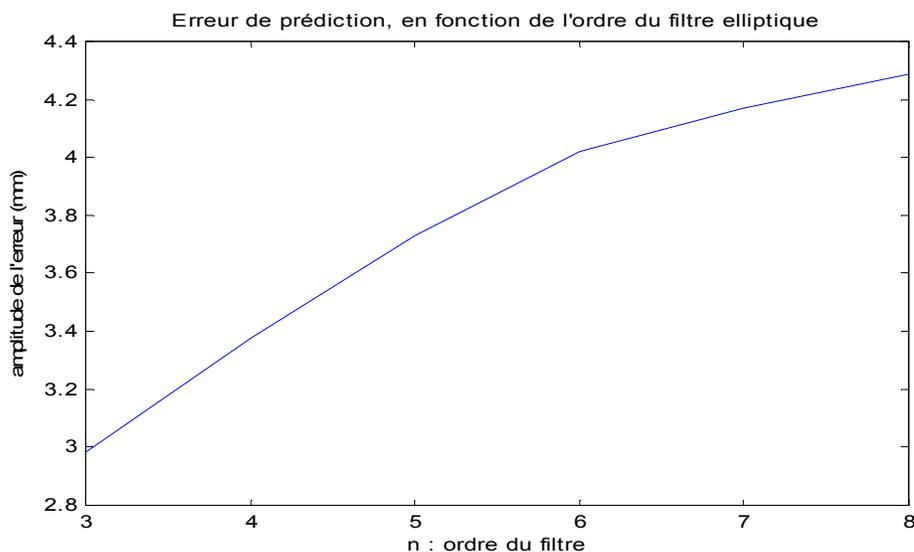


figure 8.1 : Erreur de prédiction en fonction de l'ordre du filtre elliptique utilisé pour filtrer le bruit présent dans la dérivée première du signal original, pour un délai de 70 ms.

On peut voir (Figure 8.1, page 123) que l'erreur de prédiction augmente lorsque l'ordre du filtre elliptique augmente. On serait donc tenté de prendre l'ordre le plus petit. Mais il faut observer non seulement l'erreur de prédiction mais également l'allure du signal prédit, car toute perturbation présente dans le signal prédit sera ressentie comme un choc dans la main de l'utilisateur.

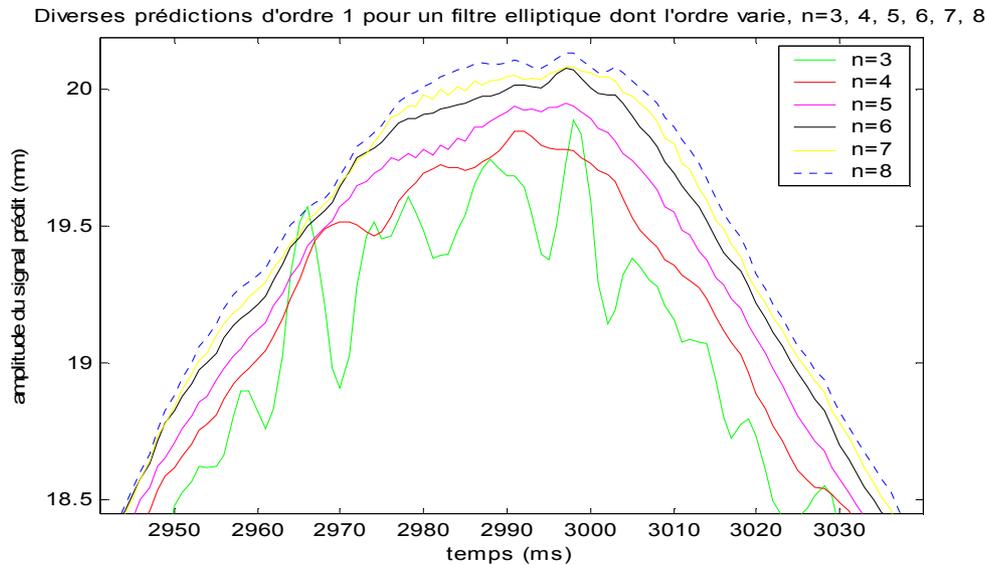


figure 8.2 : Signaux prédits par le codeur prédicteur d'ordre 1 dont l'ordre du filtre elliptique varie par pas de 1, $n=3, 4, 5, 6, 7, 8$.

On observe (Figure 8.2, page 124) que le filtre elliptique d'ordre 3 nous donne une prédiction très chahutée (courbe verte). Selon un critère de lissage, on retient donc le signal filtré par le filtre elliptique d'ordre 6 (courbe noire), car son allure est suffisamment lisse pour le préférer au signal filtré par le filtre elliptique d'ordre 5. Par ailleurs, plus n augmente et plus l'erreur de prédiction augmente, ainsi on préférera le filtre elliptique d'ordre 6 au filtre elliptique d'ordre 7.

On peut donc conclure de part l'expérience que le meilleur filtre elliptique pour le codeur prédicteur d'ordre 1 est celui dont la filtrée apparait lisse (sans discontinuité) et dont l'ordre est le plus bas. Ces caractéristiques ont déjà été présentées dans le chapitre 4 (Section 4.2.3, page 65).

8.4.2 L'architecture du codeur prédicteur d'ordre 2 (CP2)

Le codeur prédicteur d'ordre 2, n'est qu'une amélioration du codeur prédicteur d'ordre 1. En fait on utilise l'information contenue dans la dérivée seconde du signal d'origine. Comme il a été vu dans le chapitre 3 (Section 3.2.3, page 45) la dérivée seconde du signal d'origine est très bruitée. Le choix dans l'architecture du CP2 par rapport à l'architecture du CP1 ne se porte donc que sur les caractéristiques du filtre elliptique retenu pour filtrer le bruit contenu dans la dérivée seconde du signal original.

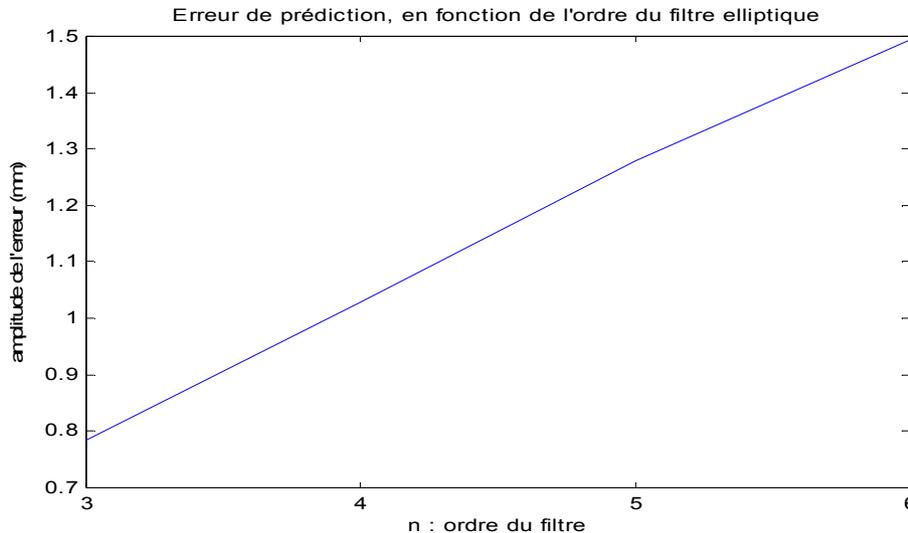


figure 8.3 : Erreur de prédiction en fonction de l'ordre du filtre elliptique utilisé pour filtrer le bruit présent dans la dérivée seconde du signal original, pour un délai de 70 ms.

On peut voir (Figure 8.3, page 125) que l'erreur de prédiction augmente lorsque l'ordre du filtre elliptique augmente. Le filtre elliptique (passe bas) utilisé n'a pas les mêmes caractéristiques que le filtre elliptique utilisé pour filtrer la dérivée première du signal à prédire. Comme précédemment, le critère d'erreur minimale n'est pas suffisant pour choisir le meilleur ordre du filtre elliptique. Il faut également observer l'allure des signaux prédits à partir des différents filtres elliptiques proposés.

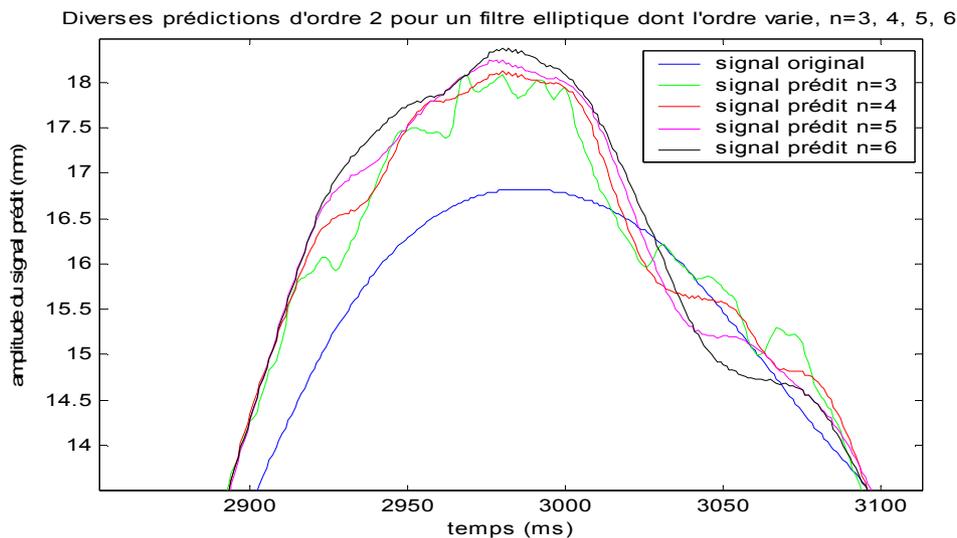


figure 8.4 : Signaux prédits par le codeur prédictif d'ordre 2 dont l'ordre du filtre elliptique varie par pas de 1, n=3, 4, 5, 6.

On observe (Figure 8.4, page 125) que le filtre elliptique d'ordre 3 nous donne une prédiction très chahutée (courbe verte); selon un critère de lissage, on retient

donc le signal filtré par le filtre elliptique d'ordre 4 (courbe rouge), car son allure est suffisamment lisse pour le préférer au signal filtré par le filtre elliptique d'ordre 3. Par ailleurs, plus l'ordre n du filtre elliptique augmente et plus l'erreur de prédiction augmente, aussi on préférera le filtre elliptique d'ordre 4 au filtre elliptique d'ordre 5.

On peut donc conclure de part l'expérience que le meilleur filtre elliptique pour le codeur prédictif d'ordre 2 est celui dont la filtrée apparaît lisse (sans discontinuité) et dont l'ordre est le plus bas. Ces caractéristiques ont déjà été présentées dans le chapitre 4 (Section 4.2.4, page 66).

8.4.3 L'architecture du codeur prédictif d'ordre 3 (CP3)

Comme pour le codeur prédictif d'ordre 2, le codeur prédictif d'ordre 3 (CP3) reprend l'architecture du CP2, en ajoutant une composante issue du filtrage de la dérivée tierce du signal d'origine. Le choix se porte directement sur le filtre elliptique utilisé afin de filtrer le bruit de quantification présent dans cette dérivée tierce.

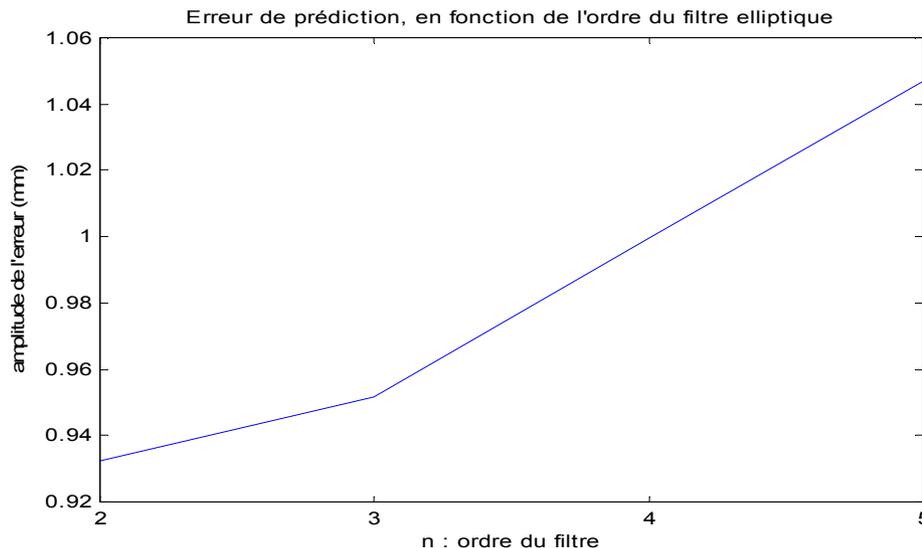


figure 8.5 : Erreur de prédiction en fonction de l'ordre du filtre elliptique utilisé pour filtrer le bruit présent dans la dérivée tierce du signal original, pour un délai de 70 ms.

On peut voir (Figure 8.5, page 126) que l'erreur de prédiction augmente lorsque l'ordre du filtre elliptique augmente. Le filtre elliptique (passe bas) utilisé possède des caractéristiques bien différentes du filtre elliptique utilisé pour filtrer la dérivée première du signal et du filtre elliptique utilisé pour filtrer la dérivée seconde du signal. Comme dans les deux cas précédents, le critère d'erreur minimale n'est pas suffisant pour choisir le meilleur ordre du filtre elliptique. Il faut également observer l'allure des signaux prédits à partir des différents filtres elliptiques proposés.

