Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Ecole Nationale Supérieure des Travaux Publics

Direction Post-Graduation & Recherche Scientifique

Mémoire de Magister

Option : Matériaux et Structures

LARBI SELMA HANANE

Ingénieur d'Etat en Travaux Publics
Diplômée de l'Ecole Nationale Supérieure des Travaux Publics

Intitulé du mémoire de Magister

CONTROLE OFFLINE DE LA TABLE VIBRANTE EN UTILISANT LES RESEAUX DE NEURONES ARTIFICIELS

Soutenu le : …….

Devant le jury composé de :

M M.BENSAIBI	Professeur, ENSTP Alger	Président
M N.BOURAHLA	Professeur, USTB	Rapporteur
M M.HADID	Professeur, ENSTP Alger	Examinateur
M H.BECHTOULA	Maître de recherche, CGS	Examinateur
M H.BENCHOUBANE	Professeur, USTB	Invité

À mon mari,

Ceci est sa revanche.

DEDICACES

Je dédie ce travail :

À mon mari, mes parents, mes petits anges YAAKOUB et MERIEM

À ma fille.

REMERCIEMENTS

Je tiens tout d'abord à remercier mon encadreur le professeur Nouredine BOURAHLA, avec qui j'ai eu la chance de travailler sur ce projet de recherche. Ce projet qui m'a permis de toucher à d'autres disciplines que la mienne comme l'Electronique ou surtout l'Automatisme. Je le remercie pour sa disponibilité, sa simplicité, ses encouragements et tout le temps qu'il m'a accordé et qu'il a accordé pour lire chaque mot de ce mémoire. Je le remercie vivement pour son investissement dans ce travail et de m'avoir poussée à aller plus loin chaque fois.

Je remercie du plus profond de mon cœur, mon mari, qui n'a jamais cessé de m'encourager, jour après jour, qui n'a jamais cessé de croire en moi, qui n'a jamais cessé de m'aider. Je le remercie pour son amour, sa patience, sa compréhension, son soutien et son réconfort continuel.

Un énorme merci à mes parents, les éternels piliers sur lesquels je me repose. Ils continuent à m'encourager, à m'aider et à me procurer les meilleures conditions possibles pour uniquement bien travailler, malgré la distance ils sont si présents, si attentifs, si réconfortants. Merci.

Je remercie ma sœur, qui m'a transmise son courage, son amour pour le travail bien fait, sa force d'aller toujours plus loin.

Je remercie ma petite cousine Soumia de m'avoir sans cesse encouragée, jour après jour. Je lui souhaite autant de réussite dans ses études.

Je remercie l'ensemble des membres du jury pour l'attention qu'ils ont porté envers ce travail et pour le temps qu'ils ont consacré à l'évaluer.

Je remercie le Professeur BENCHOUBANE et son doctorant M. CHOUTRI du département d'Aéronautique de l'Université de Blida pour m'avoir apporté une très grande aide dans un domaine que je ne maitrise pas qu'est l'Automatisme, un domaine qui s'est avéré nécessaire dans ce projet de recherche. Je les remercie de m'avoir aidée sans compter, de m'avoir accordé de leur temps et d'avoir toujours répondu présents.

Je remercie Ismail DERBAL, qui m'a beaucoup aidée et conseillée depuis le début de mon travail.

Je remercie l'ensemble de mes professeurs qui m'ont beaucoup appris tant professionnellement que personnellement durant ma première année de magister.

Je remercie tous ceux qui m'ont encouragée durant ces trois années de Magister.

ملخص

يكمن الأشكال الرئيسي في الطاولات الهزازة في محاكات اشارات أهتزاز بدقة عالية. هذه الأخيرة دائما ما تكون غير مطابقة بفعل لاخطية الأجهزة, النموذج و التفاعل بينهم.

في هذه الدراسة, تم وضع ضابط منفصل عن طريق الشبكات العصبية اللأصطناعية بهدف الحد من الفوارق بين اللأهتزازات المرجوة و تلك الناشئة.