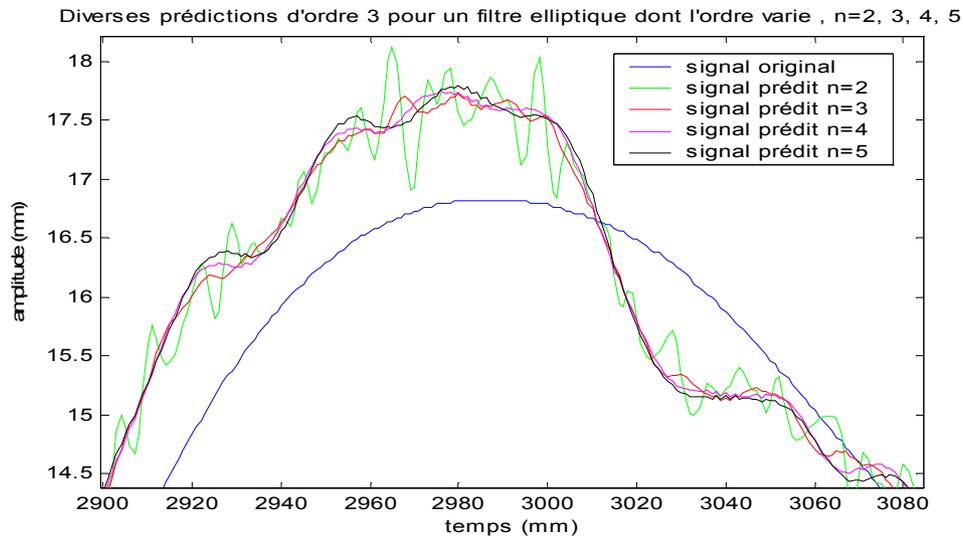


figure 8.6 : Signaux prédits par le codeur prédicteur d'ordre 3 dont l'ordre du filtre elliptique varie par pas de 1, n=2, 3, 4, 5.

On observe (Figure 8.4, page 125) que le filtre elliptique d'ordre 2 nous donne une prédiction très chahutée (courbe verte). Selon un critère de lissage, on retient donc le signal filtré par le filtre elliptique d'ordre 3 (courbe rouge), car son allure est suffisamment lisse pour le préférer au signal filtré par le filtre elliptique d'ordre 2. Par ailleurs, plus l'ordre n du filtre elliptique augmente et plus l'erreur de prédiction augmente, aussi on préférera le filtre elliptique d'ordre 3 au filtre elliptique d'ordre 4.

On peut donc conclure de part l'expérience que le meilleur filtre elliptique pour le codeur prédicteur d'ordre 3 est celui dont la filtrée apparaît lisse (sans discontinuité) et dont l'ordre est le plus bas. Ces caractéristiques ont déjà été présentées dans le chapitre 4 (Section 4.2.5, page 68).

Par ailleurs, nous avons dû pondérer le quatrième terme de l'équation de prédiction, le terme utilisant l'information contenue dans la dérivée tierce du signal à prédire :

$$\hat{s}(n) = s(n - \delta) + \delta \times \frac{d}{dt}s(n - \delta) + \frac{\delta^2}{2!} \times \frac{d^2}{dt^2}s(n - \delta) + \alpha \times \frac{\delta^3}{3!} \times \frac{d^3}{dt^3}s(n - \delta) \text{ (EQ 2)}$$

$$\text{avec } \alpha = \frac{3!}{10 \times \delta} \text{ comme terme de pondération.} \quad \text{(EQ 3)}$$

Ce terme de pondération est un terme qui a été déterminé empiriquement, en utilisant la même procédure d'optimisation que la procédure d'optimisation elliptique utilisée ci-dessus.

8.5 Résultats expérimentaux sur un signal gestuel

Nous allons analyser la composante y (ordonnée) du mot "désir". Le mot a été numérisé par l'intermédiaire d'une application d'écriture haptique, permettant de numériser le geste de la main correspondant à l'écriture d'un mot.

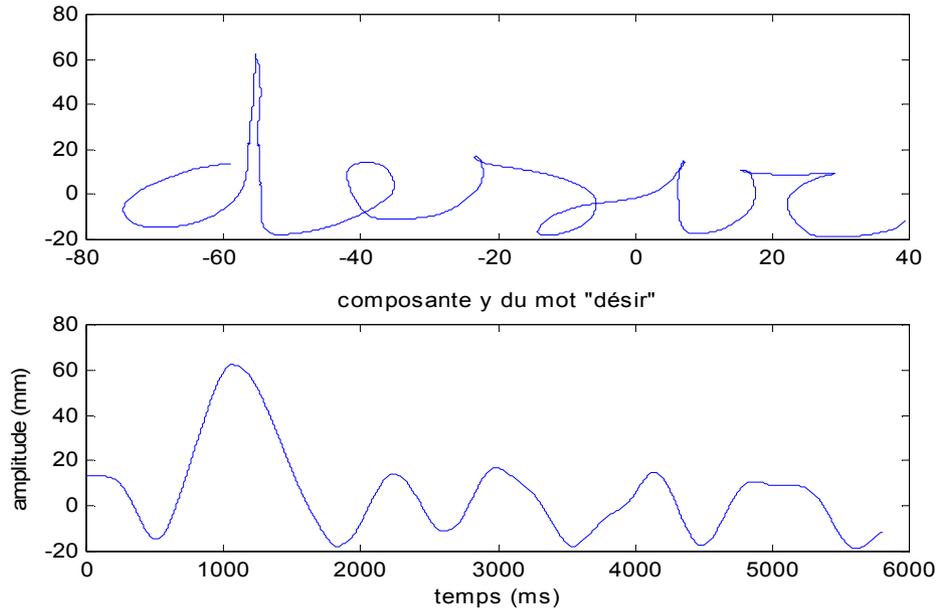


figure 8.7 : en haut le mot "désir" numérisé à l'aide d'une application haptique, en dessous la composante y de ce mot.

8.5.1 Résultats du codeur prédicteur d'ordre 1

Erreur quadratique moyenne en fonction du délai δ .

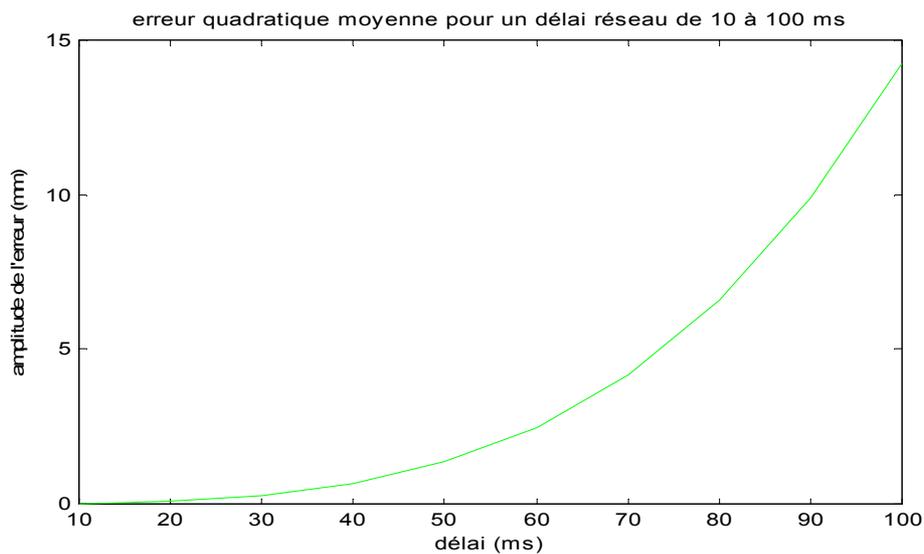


figure 8.8 : Erreur moyenne de prédiction (en mm) effectuée par le codeur prédicteur d'ordre 1, en fonction du délai réseau.

8.5.2 Résultats du codeur prédicteur d'ordre 2

Erreur quadratique moyenne en fonction du délai δ

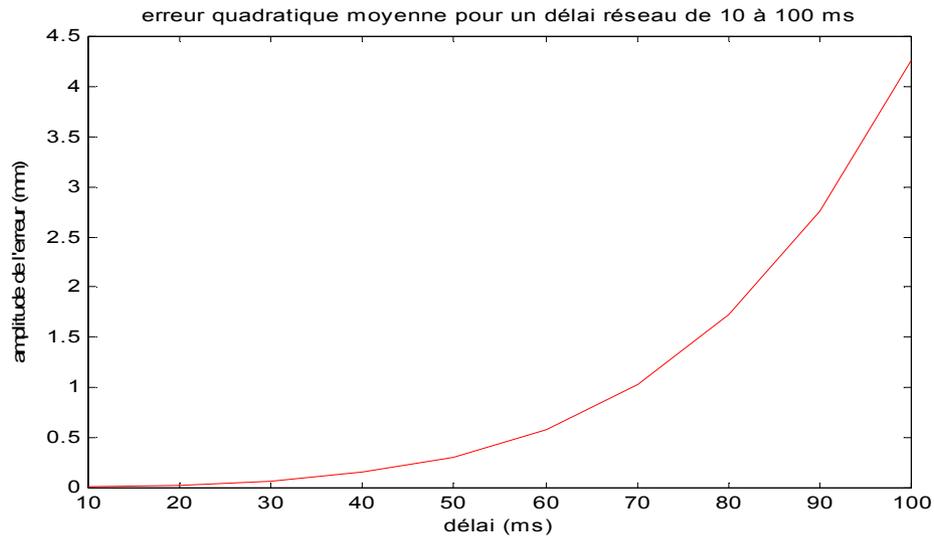


figure 8.9 : Erreur moyenne de prédiction (en mm) effectuée par le codeur prédicteur d'ordre 2, en fonction du délai réseau.

8.5.3 Résultats du codeur prédicteur d'ordre 3

Erreur quadratique moyenne en fonction du délai δ

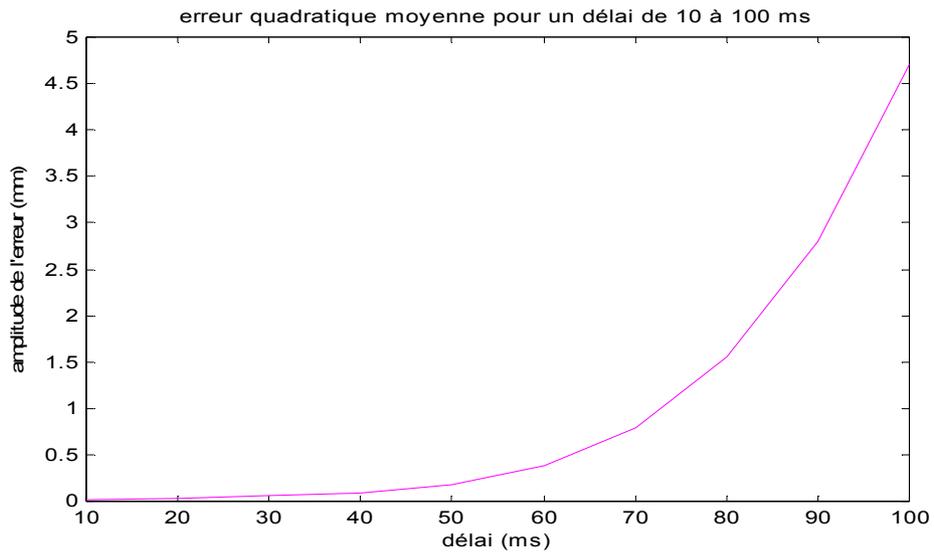


figure 8.10 : Erreur moyenne de prédiction (en mm) effectuée par le codeur prédicteur d'ordre 3, en fonction du délai réseau.

8.5.4 Résultats récapitulatifs

Comparaison pour un délai de 15 ms

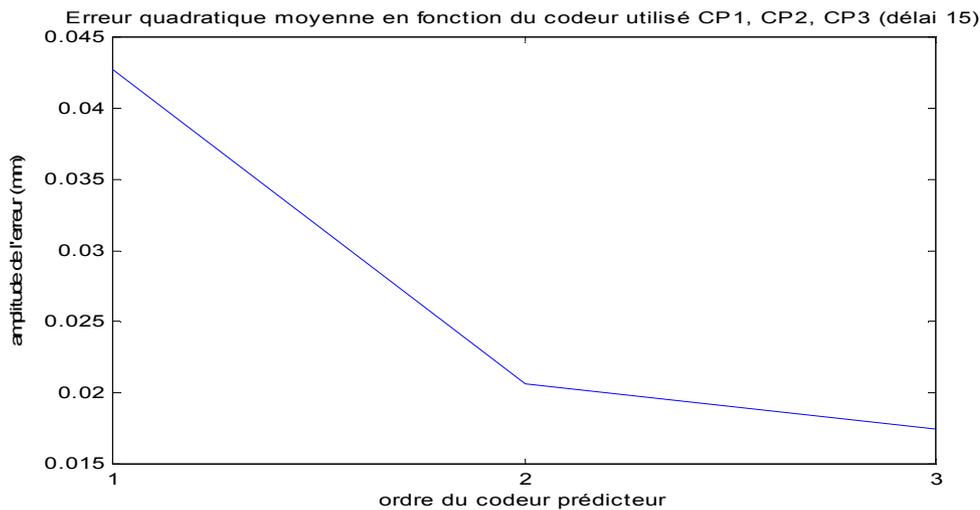


figure 8.11 : Erreur quadratique moyenne en fonction du type de codeur prédicteur utilisé , CP1, CP2, CP3 pour un délai réseau de 15 ms.

L'erreur de prédiction diminue lorsque l'ordre du codeur prédicteur augmente (Figure 8.11, page 130). Donc nous obtenons une meilleure prédiction en utilisant le codeur prédicteur d'ordre 3 (CP3), toutefois l'erreur de prédiction quand on utilise le codeur CP2 est proche de l'erreur de prédiction lorsque l'on utilise le codeur CP3.

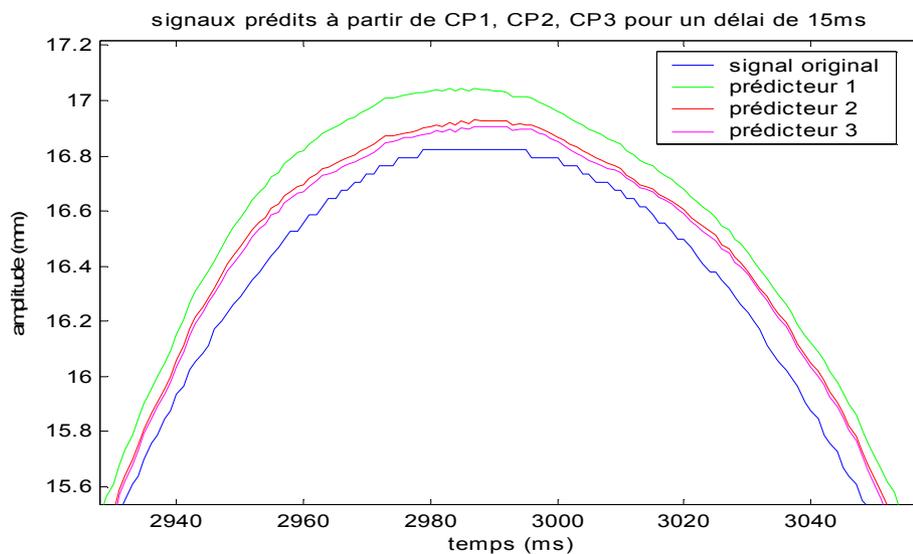


figure 8.12 : Signaux prédits par les différents codeurs prédicteurs CP1, CP2, CP3 pour un délai réseau de 15 ms.

On observe (Figure 8.12, page 130) que le signal prédit le plus proche du signal à prédire est le signal issu du codeur prédicteur d'ordre 3. On voit également que le signal prédit par le codeur CP2 est très proche du signal prédit par le codeur CP3.

En conclusion, on préférera le codeur prédicteur d'ordre 3 à tout autre codeur prédicteur lorsque le délai réseau se situe aux environs des 15 millisecondes.

Comparaison pour un délai de 40 ms

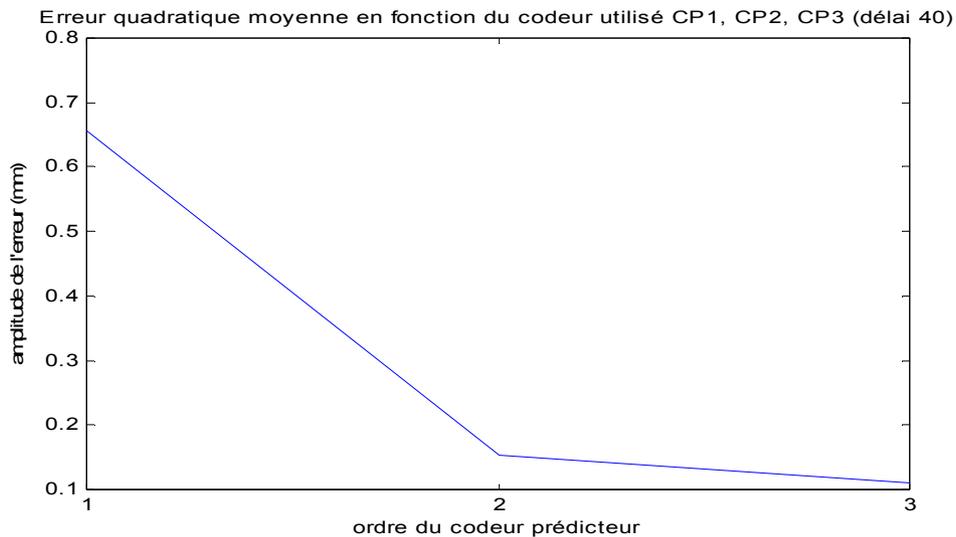


figure 8.13 : Erreur quadratique moyenne en fonction du type de codeur prédicteur utilisé , CP1, CP2, CP3 pour un délai réseau de 40 ms.

Comme dans le cas précédent, l'erreur de prédiction est une fonction décroissante de l'ordre du codeur prédicteur. En effet, on obtient le minimum d'erreur pour un signal prédit par le codeur prédicteur d'ordre 3. La différence entre le codeur prédicteur CP1 et le codeur prédicteur CP3 est assez importante, car l'erreur de prédiction a été divisée par un facteur 5 (Figure 8.13, page 131).

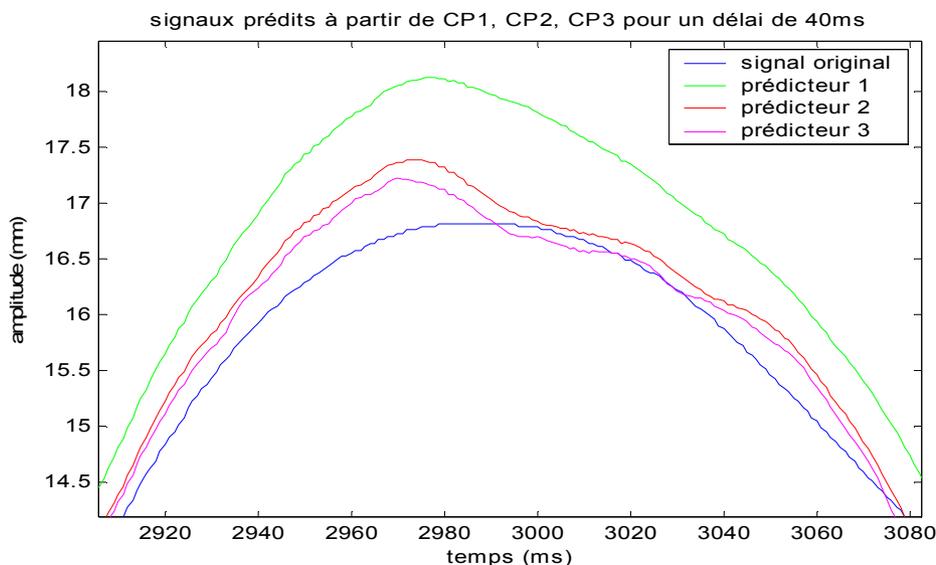


figure 8.14 : Signaux prédits par les différents codeurs prédicteurs CP1, CP2, CP3 pour un délai réseau de 40 ms.

On observe (Figure 8.14, page 132) que le signal prédit par le codeur CP3 (courbe rose) est très proche du signal à prédire (courbe bleue). Le codeur prédicteur d'ordre 2 donne également un résultat acceptable car le signal prédit par CP2 est très proche du signal original qui correspond au signal que l'on veut prédire.

En conclusion, le codeur prédicteur d'ordre 3 reste le meilleur des codeurs proposés, lorsque le délai réseau se situe aux alentours des 40 millisecondes.

Comparaison pour un délai de 70 ms

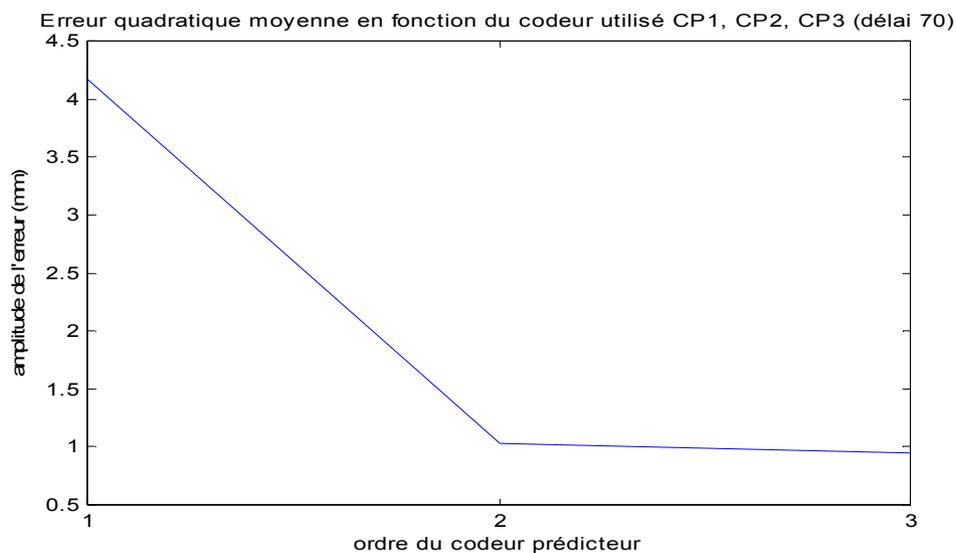


figure 8.15 : Erreur quadratique moyenne en fonction du type de codeur prédicteur utilisé , CP1, CP2, CP3 pour un délai réseau de 70 ms.

Comme dans le cas précédent, l'erreur de prédiction est une fonction décroissante de l'ordre du codeur prédicteur. En effet, on obtient le minimum d'erreur pour un signal prédit par le codeur prédicteur d'ordre 3 (Figure 8.15, page 132). La différence entre le codeur prédicteur CP2 et le codeur prédicteur CP3 est très faible, on constate donc, que plus le délai réseau augmente et plus la différence entre le codeur CP2 et le codeur CP3 diminue. Le codeur prédicteur d'ordre 2 est d'ailleurs " meilleur " lorsque le délai dépasse les 100 ms. Toutefois c'est dans la zone des 100 millisecondes de délai réseau que l'on atteint les limites des codeurs prédicteurs proposés.

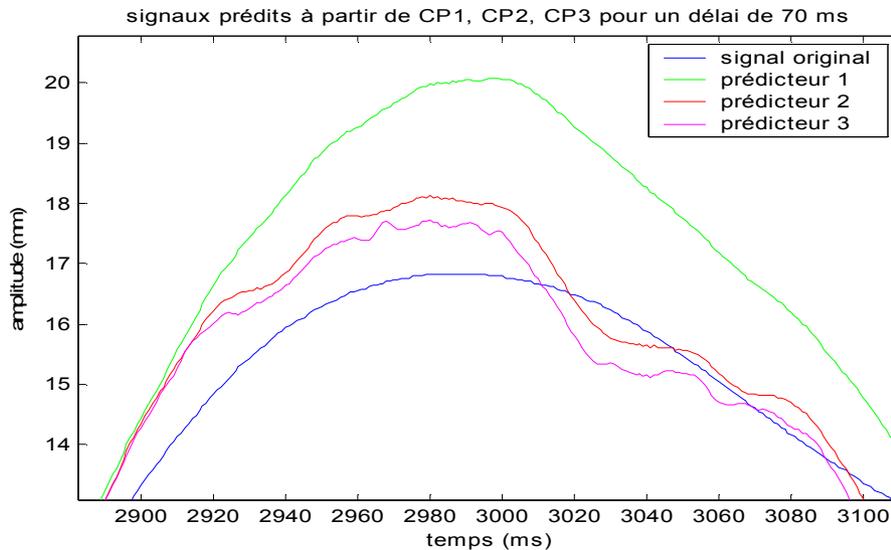


figure 8.16 : Signaux prédits par les différents codeurs prédicteurs CP1, CP2, CP3 pour un délai réseau de 70 ms.

On observe (Figure 8.16, page 133) que le signal prédit par le codeur CP3 (courbe rose) est assez proche du signal à prédire (courbe bleue). Le codeur prédicteur d'ordre 2 donne également un résultat acceptable car le signal prédit par CP2 est assez proche du signal original qui correspond au signal que l'on veut prédire.

En conclusion, le codeur prédicteur d'ordre 3 reste le meilleur des codeurs proposés, lorsque le délai réseau se situe aux alentours des 70 millisecondes.

8.5.5 Tableau récapitulatif :

Erreur Quadratique Moyenne (mm)	Codeur Prédicteur			
	CP0	CP1	CP2	CP3
délai 10 ms	0.9820	0.0201	0.0137	0.0127
délai 20 ms	3.5720	0.0855	0.0314	0.0242
délai 30 ms	7.7622	0.2689	0.0713	0.0500
délai 40 ms	13.5283	0.6556	0.1535	0.1104
délai 50 ms	20.8387	1.3505	0.3091	0.2419
délai 60 ms	29.6548	2.4753	0.5819	0.4981
délai 70 ms	39.9317	4.1684	1.0288	0.9514
délai 80 ms	51.6182	6.5837	1.7235	1.6968

CP0 correspond au signal original retardé du délai réseau, ceci permet d'étalonner les codeurs prédicteurs d'ordre 1, d'ordre 2 et d'ordre 3 (Section 3.2.1, page 40). En fait, le cas CP0 représente la situation dans laquelle on ne fait aucune prédiction, le signal reçu par le poste distant est utilisé sans qu'aucune transformation ne soit réalisée. Il est donc identique au signal transmis par le poste local, il est juste retardé car il a mis un temps certains (correspondant au délai réseau) pour effectuer le trajet "poste local-poste distant".

8.6 Conclusion

Ces résultats sont assez similaires aux résultats obtenus lors de la prédiction d'un signal déterministe non bruité, traitée dans le chapitre précédent. En effet, le prédicteur codeur d'ordre 3 présente de meilleurs résultats en terme de minimisation de l'erreur de prédiction, que les codeurs prédicteurs d'ordre 2 et d'ordre 1. On a pu visualiser l'apport de ces codeurs en comparant les prédictions d'ordre 1, 2 et 3 par rapport à l'étalon représenté par une prédiction d'ordre 0. On s'aperçoit également que les erreurs de prédictions sont très dépendantes du délai, on peut d'ailleurs définir une zone de fonctionnement du codeur prédicteur d'ordre 3, comprise entre 0 et 80 ms environ. Au delà de cette zone, l'erreur de prédiction est trop importante et les codeurs prédicteurs proposés dans cet ouvrage perdent énormément en efficacité. Notre approche du traitement des perturbations induites par le délai dans un télégeste, diffère de l'approche "Wave Transform" (Section 2.4, page 35). En effet, nous n'introduisons aucune contrainte dans le mouvement du manipulateur, ce dernier ne ressent aucun "éther visqueux", il reste libre de ses mouvements et ceci pour des délais allant jusqu'à 80 millisecondes.