تم وضع نموذج الطاولة الهزازة بطريقة العناصر المنتهية بأخذ بعين اللأعتبار الأجهزة الميكانيكية للطاولة من دون ضابط متصل, و نموذج آخر عن طريق دوال التحويل و ذلك لتمثيل نموذج مبسط للأجهزة مع الضابط المتصل PID (Proportional – Integral - Derivative)

تم القيام بتحاليل ديناميكية و ذلك لأنشاء قاعدة بيانات بغية تلقين و أختبار الشبكة العصبية. بعد ذلك تمت المصادقة بأختبار عدة تسارعات لهزات أرضية.

النتائج جاءت بتطابق كبير بين اللأهتزازات المرجوة و الناشئة, هذا ما يدل على أن الشبكة العصبية قادرة على توليد اشارات أهتزاز بدقة عالية حتى في الحالات اللاخطية للأجهزة و النماذج سواء كانت صلبة ام مرنة. في المرحلة الثانية, اشارات الأهتزازات المولدة من طرف الشبكة العصبية في ظرف وجود الضابط PID كانت قريبة من الأشارات المرجوة خاصة في حالة اشارات أهتزازذات نبض صغير لأنه تبين أن اشارات أهتزازذات نبض كبير تؤدي ألى نقص في دقة الشبكة العصبية في توليد اشارات أهتزازفي التسارع.

عموما, النتائج بينت أن طريقة الضبط بأستخدام الشبكة العصبية الأصطناعية توفر أمكانية تصحيح فعال لأشارات الأهتزاز المولدة في الطاولة الهزازة.

كلمات اساسية: طاولة هزازة, شبكة عصبية أصطناعية, ضبط, ضبط عصبي.

RESUME

Un souci majeur des tables vibrantes est la reproduction des signaux avec une grande précision. Les signaux demeurent encore simulés avec un certain degré de distorsion à cause des fortes non-linéarités du système et du spécimen, mais aussi l'interaction table-spécimen. Dans cette présente étude, un contrôleur offline basé sur les réseaux de neurones artificiels (RNA) a été développé dans le but de réduire ces distorsions de signal. Un premier modèle en éléments finis de la table vibrante a été élaboré, représentant les composants mécaniques de la table, sans aucun contrôleur online. Un second modèle de la table a été fait sous forme de fonctions de transfert implémentées dans Matlab, représentant un modèle simplifié du système entier de la table avec un contrôleur online PID (Proportional – Integral - Derivative). Des analyses dynamiques ont été réalisées afin de construire la base de données qui servira à l'apprentissage et au test du RNA. Après l'apprentissage, le RNA proposé est testé puis validé en utilisant plusieurs accélérogrammes. Dans le cas d'une table non chargée et chargée, le coefficient de corrélation R obtenu entre le signal de réponse du RNA et le signal désiré est de 99%. Ceci indique que le RNA est capable de prédire le signal désiré avec une très haute précision même en présence de vérins non linéaires et de spécimen non linéaire rigide ou flexible. En seconde partie, les signaux de commande prédits par le RNA en présence du contrôleur PID se sont avérés satisfaisants dans la réduction du déphasage temporel et des distorsions à faibles amplitudes. Comme le PID est un contrôleur de déplacements, le système de correction perd de sa précision en accélération pour de signaux à grandes amplitudes en déplacement. Globalement, les résultats ont montré que la

méthode de contrôle par RNA a un grand potentiel pour la correction efficace des signaux reproduits sur table vibrante.

Mots-clés : table vibrante, réseaux de neurones artificiels, contrôle, contrôle neuronale.

ABSTRACT

A main concern of shaking tables is the high fidelity in signal reproduction. The signals are still simulated with some degree of distortion due to inherent nonlinearities of the system and the specimen, and the interaction table-specimen. In this study, an offline neural network (NN) controller is developed in order to reduce the signal distortion. First, the shaking table is simulated using a Finite Element model representing the mechanical components of the shaking table without any online controller. A second model of the shaking table is implemented as transfer functions in Matlab representing a simplified scheme of the entire shaking table system with a PID (Proportional – Integral – Derivative) online controller. Series of dynamic analysis are carried out to constitute a database to train and test the NN. The proposed NN algorithm is trained, tested then validated using several acceleration signals. For a bare and loaded table, the correlation coefficient R between the network response and the target lines 99%. It indicates that the NN is able to predict the desired signal with a high accuracy even for nonlinear actuators and nonlinear rigid or flexible specimens. In the second stage, the commanded signals predicted by the NN in presence of the online PID controller were successful in reducing delay and amplitude distortions at low amplitudes. Hence the PID controller is based on displacement, the correction may lose accuracy in acceleration for large displacement signals. The overall results show that the NN control method has a great potential to help matching desired and reproduced signals in shaking tables.