Chapitre 9

Application: le circuit automobile

Introduction. Dans le chapitre précédent, nous avons évalué les performances des différents prédicteurs proposés, en prenant comme critère d'évaluation : l'erreur quadratique moyenne. Il est intéressant de mesurer la pertinence qualitative de nos prédicteurs. On désire percevoir de façon tactile l'apport des prédicteurs lors d'un télégeste. Pour ce faire, nous avons élaboré une application haptique de télégeste, permettant de mettre en oeuvre les prédicteurs proposés dans cette étude.

Dans un premier temps nous présentons une description détaillée de l'application nommée "le circuit automobile". Puis nous présentons des résultats quantitatifs, en mesurant la distance qui sépare la position locale de la position prédite et ce en fonction de l'ordre du prédicteur utilisé et du délai réseau. Nous poursuivons par une mesure qualitative en terme de ressenti dans la main des forces de contre-réactions.

9.1 Description de l'application proposée

Tout d'abord, rappelons brièvement le fonctionnement du système à retour d'effort. Le système à retour d'effort mesure des données de position de la main du manipulateur dans un espace appelé "WorkSpace", c'est à dire "espace de travail". Il envoie ces données de position au logiciel permettant d'accéder à l'application qui lui est attribuée, le logiciel calcule alors une force de retour appropriée, c'est à dire forte s'il y a collision avec un objet dans la scène virtuelle, et nulle autrement.

9.1.1 Description graphique

Le circuit est constitué par une trajectoire centrale (ligne blanche), cette trajectoire a été élaborée dans le logiciel de création graphique 3D Studio Max. Puis nous avons suréchantillonné la trajectoire afin d'atteindre un nombre de 6869 points afin d'avoir un écart entre chaque point qui ne soit pas ressenti par la main de l'utilisateur du système à retour d'effort. Le circuit s'étend sur une longueur de 200 mm et une hauteur de 120 mm. Nous avons construit les bords du circuit (lignes violettes) en calculant une largeur de route de 20 mm. Dans cet espace nous avons deux variables, la position locale de la main de l'utilisateur représentée par une bille verte et la position distante prédite de la main du manipulateur représentée par une bille rouge (Figure 9.1, page 136). La ligne de départ est représentée par une droite verte, elle permet de lancer le chrono lorsqu'elle est franchie. Lors des sorties de route, les bords du circuit clignotent ce qui permet au manipulateur d'avoir un retour visuel et de corriger sa trajectoire. De plus, nous avons décidé d'introduire des pénalités temporelles lors des sorties de route, le temps passé en dehors du circuit est multiplié par trois.

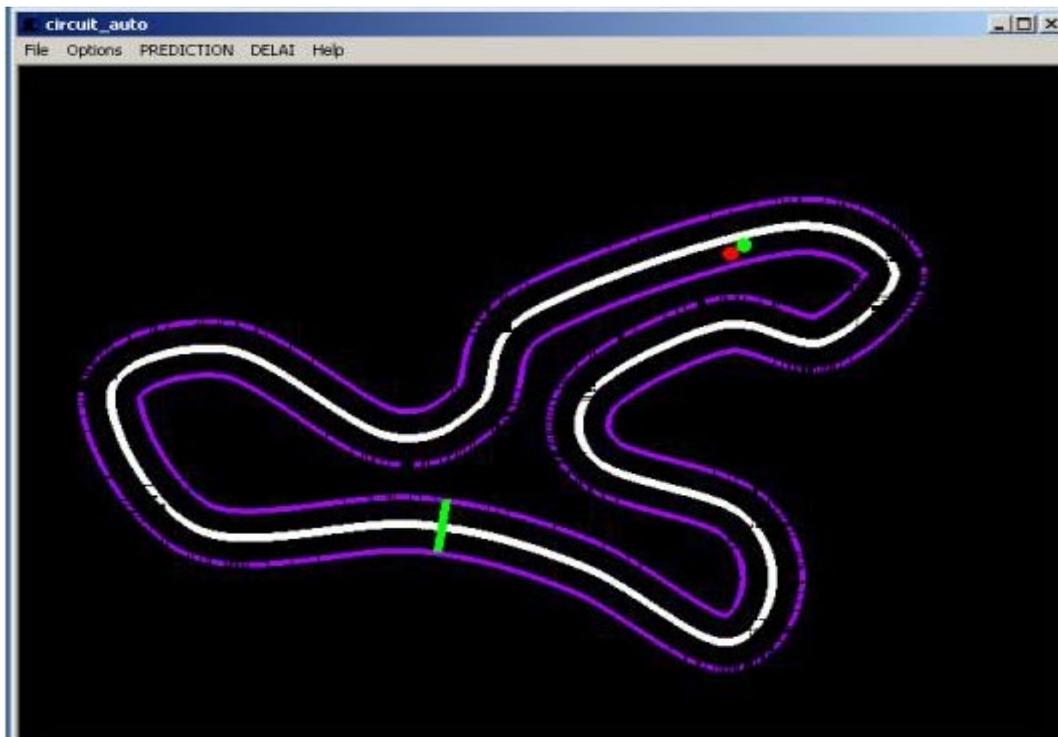


figure 9.1 : Le circuit automobile, la boule verte correspond à la position locale de la main du manipulateur, la boule rouge correspond à la position prédite distante de la main dans la scène distante.

9.1.2 Description haptique

Après la description graphique de l'application, il convient de décrire les forces ressenties par la main du manipulateur du SRE. En premier lieu, il est important de dire que les forces calculées par le logiciel couplé au système à retour d'effort sont évaluées non pas par rapport à la position locale (la bille verte) mais par rapport à la position distante prédite (la bille rouge).

La force totale renvoyée dans la main de l'opérateur est en fait la somme de trois forces.

Une première force de type "ressort" nous astreint à un mouvement plan, comme si l'on écrivait sur un tableau d'école. Le circuit est lui-même dessiné dans ce plan, d'équation $z=0$, on ne peut donc faire que des mouvements longitudinaux et transversaux, mais nous ne pouvons pas exécuter de mouvements dans le sens de la profondeur.

Une seconde force toujours de type "ressort" empêche la main de l'opérateur de s'éloigner de trop du centre de la route. En fait, on calcule en permanence le point sur la trajectoire du centre de la route le plus proche de la position distante prédite, ce calcul est fait dans le modèle distant, puis on calcule une force de rappel au niveau du poste local du type :

$$\text{cursor_force_distant} = k_circuit * (\text{Position_la_plus_proche} - \text{Position_prédite})$$

- le `cursor_force` correspond à la force renvoyée par les moteurs du SRE dans la main du manipulateur.
- `k_circuit` correspond à la constante de raideur du ressort qui relie la `Position_la_plus_proche` et la `Position_prédite` (bille rouge)

Enfin, une dernière force est une force de viscosité. Cette force s'oppose au mouvement comme lorsque l'on déplace une cuillère dans du miel. Le coefficient de viscosité est ici proportionnel à la distance qui sépare la position locale de la main de l'utilisateur (bille verte) et la position prédite distante de la main de l'opérateur (bille rouge). Cette force est exprimée par la relation suivante :

$$\text{cursor_force_distant} = -\text{coefficient_de_viscosité} * \text{vitesse_prédite}$$

La vitesse prédite correspond à la dérivée première par différences finies de la position distante prédite de la main de l'opérateur.

Enfin, il nous faut donc ramener cette information de force vers le poste local et il nous faut faire une prédiction de cette force au niveau du poste local afin de limiter l'effet perturbateur résultant du retard entraîné par le transport de cette information.

On transmet donc au moteur du système à retour d'effort une force locale prédite par les prédicteurs (similaires aux prédicteurs du poste distant) contenus dans le modèle local (Figure 9.2, page 138).

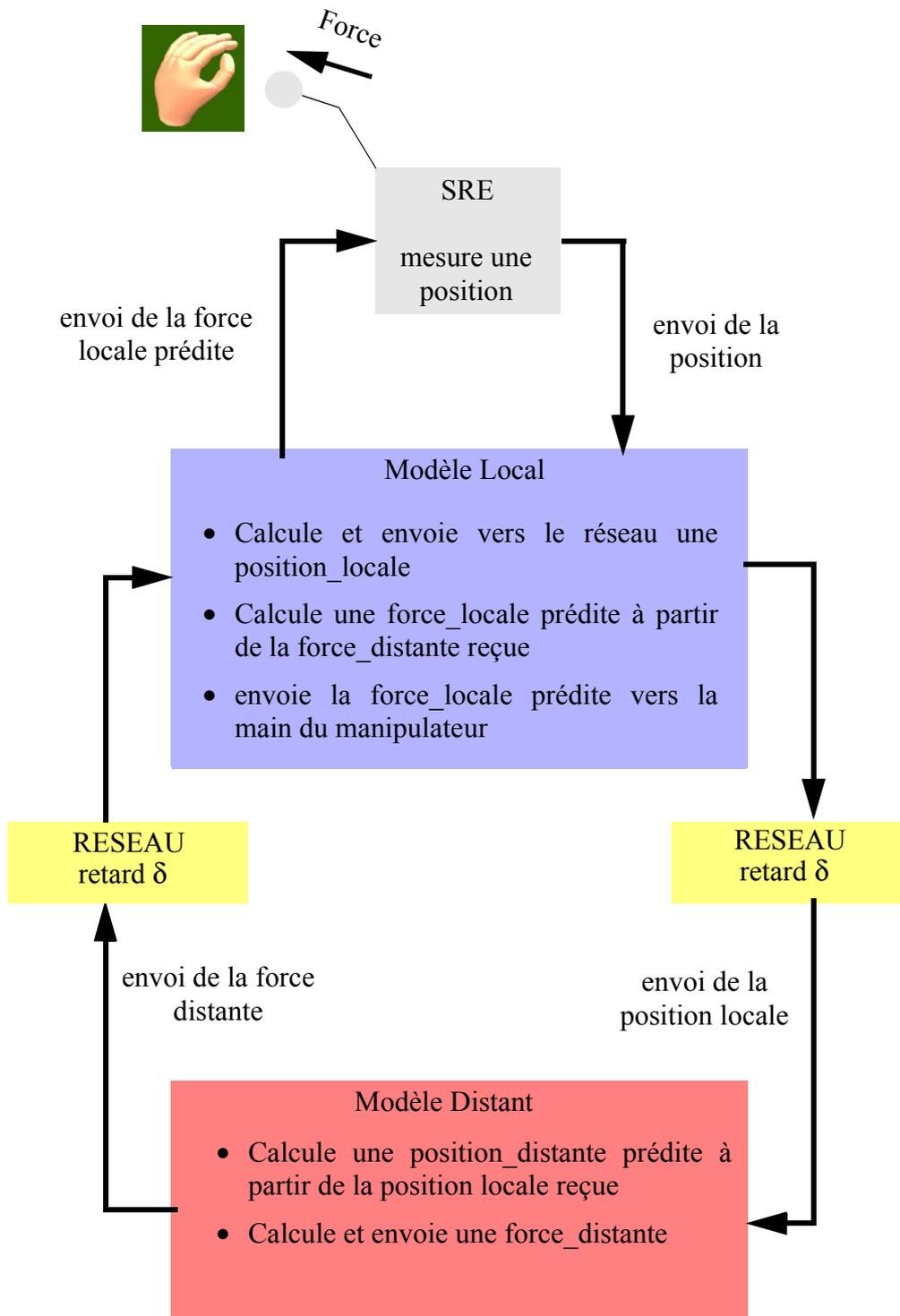


figure 9.2 : synopsis de la boucle haptique pour l'application : "le circuit automobile".

9.1.3 Description conceptuelle

Nous avons construit un premier menu défilant permettant de régler l'ordre du modèle de prédiction que l'on désire utilisé pour calculer la position de la main de

l'opérateur dans la scène distante, nous rappelons que cette position est représentée graphiquement par une bille rouge (Figure 9.3, page 139).

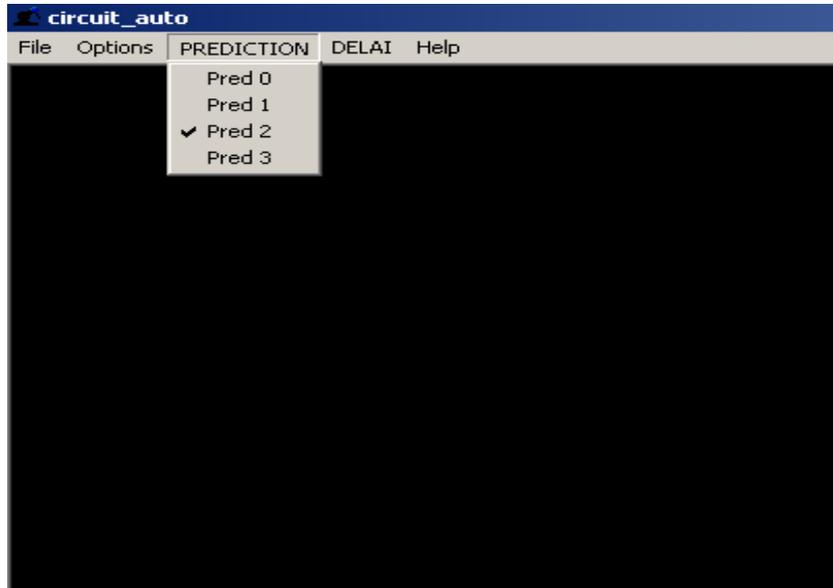


figure 9.3 : Le menu déroulant permettant de régler l'ordre de prédiction.

Un second menu défilant a été programmé, il permet de régler le délai réseau que l'on désire avoir entre la scène locale et la scène distante. Ceci permet d'évaluer la performance des différents prédicteurs pour des délais réseaux variant de 0 à 100 ms, par pas de 10 ms (Figure 9.4, page 139).

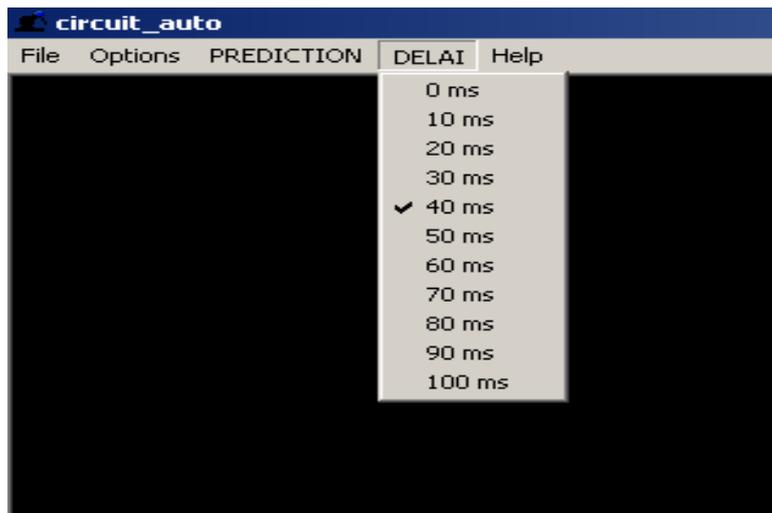


figure 9.4 : Le menu déroulant permettant de régler le délai réseau.

9.2 Mesure quantitative : distance position locale-position prédite

Comme nous l'avons vu dans l'introduction, le problème dans un télégeste résulte de l'instabilité de la boucle haptique (homme-système à retour d'effort-réseau-retour de la force dans la main du manipulateur). Cette instabilité est provoquée par la latence du réseau qui rend impossible une réponse simultanée (inférieure à 1 ms) à la sollicitation du manipulateur, la boucle capteur-effecteur n'est plus stable.

Cependant on peut voir ce problème sous un autre angle, l'instabilité est en fait générée par l'existence d'une distance entre la position du manipulateur dans la scène locale et la position distante de la main du manipulateur dans la scène distante. Lorsque nous n'utilisons pas de prédicteur, la position distante correspond en fait à la position locale de la main du manipulateur mais retardée du délai réseau (Figure 1.3, page 22). Le fait de prédire la position distante de la main du manipulateur, en disposant uniquement de la mémoire de la position locale jusqu'à une date $t-\delta$, nous permet de diminuer de façon importante la distance entre la position locale et la position "prédite" distante, ce qui permet d'améliorer l'ergonomie d'un télégeste.

9.2.1 Distance position locale-position prédite, pour un délai de 10 ms

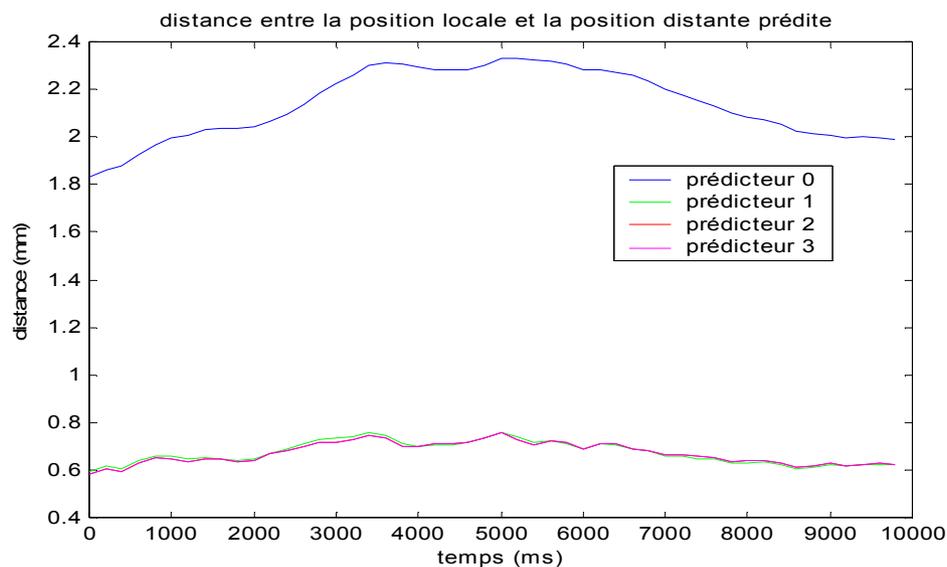


figure 9.5 : distance entre la position locale de la main de l'utilisateur et sa position distante prédite, avec un délai réseau de 10 ms, cette distance étant évaluée pour les prédicteurs d'ordre 0, 1, 2 et 3.

La distance entre la position locale et la position distante prédite a été évaluée pour les trois prédicteurs proposés dans cette étude ainsi que pour la prédiction d'ordre 0 qui nous sert d'étalon, afin d'évaluer l'apport des prédicteurs proposés. Cette distance a été calculée à la milliseconde par l'application : "le circuit automobile". Afin de présenter des résultats plus lisibles, nous avons fait une extraction de 10000 points que nous avons moyenné sur des fenêtres de largeur 200 ms.

Nous pouvons constater sur la Figure 9.5, page 140, que la distance moyenne entre la position locale et la position distante prédite est de l'ordre de 2.2 mm dans le cas du prédicteur d'ordre 0. On s'aperçoit que cette distance est environ de 0.6 mm pour le prédicteur d'ordre 1 (courbe verte). Les prédicteurs d'ordre plus élevé, c'est à dire le prédicteur d'ordre 2 et le prédicteur d'ordre 3 nous donnent une distance entre la position locale et la position distante prédite très proche de celle obtenu par le prédicteur d'ordre 1. Pour des délais inférieurs ou égaux à 10 ms, le prédicteur d'ordre 1 est donc en général suffisant.

9.2.2 Distance position locale-position prédite, pour un délai de 30 ms



figure 9.6 : distance entre la position locale de la main de l'utilisateur et sa position distante prédite, avec un délai réseau de 30 ms, cette distance étant évaluée pour les prédicteurs d'ordre 0, 1, 2 et 3.

De la même façon que dans l'exemple précédent, la distance a été calculée à la milliseconde par l'application : "le circuit automobile". Nous avons également extrait une portion de 10000 points que nous avons moyenné sur des fenêtres de largeur 200 ms.

Nous pouvons constater sur la Figure 9.6, page 141, que la distance moyenne entre la position locale et la position distante prédite est de l'ordre de 5 mm dans le cas du prédicteur d'ordre 0. Cette distance a par conséquent plus que doublé, alors que le délai réseau n'augmentait pour sa part que de 20 ms. On s'aperçoit que cette distance est environ de 0.8 mm pour le prédicteur d'ordre 1 (courbe verte). Le prédicteur d'ordre 2 et le prédicteur d'ordre 3 nous donnent des distances entre la position locale et la position distante prédite très proches l'une de l'autre, de l'ordre de 0.6 mm. Pour des délais compris entre 10 et 30 ms, il est préférable d'employer le prédicteur d'ordre 2 ou le prédicteur d'ordre 3.

9.2.3 Distance position locale-position prédite, pour un délai de 50 ms

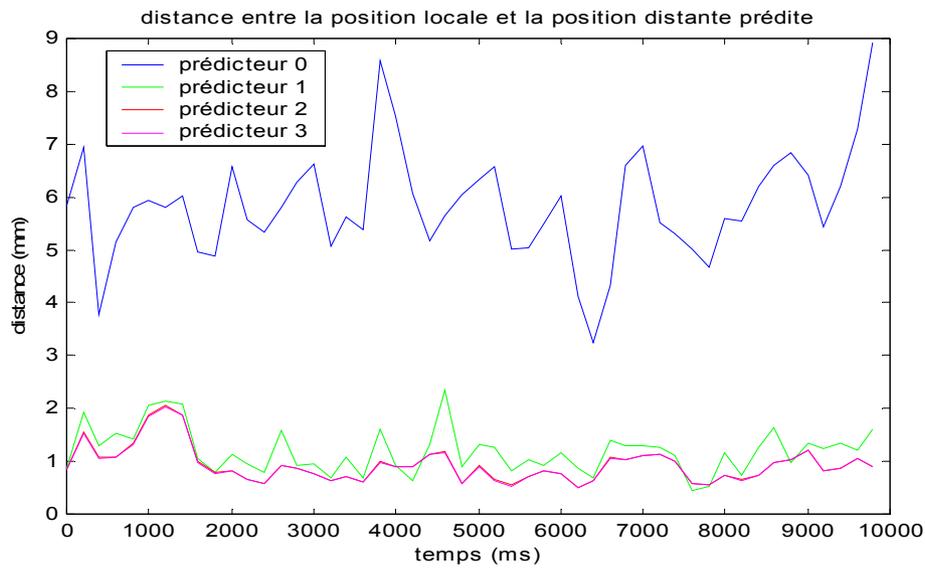


figure 9.7 : distance entre la position locale de la main de l'utilisateur et sa position distante prédite, avec un délai réseau de 50 ms, cette distance étant évaluée pour les prédicteurs d'ordre 0, 1, 2 et 3.

Nous pouvons constater sur la Figure 9.7, page 142, que la distance moyenne entre la position locale et la position distante prédite est de l'ordre de 6 mm dans le cas du prédicteur d'ordre 0. Nous rappelons que cette distance est une moyenne sur des fenêtres de largeur 200 ms, en fait on peut atteindre des distances instantanées (à la milliseconde) de l'ordre de 13 à 15 mm. Cette distance devient alors très préjudiciable pour la stabilité de la boucle haptique, comme nous le verrons dans le paragraphe suivant, à partir d'environ 30 ms, il devient très difficile de conduire l'automobile sans avoir recours aux prédicteurs proposés dans cette étude.

On s'aperçoit que cette distance est environ de 1.5 mm pour le prédicteur d'ordre 1 (courbe verte). Le prédicteur d'ordre 2 et le prédicteur d'ordre 3 nous donnent des distances entre la position locale et la position distante prédite très proches l'une de l'autre, de l'ordre de 0.9 mm. Pour des délais compris entre 30 et 50 ms, il est préférable d'employer le prédicteur d'ordre 2 ou le prédicteur d'ordre 3.

9.3 L'apport des prédicteurs en terme de confort dans un télégeste

Nous utilisons l'échelle de valeur suivante afin de donner une appréciation du ressenti de la force renvoyée dans la main de l'utilisateur lors de l'application: "le circuit automobile", en fonction du délai réseau et de l'ordre du prédicteur utilisé.

- Très Bon : on ne ressent aucune perturbation dans le geste, celui-ci se déroule comme s'il n'y avait pas de délai réseau.
- Bon : quelques perturbations sont légèrement ressenties mais elles ne gênent pas le geste à proprement parlé.
- Acceptable : les sensations de perturbations et notamment des sensations d'oscillation du système à retour d'effort commencent à gêner le mouvement de la main qui commande le véhicule (représenté par la bille rouge).
- Moyen : les oscillations sont maintenant nettement perceptibles, la progression est nettement perturbée, le conducteur est obligé de se concentrer afin de rester au centre de la route et de pouvoir terminer la course sans quitter l'application.
- Mauvais : les perturbations introduites par le délai réseau deviennent franchement pénalisantes, il est difficile de terminer les trois tours. On ressent de façon presque permanente la tendance du bras à retour d'effort à partir en oscillation.
- Impossible : nous sommes dans l'impossibilité d'effectuer les trois tours requis dans l'application afin d'être chronométré, l'appareil rentre en oscillation, on est obligé de quitter l'application.

Les résultats sont présentés dans le tableau suivant :

	Codeur Prédicteur			
	prédicteur 0	prédicteur 1	prédicteur 2	prédicteur 3
délai 10 ms	Bon	Très Bon	Très Bon	Très Bon
délai 20 ms	Acceptable	Acceptable	Très Bon	Très Bon
délai 30 ms	Mauvais	Acceptable	Très Bon	Très Bon
délai 40 ms	Mauvais	Acceptable	Bon	Bon
délai 50 ms	Impossible	Moyen	Acceptable	Acceptable
délai 60 ms	Impossible	Mauvais	Mauvais	Mauvais
délai 70 ms	Impossible	Impossible	Impossible	Impossible
délai 80 ms	Impossible	Impossible	Impossible	Impossible

Nous avons également mesuré les chronos effectués par un conducteur expérimenté, en l'occurrence l'auteur de la thèse.

Ces temps sont récapitulés dans le tableau suivant :

	en seconde	Codeur Prédicteur			
		prédicteur 0	prédicteur 1	prédicteur 2	prédicteur 3
délai 0 ms	temps	12.210	X	X	X
	pénalité	1.954	X	X	X
délai 10 ms	temps	14.495	13.102	13.001	12.157
	pénalité	2.436	1.285	1.350	0.952
délai 20 ms	temps	15.620	14.403	13.646	11.392
	pénalité	1.544	1.264	1.465	1.578
délai 30 ms	temps	19.625	15.108	14.141	12.913
	pénalité	3.025	2.732	3.726	3.054
délai 40 ms	temps	23.192	17.105	14.114	12.620
	pénalité	6.781	2.916	3.248	2.154
délai 50 ms	temps	X	19.681	15.080	12.657
	pénalité	X	3.942	3.591	2.384
délai 60 ms	temps	X	20.181	14.767	14.461
	pénalité	X	4.552	2.424	4.235

Le temps correspond au temps mis par le conducteur pour effectuer trois tours de circuit, les pénalités correspondent au temps passé en dehors de la route, ce temps est multiplié par trois. Nous n'avons pas fait de courses avec des délais supérieurs à 40 ms pour le prédicteur d'ordre 0, car il est très difficile de terminer la course. Pour ces mêmes raisons, nous n'avons pas fait de courses avec des délais réseaux dépassant les 60 ms, pour les autres prédicteurs. Pour un délai de 0 ms, nous n'avons effectué qu'une course avec le prédicteur 0, ce temps nous sert de temps de référence car il correspond à une course effectuée sans réseau.