Key words: shaking table, neural networks, control, neural network control.

LISTE DES FIGURES

Figure.2.1. Schéma simplifié et photos réelles de la table vibrante du CGS 6

Figure 2.2. Effet de l'interaction table-spécimen sur la fonction de transfert du système 12

Figure 2.3. Schéma représentant le principe de contrôle d'une table vibrante 13

Figure 3.1. Cellule nerveuse et ses composantes 23

Figure 3.2. Modèle mathématique d'un neurone formel 24

Figure 3.3. Schéma d'un réseau récurrent 26

Figure 3.4. Schéma d'un réseau de neurones à couches 27

Figure 3.5. Schéma de (a) connexion directe et (b) connexions récurrentes d'un réseau de neurones 28

Figure 4.1. Mise en évidence d'une distorsion de signal 33

Figure 4.2. Schéma de la procédure d'ajustement du signal par RNA 33

Figure 4.3. Signal d'entrée du réseau 34

Figure 4.4 : signal désiré du réseau 35

Figure 4.5. schéma mathématique de la table vibrante [Ozcelik et al., 2008] 36

Figure 4.6. Schéma représentatif de la table vibrante et de ses équipements 37

Figure 4.7. Modèle par Eléments Finis de la table avec le spécimen 38

Figure 4.8. Composants et diagramme d'interaction de la table vibrante [luco, conte et ozceli, workshop, 2010] 38

Figure 4.9. Schéma des fonctions de transfert du système [Conte et Trombetti, 2000] 39

Figure 4.10. Fonction de transfert totale de la table vibrante sans spécimen 42

Figure 4.11. Réseau de neurones de type (a) Feed-Forward (b) Recurrent 45

Figure 4.12. Réseau d'Elman 47

Figure 4.13. (a) bonne généralisation (b) mauvaise généralisation- réseau souple [Svozil et al., 1997]..... 49

Figure 4.14. Structure du réseau de neurones récurrent utilisé 53

Figure 5.1. Table non chargée avec vérins linéaires ou non linéaires (linear/nonlinear damper) 55

Figure 5.2. Table chargée (spécimen linéaire) avec vérins linéaires ou non linéaires 56

Figure 5.3. Table chargée avec spécimen non linéaire (plastic), vérins non linéaires (nonlinear damper)..... 56

Figure 5.4. Mise en évidence de la distorsion de signal (différence entre le signal de réponse et le signal cible) du modèle N°1 57

Figure 5.5. L'erreur quadratique moyenne entre le signal de réponse et le signal cible du modèle N°1…… 58

Figure5.6. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°1 (accélérogramme d'El Centro) …….. 58

Figure 5.7. Signal de réponse et signal cible du modèle N°1 …… 59

Figure 5.8. Mise en évidence de la distorsion de signal (différence entre le signal de réponse et le signal cible) du modèle N°2 …… 59

Figure 5.9. L'erreur quadratique moyenne entre le signal de réponse et le signal cible du modèle N°2…….. 60

Figure 5.10. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°2 accélérogramme d'El Centro) …….. 60

Figure 5.11. Signal de réponse et signal cible du modèle N°2 …… 61

Figure 5.12. Courbe force-déplacement de vérin horizontal de la table vibrante …….. 61

Figure 5.13. Mise en évidence de la distorsion de signal (différence entre le signal de réponse et le signal cible) du modèle N°3…… 62

Figure 5.14. L'erreur quadratique moyenne entre le signal de réponse et le signal cible du modèle N°3…….. 62

Figure 5.15. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°3 (accélérogramme d'El Centro)…… 63

Figure 5.16. Signal de réponse et signal cible du modèle N°3…….. 63

Figure 5.17. Mise en évidence de la distorsion de signal (différence entre le signal de réponse et le signal cible) du modèle N°4…….. 64

Figure 5.18. L'erreur quadratique moyenne entre le signal de réponse et le signal cible du modèle N°4…….. 64

Figure 5.19. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°4 (accélérogramme d'El Centro)…… 65

Figure 5.20. Signal de réponse et signal cible du modèle N°4…….. 65

Figure 5.21. Courbe force-déplacement de vérin horizontal de la table vibrante…….. 66