9.4 Conclusion

Dans ce type d'application, nous pouvons directement quantifier les besoins en terme de prédiction dans le cas d'un télégeste. Ainsi nous avons pu évaluer que le système peut fonctionner sans prédicteur (prédicteur d'ordre 0), de façon satisfaisante jusqu'à un délai réseau d'environ 10 ms. Au delà des 20 ms, le geste est alors fortement perturbé, il sera donc préférable d'utiliser les prédicteurs proposés dans cette étude. Nous recommandons, le prédicteur d'ordre 3 afin d'avoir un très bon ressenti du télégeste pour des délais allant jusqu'à 40 ms, il conserve d'ailleurs des résultats tout à fait honorables jusqu'à pratiquement 60 ms.

Dans de prochaines études, nous désirons complexifier légèrement le modèle, en introduisant dans la force retournée par le modèle local vers la main de l'utilisateur, une force d'inertie. En fait, on désire attribuer une masse à la voiture, ce qui permettrait d'introduire des notions de forces centripètes qui pousseraient le véhicule à se déporter dans les virages.

Chapitre 10

Conclusion

10.1 Bilan

Nos travaux se situent au croisement de trois domaines :

- l'haptique ou la technologie du toucher,
- les réseaux de télécommunications,
- le traitement du signal.

A défaut d'être exhaustif, il nous a paru important d'en proposer un point de vue aussi synthétique que possible, tout en précisant certains détails nécessaires à la bonne compréhension des tenants et des aboutissants de notre approche.

Ainsi, nous avons dans un premier temps présenté le domaine de l'haptique, en présentant notamment plusieurs applications qui s'appuient sur la technologie du toucher.

Puis nous avons positionné le problème du télégeste. Le but de l'étude était de proposer une solution à l'instabilité de la boucle haptique (homme-système à retour d'effort-réseau-retour de la force dans la main du manipulateur) lors d'un télégeste. Cette instabilité résulte du retard introduit par la latence incompressible des réseaux de télécommunication.

L'approche "Wave Transform" de John Wilson permet de résoudre ce problème d'instabilité de la boucle haptique (Section 2.4, page 35). Il le réalise en limitant la vitesse de déplacement de la main de l'utilisateur en fonction du délai réseau. Cette démarche possède l'avantage d'être adaptative. En effet, la viscosité introduite afin de limiter la vitesse des mouvements est directement proportionnelle au délai réseau. Par contre, cet artifice qui vise à stabiliser la boucle haptique a le désavantage de limiter le manipulateur dans ses mouvements.

L'aspect innovant de cette thèse consiste donc en l'introduction des modules de prédiction dans le modèle local mais également dans le modèle distant. En fait, le module de prédiction du côté du poste distant se sert des données de position (de la main du manipulateur de SRE) pour calculer une position prédite distante qui lui permet d'évaluer la force distante que le modèle distant va envoyer par le réseau au modèle local. Le module de prédiction du modèle local calcule pour sa part la force locale prédite à partir de la force distante reçue, puis le modèle local envoie cette force vers le système à retour d'effort.

Contrairement à l'approche "Wave Transform", nous n'introduisons aucune viscosité, ce qui permet de laisser le manipulateur du SRE libre de ses mouvements, ceci pour des délais allant jusqu'à environ 60 ms.

L'étude d'un point de vue traitement du signal, a permis de dégager les caractéristiques intrinsèques des prédicteurs dérivés du modèle de l'algorithme de prédiction proposé.

Par une étude des performances des prédicteurs proposés, en premier lieu sur un signal purement théorique représenté par une sinusoïde, puis directement sur un signal gestuel numérisé à l'aide du système à retour d'effort, nous avons pu évaluer les limites des prédicteurs proposés, et leurs domaines d'application en terme de délai réseau.

Enfin, en élaborant une application réelle de télégeste, sur réseau à retard, nous avons mesuré, en terme de confort, l'apport des prédicteurs présentés dans notre étude. Nous avons également délimité le cadre d'utilisation des différents prédicteurs proposés, en terme de délai introduit par la latence incompressible des réseaux de télécommunications.

Nous pouvons conclure, en présentant le codeur prédicteur d'ordre 3, comme le plus performant des prédicteurs proposés. Cependant, son cadre d'utilisation s'étend jusqu'à environ 60 ms de délai réseau.

10.2 Perspectives

Nous pourrions envisager une mise en parallèle de notre approche basée sur le modèle de l'algorithme de prédiction et l'approche "Wave transform" pour des délais plus importants. Nous rappelons que cette approche proposée par John

Wilson [WIL 99] consiste à gérer l'instabilité de la boucle haptique (homme-système à retour d'effort-réseau-retour de la force dans la main du manipulateur) lors d'un toucher distant, en adaptant la vitesse de la main du manipulateur du SRE au délai réseau, en introduisant une contrainte de viscosité directement dépendante du délai réseau.

Il serait intéressant d'introduire le filtrage de Kalman [BER 98], dans notre modèle de prédiction. Nous rappelons que le filtrage de Kalman est couramment utilisé dans des problèmes d'estimation et notamment en prédiction. Le filtrage de Kalman s'appuie sur deux équations d'états, l'équation d'évolution et l'équation d'observation. L'élaboration de l'équation d'évolution ne poserait pas de problème car elle correspondrait à nos équations de prédictions proposées dans cette étude. Tout le problème se situe dans l'élaboration de l'équation d'observation, il serait intéressant de se documenter sur les algorithmes de suivi de trajectoire, traité notamment en robotique mais aussi dans le domaine des technologies spatiales. Il serait alors possible de dégager de ces études une équation d'observation sur la vitesse de la main.

Afin d'optimiser la procédure de prédiction, il serait intéressant de détecter les périodes durant lesquelles le manipulateur du système à retour d'effort ne bouge pas lors d'un télégeste. Durant cette période, le meilleur prédicteur n'est plus le prédicteur d'ordre 3. En effet, lorsqu'il n'y a pas de mouvement le signal est très perturbé par le bruit de quantification, ceci se répercute sur le signal prédit, par le prédicteur d'ordre 3 et se traduit par des oscillations assez désagréables. La meilleure prédiction lorsque le main du manipulateur est immobile est sans conteste le prédicteur d'ordre 0, aussi il faudrait élaborer un détecteur de "non-mouvement" qui permettrait de basculer du prédicteur d'ordre 3 au prédicteur d'ordre 0.

Il serait également nécessaire d'élaborer un module d'interpolation sur le poste distant et sur le poste local dans le cas de perte de message, car les prédicteurs proposés dans cette étude sont très sensibles aux discontinuités présentes dans le signal transmis, aussi une perte trop importante d'informations provoquerait une discontinuité qui serait alors amplifiée par les prédicteurs.

Bibliographie

Références bibliographiques

- [AND 89] Robert J. Anderson, Mark W. Spong, *Bilateral Control of Teleoperators with Time Delay*, IEEE Transactions on automatic control, vol.34, no.5, 1989.
- [AVI 99] Walter A. Aviles, John F. Ranta. *Haptic Interaction with Geoscientific Data*, Teneo Computing, PUG 1999.
- [ARI 02] H. Airoui, S. Mammar, T. Hamel, *A Smith-prediction based Haptic Feedback Controller for Time Delayed Virtual Environments systems*, Proceedings of the American Control Conference, Anchorage, AK May 2002.
- [BAR 46] M.S. Bartlett, *On the Theoretical Specification and Sampling properties of autocorrelated times series*, Journal of the Royal Statistical Society, vol. B-8, no.27, 1946.
- [BEL 90] Maurice Bellanger, *Traitement Numérique Du Signal, Théorie et Pratique*, quatrième édition, Masson, 1990.

- [BEL 01] I.Belghit, A.Guerraz, B.Hennion, A.Vienne. *The tele-gesture : problems of networked gestures*, dans les proceedings EuroHaptics 2001, pages 65-70, 2001.
- [BEL 02a] I.Belghit, A.Guerraz, B.Hennion, P.Thorel, F.Pellissier. *A Haptic Command Station for Remote Ultrasound Examinations*, Haptic Symposium, VR 2002.
- [BEL 02b] I.Belghit, A.Guerraz, B.Hennion. *Predictive Algorithms for distant touching*, dans les proceedings EuroHaptics 2002, pages 61-65, 2002.
- [BER 98] Jean Claude Bertein, Roger Ceschi, *Processus Stochastiques et filtrage de Kalman*, Collection Traitement du Signal, Hermès, 1998.
- [BLA 58] R.B. Blackman, J.W. Tukey, *The Measurement of Power Spectra*, Dover, New York, 1958.
- [BLA 98] Gérard Blanchet, Maurice Charbit, *Traitement numérique du signal, simulation sous Matlab*, Collection Pédagogique de Télécommunication, Hermès, 1998.
- [BRO 96] A. Brown. *Interpretation of Three-Dimension Seismic Data*, American Association of Petroleum Geologists, 1996.
- [CAR 61] H. Cartan, *Théorie élémentaire des fonctions analytiques d'une ou plusieurs variables complexes*, Herman, Paris, 1961.
- [CHEN 79] Chi-Tsong Chen, *One dimensional Digital Signal Processing*, Electrical Engineering and Electronics, vol.9, 1979.
- [DEL 93] J.P Delmas, *Introduction aux Probabilités*. Collection Pédagogique de Télécommunication. Ellipses, 1993.
- [DOU 96] Douglas, Comer, traduit de l'américain par J.A. Hernandez, B. Joachim et R. Joly, *TCP/IP : Architecture, protocoles, applications*, 3^e éditions, InterEditions, 1996.
- [GUI 77] L. Guikhman, A.V. Skorokhod, *Introduction à la théorie des processus aléatoires*, Editions MIR, Moscou, 1977.
- [HEN 91] J. Henshall, S. Shaw, *OSI Les normes de communication entre les systèmes ouverts*, Editions Masson, 1991.

- [HOG 88] Neuville Hogan, *On the Stability of Manipulators Performing Contact Tasks*, IEEE Journal of Robotics and Automotion, vol.4, no.6, 1988.
- [JEN 68] G.M. Jenkins, D.G. Watt, *Spectral analysis and its applications*, Holden Day, San francisco, 1968.
- [KAI 66] J.F. Kaiser, *Digital Filters*, chap.7, *System Analysis by Digital Computer*, F.F. Kuo, J.K. Kaiser (eds), Wiley, New York, 1966.
- [KAT 66] T. Kato, *Perturbation Theory for linear operators*, Springer-Verlag, New-York, 1966.
- [KAY 88] S.M Kay, *Modern Spectral Estimation, Theory and Application*, Prentice Hall, Englewoods Cliffs, 1988.
- [KLI 99] R.Kline-Schoder, J.Wilson, *Algorithm for Network-based Force Feedback*, Creare Incorporated Neville Hogan, Massachusetts Institute of Technology, PUG 1999.
- [KUN 86] Murat Kunt. *Digital Signal Procesing*. ARTECH HOUSE 1986.
- [LIN 98] S. Lin, B. Loftin, T. Start, *Virtual Reality for Geosciences Visualization*, Proceedings of Asia Pacific Human Interaction, 1998.
- [MAC 91] D. MacKinnon, W. McCrum, D. Sheppard, *Introduction à l'OSI*, Afnor Technique, 1991.
- [MAK 75] J. Makhoul, *Linear Prediction : A Tutorial Review*, proceedings IEEE, vol.63, pp 561-580, 1975.
- [MAS 93] Thomas H. Massie, *Design of a Three Degree of Freedom Force Reflecting Haptic Interface*, SB Thesis, Department of Electrical Engineering and Computer Science, MIT, 1993.
- [MAS 94] Thomas H. Massie, J.K. Salisbury, *The Phantom Haptic Interface : A Device for Probing Virtual Objects*, Proceedings of the ASME Winter Annual Meeting, Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, Chicago, IL, 1994.
- [MIC 92] François Michaut, *Méthodes adapatatives pour le signal, outils mathématiques et mise en oeuvre des algorithmes*, Traité des Nouvelles Technologies, série Traitement du signal, Hermès, 1992.

- [NIE 97] G. Niemeyer, J. Slotine, *Using Wave Variables for System Analysis and Robot Control*, IEEE International Conference on Robotics and Automation, pp. 1619-1625, Albuquerque, New Mexico, April 1997.
- [NYQ 28] H. Nyquist, *Certain topics in telegraph transmission theory*, Transactions of the American Institute of Electrical Engineers, Avril 1928.
- [OPP 89] A.V.Oppenheim, R.W. Schafer, *Discrete-Time Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1989.
- [PAL 34] Paley-Wiener, *Fourier transforms in the complex domain*, American Mathematical Society Colloquium, New York, 1934.
- [PRA 89] Jacques Prado, *Traitement numérique du signal*, TELECOM PARIS, Département Signal, ENST, 1989.
- [PRI 81] M.B. Priestley, *Spectral Analysis and Time Series*, Academic Press, New-York, 1981.
- [PUJ 89] G.Pujolle, D. Seret, D. Dromard, E. Horlait, *Réseaux et Télématique*, Editions Eyrolles , 1989.
- [RAN 99] John F. Ranta, Walter A. Aviles, *The Virtual Reality Dental Training System - Simulating dental procedures for the purpose of training dental students using haptics*, Teneo Computing, Phantom User Group 1999.
- [ROL 97] P. Rolin, G. Martineau, L. Toutain, A. Leroy, *Les réseaux, principes fondamentaux*, Editions Hermes, 1997.
- [ROS 85] M. Rosenblatt, *Stationnary Processes and Random Fields*, Birkhauser, 1985.
- [SAG 01] M. Saghir, W. Book, *Wave-Based Teleoperation with Prediction*, *Proceedings of the American Control Conference*, pp. 4605-4611, Arlington, June 2001.
- [SAL 96] J.K. Salisbury, M.A. Srinivasan, eds., *Proc. First Phantom Users Group Workshop*, AI Lab Tech. Report 1596 and RLE Tech. Report 612, Massachussetts Institute od Technology, 1996.

- [**SAL 97**] J.K. Salisbury, C. Tarr, *Haptic Rendering of Surfaces Defined by Implicit Functions*, Proc. Sixth Annual Symp. on Haptic Interfaces for virtual Environment and Teleoperator Systems, ASME, Fairfield, N.J., 1997.
- [**SCH 96**] Richard S. Schwartz, DDS, *Fundamentals of Operative Dentistry*, Quintessence Publishing, 1996.
- [**SER 97**] Claude Servin, *Télécoms de la Transmission à l'Architecture de Réseaux*, seconde édition, Masson, 1997.
- [**SHA 49**] C. Shannon, W. Weaver. *The Mathematical Theory of Communication*, University of Illinois Press, 1949.
- [**SMI 59**] O. Smith, *A Controller to overcome dead time*, ISA J., vol.6, no 2, pp.28-33, 1959.
- [**SOI 93**] Christian Soize, *Méthodes mathématiques en analyse du signal*, Masson, 1993.
- [**SRI 97**] M.A. Srinivasan, C. Basdogan, *Haptics in Virtual Environments : Taxonomy, Research Status, and Challenges*, *Computers Graphics*, Special Issue on Haptic Displays in Virtual Environments, Vol. 21, No.4, 1997.
- [**SUT 89**] P.H. Sutter, J.C. Iatridis, N.V. Thakor, *Response to reflected Force Feedback to Fingers in Teleoperations*, Proceedings of the NASA Conference On Space Telerobotics, 1989.
- [**TER 01**] G.A.Vilchis, P.Cinquin, J.Trocaz, A.Guerraz, B.Hennion, F.Pellissier, P.Thorel, F.Courreges, A.Gourdon, G.Poisson, P.Vieyres, P.Caron, O.Mérigeaux, L.Urbain, C.Daimo, S.Lavalle, P.Arbeille, M.Althuser, M.Ayoubi, B.Tondu, S.Ippolito. TER : *a system robotic tele-Echography*. Dans Lectures Notes in computer Science. Medical Image Computing and Computer-Assisted Intervention (MICCAI'01), 2001.
- [**TRE 82**] T.E Tremain, *The Governement Standard Linear Predictive Coding Algorithm*, *Speech Technology*, vol.1, pp. 40-49, 1982.
- [**Van 68**] H.L. Van Trees, *Detection, Estimation and Modulation Theory*, Part 1, Wiley, New York, 1968.

- [WEL 67] P.D. Welch, *The Use of Fast Fourier Transform for the Estimation of Power Spectra*, IEEE Trans. Audio Electroacoustics, vol. AU-15, pages 70-73, 1967.
- [WIL 99] J.P.Wilson, R.J.Kline-Schoder, M.A.Kenton, N.Hogan. *Algorithms for Network-based Force Feedback*. Fourth PHANToM Users Group Workshop (PUG99), 1999.
- [WIN 96] A-M.Wing, P.Haggard, J-R. Flanagan. *The neurophysiology and Psychology of Hand Movements*. Academic Press, 1996.

Annexe A

La Transformée en Z

Cette annexe présente formellement l'outil mathématique appliqué au traitement des signaux numériques : la transformée en Z.

La transformée en Z, $X(Z)$ de la suite $x(n)$ est définie par la relation suivante :

$$X(Z) = \sum_{n=-\infty}^{\infty} x(n) \times Z^{-n} \quad (\text{EQ 1})$$

Z est une variable complexe et la fonction $X(Z)$ possède un domaine de convergence qui en général est un anneau centré sur l'origine, de rayons $R1$ et $R2$. C'est à dire que $X(Z)$ est défini pour $R1 < |Z| < R2$. Les valeurs $R1$ et $R2$ dépendent de la suite $x(n)$. Si la suite $x(n)$ représente la suite des échantillons d'un signal prélevés avec la période T , la transformée de Fourier de cette suite s'écrit :

$$S(f) = \sum_{n=-\infty}^{\infty} x(n) \times e^{-j2\pi fnT} \quad (\text{EQ 2})$$

Ainsi, pour $Z=e^{j2\pi fT}$ la transformée en Z de la suite $x(n)$ coïncide avec sa transformée de Fourier. C'est à dire que l'analyse d'un système discret peut se faire avec la transformée en Z , pour connaître une réponse en fréquence il suffit de remplacer Z par $e^{j2\pi fT}$.

Cette transformée possède une transformée inverse. Soit Γ un contour fermé contenant tous les points singuliers, ou pôles, de $X(Z)$ ainsi que l'origine; on peut écrire :

$$\begin{aligned} Z^{m-1} \times X(Z) &= \sum_{n=-\infty}^{\infty} x(n) \times Z^{m-1-n} \\ &= \left(x(m) \times Z^{-1} + \sum_{n \neq m} x(n) \times Z^{m-1-n} \right) \end{aligned} \quad (\text{EQ 3})$$

et d'après le théorème des résidus :

$$x(m) = \frac{1}{2\pi j} \times \oint_{\Gamma} Z^{m-1} \times X(Z) \times dZ \quad (\text{EQ 4})$$

Par exemple si $x(Z) = \frac{1}{1-p \times Z^{-1}}$, on obtient par application directe de l'équation ci-dessus :

$$\begin{aligned} x(n) &= p^n & \text{pour} & \quad n \geq 0 \\ x(n) &= 0 & \text{pour} & \quad n < 0 \end{aligned}$$

De même à $X(Z)$ définie par :

$$X(Z) = \sum_{i=1}^N \frac{a_i}{1-p_i \times Z^{-1}} \quad (\text{EQ 5})$$

correspond la suite $x(n)$ telle que :

$$\begin{aligned} x(n) &= \sum_{i=1}^N a_i \times p_i^n & \text{pour} & \quad n \geq 0 \\ x(n) &= 0 & \text{pour} & \quad n < 0 \end{aligned} \quad (\text{EQ 6})$$

Une condition de stabilité apparaît très simplement en observant que la suite est bornée si, et seulement si $|p_i| < 1$ pour $1 \leq i \leq N$, c'est à dire que les pôles de $X(Z)$ sont à l'intérieur du cercle de rayon unité.

Dans ces exemples, les termes de la suite $x(n)$ peuvent aussi être obtenus directement par développement en série. Quand $X(Z)$ est une fraction rationnelle,

une méthode très simple pour obtenir les premières valeurs de la suite $x(n)$ consiste à faire une division de polynômes. Par exemple pour :

$$X(Z) = \frac{1 + 2 \times Z^{-1} + Z^{-2} + Z^{-3}}{1 - Z^{-1} - 8 \times Z^{-2} + 12 \times Z^{-3}} \quad (\text{EQ 7})$$

la division directe donne :

$$X(Z) = 1 + 3 \times Z^{-1} + 12 \times Z^{-2} + 25 \times Z^{-3} + \dots$$

d'où :

$$x(0) = 1; \quad x(1) = 3; \quad x(2) = 12; \quad x(3) = 25;$$

La transformation en Z possède la propriété de linéarité. D'autre part la transformée en Z de la suite retardée $x(n-n_0)$ s'écrit :

$$X_{n_0}(Z) = Z^{-n_0} \times X(Z)$$

Ces deux propriétés sont utilisées pour calculer la transformée en Z , $Y(Z)$, de la suite $y(n)$ obtenue en sortie d'un système linéaire discret, par convolution des suites $x(n)$ et $h(n)$ qui ont pour transformées $X(Z)$ et $H(Z)$.

En calculant la transformée en Z des deux membres de l'équation de convolution :

$$y(n) = \sum_m h(m) \times x(n - m)$$

il vient :

$$Y(Z) = \sum_m h(m) \times Z^{-m} \times X(Z) = H(Z) \times X(Z)$$

Par suite la transformée en Z d'un produit de convolution est le produit des transformées. La fonction $H(Z)$ est appelée fonction de transfert en Z du système LIT (Linéaires discrets Invariants dans le Temps) considéré.

La transformée en Z du produit de deux suites $x_3(n) = x_1(n) \times x_2(n)$ est la fonction $X_3(Z)$ définie par :

$$X_3(Z) = \frac{1}{2\pi j} \times \oint_{\Gamma} X_1(v) \times X_2\left(\frac{Z}{v}\right) \times v^{-1} \times dv \quad (\text{EQ 8})$$

Le contour d'intégration est à l'intérieur du domaine de convergence des fonctions $X_1(v)$ et $X_2\left(\frac{Z}{v}\right)$.

L'application aux suites causales amène à introduire la transformation en Z monolatérale.

La transformation en Z monolatérale de la suite $x(n)$ s'écrit :

$$X(Z) = \sum_{n=0}^{\infty} x(n) \times Z^{-n} \quad (\text{EQ 9})$$

Les propriétés sont les mêmes que celles de la transformation définie par la relation (EQ 1), sauf pour les suites retardées. En effet la transformée de la suite $x(n-n_0)$ s'écrit :

$$X_{n_0}(Z) = \sum_{n=0}^{\infty} x(n-n_0) \times Z^{-n} = Z^{-n_0} \times X(Z) + \sum_{n=1}^{n_0} x(-n) \times Z^{-(n_0-n)} \quad (\text{EQ 10})$$

L'intérêt de cette transformation est de prendre en compte les conditions initiales et de faire apparaître les régimes transitoires dans l'étude de la réponse d'un système. D'autre part elle permet de déterminer à partir de $X(Z)$ les valeurs extrêmes de la suite $x(n)$. La valeur initiale $x(0)$ s'écrit :

$$x(0) = \lim_{Z \rightarrow \infty} X(Z)$$

et la valeur finale :

$$x(\infty) = \lim_{Z \rightarrow 1} (Z-1) \times X(Z)$$

Pour des développements plus importants sur la transformation en Z et ses applications, on peut se reporter à la référence [SOI 93].

Annexe B

Quelques procédures

Ci-joint, quelques procédures (programmée en C) utilisées dans l'application du circuit automobile. En premier lieu, nous présentons une procédure qui permet de calculer les dérivées successives d'un signal par différences finies. Puis nous proposons une procédure permettant de gommer les discontinuités présentes dans un signal, elle nous permet de nourrir les prédicteurs avec des signaux lissés afin d'obtenir de meilleurs résultats. Nous proposons également une procédure permettant de réaliser un filtrage de type elliptique, c'est plus précisément la procédure qui nous permet d'éliminer le bruit de quantification présent dans la dérivée première du signal.

B.1 La dérivation par différence finies

Cette procédure nous permet d'évaluer les dérivées successives de la position de la main du manipulateur, ce qui nous donne accès à la vitesse mais aussi à l'accélération de la main.

```

/*variables globales*/
int i_date_dP1=0; /* derivationP1*/
gstVector entree_dP1_pred;

/*procédure de dérivation position prédit par différences finies*////
gstVector derivation_P1(gstVector entree1_dP1,gstVector entree2_dP1)
{
    gstVector sortie_dP1;
    if(i_date_dP1==0) entree_dP1_pred=entree2_dP1;
    sortie_dP1=entree1_dP1-entree_dP1_pred;
    entree_dP1_pred=entree1_dP1;
    i_date_dP1++;
    if (i_date_dP1>0) i_date_dP1=1;
    return (sortie_dP1);
}
/*fin de la procédure*/

```

B.2 Le filtre RIF 10

Cette procédure nous permet de filtrer par un système MA (movind average), le signal représentant la force de retour dans la main de l'utilisateur, en éliminant les discontinuités trop importantes.

```

/*variables globales*/
int i_date_R10=0;
const cpmax_R10=10; /* ordre du rif 10*/
gstVector rgt_entree_R10[cpmax_R10];
/*procédure de filtrage par un filtre rif d'ordre 10*////
gstVector rif10(gstVector entree)
{
    int i;
    double som0_entree=0;
    double som1_entree=0;
    double som2_entree=0;
    double order;
    gstVector sortie;
    rgt_entree_R10[cpmax_R10-1][0]=entree[0];
    rgt_entree_R10[cpmax_R10-1][1]=entree[1];
    rgt_entree_R10[cpmax_R10-1][2]=entree[2];
    if(i_date_R10==0)
    {
        sortie=entree;
    }
    else
    {
        order=i_date_R10+1;
        for (i=cpmax_R10-1;i>cpmax_R10-1-order;i--)
        {
            som0_entree+=rgt_entree_R10[i][0];
            som1_entree+=rgt_entree_R10[i][1];
            som2_entree+=rgt_entree_R10[i][2];
        }
        sortie[0]=som0_entree/order;
        sortie[1]=som1_entree/order;
        sortie[2]=som2_entree/order;
    }
    if(i_date_R10<cpmax_R10-1) i_date_R10++;
    for(i=0;i<cpmax_R10-1;i++)
        rgt_entree_R10[i]=rgt_entree_R10[i+1];
    return(sortie);
}
/*fin de la procédure*/

```

B.3 Le filtre elliptique d'ordre 5

Cette procédure permet de lisser la dérivée de la vitesse par un filtre elliptique d'ordre 5, afin de rendre la contribution de l'accélération exploitable dans notre calcul de prédiction.

```
/* variables globales*/
const coeff_min_acc=0;
const coeff_max_acc=5;
double coeff_bv5[coeff_max_acc];
double coeff_av5[coeff_max_acc];

/* chargement des tableaux coeff_bv5 et coeff_av5*/
file_coeff_vit_E5=fopen("c:\\isma_c\\program\\test\\matlab_test\\
                                coeff_vitesse_E5.dat","r");
for (i=coeff_min_acc;i<coeff_max_acc;i++)
{
if(file_coeff_vit_E5)fscanf(file_coeff_vit_E5,"%lf %lf\n",
                                &coeff_bv5[i],&coeff_av5[i]);
}
fclose(file_coeff_vit_E5);

/*procédure de filtrage de l'accélération par un filtre elliptique
passe bas d'ordre 5*////

gstVector vit_liss_elliptic5(gstVector entree_E5)
{
    int i;
    gstVector sortie_E5;
    if (i_date_E5==0)
    {
        rgt_entree_E5[0]=entree_E5;
        rgt_sortie_lisse_E5[0][0]=coeff_bv5[0]*entree_E5[0];
        rgt_sortie_lisse_E5[0][1]=coeff_bv5[0]*entree_E5[1];
        rgt_sortie_lisse_E5[0][2]=coeff_bv5[0]*entree_E5[2];
        sortie_E5=rgt_sortie_lisse_E5[0];
    }
    if (i_date_E5==1)
    {
        rgt_entree_E5[1]=entree_E5;
        rgt_sortie_lisse_E5[1][0]=coeff_bv5[0]*rgt_entree_E5[1][0]
            +coeff_bv5[1]*rgt_entree_E5[0][0]
            -coeff_av5[1]*rgt_sortie_lisse_E5[0][0];

        rgt_sortie_lisse_E5[1][1]=coeff_bv5[0]*rgt_entree_E5[1][1]
            +coeff_bv5[1]*rgt_entree_E5[0][1]
            -coeff_av5[1]*rgt_sortie_lisse_E5[0][1];

        rgt_sortie_lisse_E5[1][2]=coeff_bv5[0]*rgt_entree_E5[1][2]
            +coeff_bv5[1]*rgt_entree_E5[0][2]
            -coeff_av5[1]*rgt_sortie_lisse_E5[0][2];
        sortie_E5=rgt_sortie_lisse_E5[1];
    }
    if (i_date_E5==2)
    {
        rgt_entree_E5[2]=entree_E5;
        rgt_sortie_lisse_E5[2][0]=coeff_bv5[0]*rgt_entree_E5[2][0]
            +coeff_bv5[1]*rgt_entree_E5[1][0]
            +coeff_bv5[2]*rgt_entree_E5[0][0]
            -coeff_av5[1]*rgt_sortie_lisse_E5[1][0]
            -coeff_av5[2]*rgt_sortie_lisse_E5[0][0];

        rgt_sortie_lisse_E5[2][1]=coeff_bv5[0]*rgt_entree_E5[2][1]
            +coeff_bv5[1]*rgt_entree_E5[1][1]
            +coeff_bv5[2]*rgt_entree_E5[0][1]
            -coeff_av5[1]*rgt_sortie_lisse_E5[1][1]
            -coeff_av5[2]*rgt_sortie_lisse_E5[0][1];
    }
}
```

```

    rgt_sortie_lisse_E5 [2] [2]=coeff_bv5[0]*rgt_entree_E5 [2] [2]
        +coeff_bv5[1]*rgt_entree_E5 [1] [2]
        +coeff_bv5[2]*rgt_entree_E5 [0] [2]
        -coeff_av5[1]*rgt_sortie_lisse_E5 [1] [2]
        -coeff_av5[2]*rgt_sortie_lisse_E5 [0] [2];
    sortie_E5=rgt_sortie_lisse_E5 [2];
}
if (i_date_E5==3)
{
    rgt_entree_E5 [3]=entree_E5;
    rgt_sortie_lisse_E5 [3] [0]=coeff_bv5[0]*rgt_entree_E5 [3] [0]
        +coeff_bv5[1]*rgt_entree_E5 [2] [0]
        +coeff_bv5[2]*rgt_entree_E5 [1] [0]
        +coeff_bv5[3]*rgt_entree_E5 [0] [0]
        -coeff_av5[1]*rgt_sortie_lisse_E5 [2] [0]
        -coeff_av5[2]*rgt_sortie_lisse_E5 [1] [0]
        -coeff_av5[3]*rgt_sortie_lisse_E5 [0] [0];

    rgt_sortie_lisse_E5 [3] [1]=coeff_bv5[0]*rgt_entree_E5 [3] [1]
        +coeff_bv5[1]*rgt_entree_E5 [2] [1]
        +coeff_bv5[2]*rgt_entree_E5 [1] [1]
        +coeff_bv5[3]*rgt_entree_E5 [0] [1]
        -coeff_av5[1]*rgt_sortie_lisse_E5 [2] [1]
        -coeff_av5[2]*rgt_sortie_lisse_E5 [1] [1]
        -coeff_av5[3]*rgt_sortie_lisse_E5 [0] [1];

    rgt_sortie_lisse_E5 [3] [2]=coeff_bv5[0]*rgt_entree_E5 [3] [2]
        +coeff_bv5[1]*rgt_entree_E5 [2] [2]
        +coeff_bv5[2]*rgt_entree_E5 [1] [2]
        +coeff_bv5[3]*rgt_entree_E5 [0] [2]
        -coeff_av5[1]*rgt_sortie_lisse_E5 [2] [2]
        -coeff_av5[2]*rgt_sortie_lisse_E5 [1] [2]
        -coeff_av5[3]*rgt_sortie_lisse_E5 [0] [2];
    sortie_E5=rgt_sortie_lisse_E5 [3];
}
if (i_date_E5==4)
{
    rgt_entree_E5 [4]=entree_E5;
    rgt_sortie_lisse_E5 [4] [0]=coeff_bv5[0]*rgt_entree_E5 [4] [0]
        +coeff_bv5[1]*rgt_entree_E5 [3] [0]
        +coeff_bv5[2]*rgt_entree_E5 [2] [0]
        +coeff_bv5[3]*rgt_entree_E5 [1] [0]
        +coeff_bv5[4]*rgt_entree_E5 [0] [0]
        -coeff_av5[1]*rgt_sortie_lisse_E5 [3] [0]
        -coeff_av5[2]*rgt_sortie_lisse_E5 [2] [0]
        -coeff_av5[3]*rgt_sortie_lisse_E5 [1] [0]
        -coeff_av5[4]*rgt_sortie_lisse_E5 [0] [0];

    rgt_sortie_lisse_E5 [4] [1]=coeff_bv5[0]*rgt_entree_E5 [4] [1]
        +coeff_bv5[1]*rgt_entree_E5 [3] [1]
        +coeff_bv5[2]*rgt_entree_E5 [2] [1]
        +coeff_bv5[3]*rgt_entree_E5 [1] [1]
        +coeff_bv5[4]*rgt_entree_E5 [0] [1]
        -coeff_av5[1]*rgt_sortie_lisse_E5 [3] [1]
        -coeff_av5[2]*rgt_sortie_lisse_E5 [2] [1]
        -coeff_av5[3]*rgt_sortie_lisse_E5 [1] [1]
        -coeff_av5[4]*rgt_sortie_lisse_E5 [0] [1];

    rgt_sortie_lisse_E5 [4] [2]=coeff_bv5[0]*rgt_entree_E5 [4] [2]
        +coeff_bv5[1]*rgt_entree_E5 [3] [2]
        +coeff_bv5[2]*rgt_entree_E5 [2] [2]
        +coeff_bv5[3]*rgt_entree_E5 [1] [2]
        +coeff_bv5[4]*rgt_entree_E5 [0] [2]
        -coeff_av5[1]*rgt_sortie_lisse_E5 [3] [2]
        -coeff_av5[2]*rgt_sortie_lisse_E5 [2] [2]
        -coeff_av5[3]*rgt_sortie_lisse_E5 [1] [2]
        -coeff_av5[4]*rgt_sortie_lisse_E5 [0] [2];
    sortie_E5=rgt_sortie_lisse_E5 [4];
}
if (i_date_E5>4)
{
    for (i=0;i<4;i++)
    {
        rgt_entree_E5 [i]=rgt_entree_E5 [i+1];
        rgt_sortie_lisse_E5 [i]=rgt_sortie_lisse_E5 [i+1];
    }
    rgt_entree_E5 [4]=entree_E5;
}

```

```

rgt_sortie_lisse_E5[4][0]=coeff_bv5[0]*rgt_entree_E5[4][0]
+coeff_bv5[1]*rgt_entree_E5[3][0]
+coeff_bv5[2]*rgt_entree_E5[2][0]
+coeff_bv5[3]*rgt_entree_E5[1][0]
+coeff_bv5[4]*rgt_entree_E5[0][0]
-coeff_av5[1]*rgt_sortie_lisse_E5[3][0]
-coeff_av5[2]*rgt_sortie_lisse_E5[2][0]
-coeff_av5[3]*rgt_sortie_lisse_E5[1][0]
-coeff_av5[4]*rgt_sortie_lisse_E5[0][0];

rgt_sortie_lisse_E5[4][1]=coeff_bv5[0]*rgt_entree_E5[4][1]
+coeff_bv5[1]*rgt_entree_E5[3][1]
+coeff_bv5[2]*rgt_entree_E5[2][1]
+coeff_bv5[3]*rgt_entree_E5[1][1]
+coeff_bv5[4]*rgt_entree_E5[0][1]
-coeff_av5[1]*rgt_sortie_lisse_E5[3][1]
-coeff_av5[2]*rgt_sortie_lisse_E5[2][1]
-coeff_av5[3]*rgt_sortie_lisse_E5[1][1]
-coeff_av5[4]*rgt_sortie_lisse_E5[0][1];

rgt_sortie_lisse_E5[4][2]=coeff_bv5[0]*rgt_entree_E5[4][2]
+coeff_bv5[1]*rgt_entree_E5[3][2]
+coeff_bv5[2]*rgt_entree_E5[2][2]
+coeff_bv5[3]*rgt_entree_E5[1][2]
+coeff_bv5[4]*rgt_entree_E5[0][2]
-coeff_av5[1]*rgt_sortie_lisse_E5[3][2]
-coeff_av5[2]*rgt_sortie_lisse_E5[2][2]
-coeff_av5[3]*rgt_sortie_lisse_E5[1][2]
-coeff_av5[4]*rgt_sortie_lisse_E5[0][2];
sortie_E5=rgt_sortie_lisse_E5[cpmax_E5-1];
}
i_date_E5++ ;
return(sortie_E5);
}
/*fin de la procédure*/

```


Annexe C

La prédiction de Smith

Cette annexe présente l'utilisation de la prédiction de Smith sur un exemple d'école. Rappelons que cet outil proposé par O. Smith en 1959, a été dérivé par M. Saghir et H. Arioui afin de stabiliser le processus de contrôle d'une téléopération.

Considérons le système mono-variable défini par les équations suivantes :

$$\begin{aligned}x(t) &= Ax(t) + Bu(t - \delta) \\y(t) &= Cx(t) \\x(0) &= x_0 \quad u(h) = u_0(h) \quad -\delta \leq h < 0\end{aligned} \quad (\text{EQ 1})$$

où u correspond l'entrée de contrôle, x est la variable d'état, y correspond à la sortie, δ correspond au délai réseau et u_0 est l'état initial de l'entrée de contrôle.

A , B , C correspondent à des matrices à coefficients constants, dont les dimensions sont appropriées au processus.

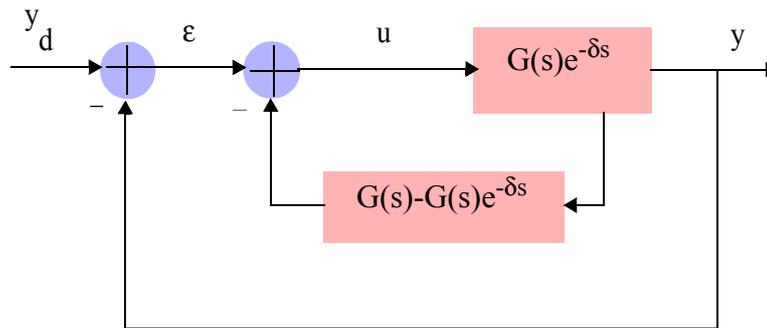
La fonction de transfert du système présenté par l'équation (EQ 1) est donnée par l'équation suivante :

$$\frac{Y(s)}{Y_d(s)} = \frac{C(sI - A)^{-1} B e^{-s\delta}}{1 + C(sI - A)^{-1} B e^{-s\delta}} \quad (\text{EQ 2})$$

où s est la variable de la transformée de Laplace.

Dans cette équation, on observe que l'équation caractéristique de la boucle fermée du système contient un terme fonction du délai. Il en résulte que l'équation caractéristique possède un nombre infini de valeurs propres.

Or il est impossible de contrôler un nombre infini de valeurs propres. En utilisant la prédiction de Smith, le terme comprenant le délai peut être éliminer de l'équation caractéristique de la boucle fermée du système. En fait, le circuit représentant le processus peut être remplacé par un circuit dans lequel le terme comprenant le délai aura été éliminé.



Architecture de contrôle du système représenté par (EQ 1), basée sur la prédiction de Smith

où ϵ représente l'erreur, y_d est le signal d'entrée, y correspond à la sortie du système et u est le signal de contrôle.

Le bloc $G(s)e^{-\delta s}$, dans lequel $G(s)=C(sI-A)^{-1}B$, est le modèle du processus sans délai. L'élément $G(s)-G(s)e^{-\delta s}$ correspond au modèle nominal qui est considéré comme bien identifié.

La fonction de transfert (EQ 2) devient après l'application de la prédiction de Smith de la forme suivante :

$$\frac{Y(s)}{Y_d(s)} = \frac{C(sI - A)^{-1} B e^{-s\delta}}{1 + C(sI - A)^{-1} B} \quad (\text{EQ 3})$$

Le système représenté par la fonction de transfert ci-dessus est alors stable sous certaines conditions liées aux matrices A , B , C .

On pose : $Q(s) = \frac{Y(s)}{Y_d(s)}$

Pour que le système soit stable en utilisant la prédiction de Smith, il est impératif que :

$$\lim_{w \rightarrow \infty} |Q(jw)| < \frac{1}{2} \quad (\text{EQ 4})$$

La boucle fermée du système sera stable pour tous les variations du délai si :

$$|Q(jw)| < \frac{1}{2} \quad \forall (w \geq 0) \quad (\text{EQ 5})$$