Figure 5.22. L'erreur quadratique moyenne entre le signal de réponse et le signal cible du modèle N°5…….. 66

Figure 5.23. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°5…….. 67

Figure 5.24. Signal de réponse et signal cible du modèle N°5…… 67

Figure 5.25. Courbe force-déplacement de vérin horizontal de la table vibrante…….. 68

Figure 5.26. Courbe force-déplacement de la base du spécimen…… 68

Figure 5.27. L'erreur quadratique moyenne entre le signal de réponse et le signal cible du modèle N°6…… 69

Figure 5.28. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°6…… 69

Figure 5.29. Signal de réponse et signal cible du modèle N°6…….. 70

Figure 5.30. Courbe force-déplacement de vérin horizontal de la table vibrante…….. 70

Figure 5.31. Courbe force-déplacement de la base du spécimen…….. 71

Figure 5.32. Schéma illustratif de la procédure de contrôle par RNA…….. 72

Figure 5.33. Signal de réponse de la table et signal cible (El Centro)……72

Figure 5.34. L'erreur quadratique moyenne entre le signal de réponse et le signal cible du modèle N°1…… 74

Figure 5.35. La régression linéaire entre la réponse du RNA et le signal cible (Friuli) …….. 74

Figure 5.36. La régression linéaire entre la réponse du RNA et le signal cible (Kocaeli) …….. 75

Figure 5.37. La régression linéaire entre la réponse du RNA et le signal cible (El Centro) …….. 75

Figure 5.38. Signal de réponse et signal cible (El-Centro) …….. 75

Figure 5.39. Signal désiré, signal de réponse corrigé par PID seul et signal de réponse corrigé par PID+RNA (El Centro) de la table sans spécimen…….. 76

Figure 5.40. La régression linéaire entre la réponse du RNA et le signal cible (El Centro)…….. 77

Figure 5.41. Signal désiré, signal de réponse par PID seul et signal de réponse corrigé par PID+RNA (El Centro) de la table avec spécimen à 1DDL…….. 77

Figure 5.42. Signal de réponse et signal cible (Northridge) de la table avec spécimen à 1DDL sans correction baseline du signal d'excitation…… 78

Figure 5.43 : signal de Friuli (a) sans correction baseline et (b) avec correction baseline…….. 79

Figure 5.44 : signal de Friuli en déplacement (a) input du RNA en déplacement sans correction baseline, (b) output du RNA avant correction baseline et (c) output du RNA après correction baseline…….. 79

Figure 5.45. Signal de réponse et signal cible (El Centro) de la table+spécimen…….. 80

LISTE DES TABLEAUX

Tableau 2.1. Principales tables vibrantes dans le monde …… 8

Tableau 2.2. Spécifications de la table vibrante du laboratoire du CGS …… 9

Tableau 2.3. Différentes sources de distorsion du signal de commande et des exemples d'enregistrements dus à ces sources …… 11

Tableau 3.1. Exemples de fonctions de transfert a=f(n) et nomenclature Matlab …… 24

TABLE DES MATIERES

DEDICACES..... ii

REMERCIEMENTS iii

RESUMES..... iv

LISTE DES FIGURES vi

LISTE DES TABLEAUX ix

TABLE DES MATIERES x

Chapitre I : INTRODUCTION GENERALE

I.1. Généralités 1

I.2. Problématique 1

I.3. Motifs et objectifs 2

I.4. Méthodologie de l'étude 3

I.5. Plan de travail 4

Chapitre II : CONTROLE DE LA TABLE VIBRANTE

II.1. Introduction 5

II.2. Problèmes des tests sur table vibrante 10

II.3. Revue des travaux sur le contrôle et l'amélioration des performances des tables vibrantes..... 14

II.4. Contrôle actuel de la table vibrante du CGS 18

II.5. Nouveau contrôle : le Contrôle Intelligent 19

II.6. Conclusion 21

Chapitre III : Les Réseaux de Neurones Artificiels

III.1. Introduction 22

III.2. Présentation générale 22

III.2.1. Les réseaux de neurones biologiques..... 22

III.2.2. Les réseaux de neurones formels 23

III.3. Types de réseaux 25

III.4. Architecture du réseau 26

III.5. Apprentissage du réseau 28
III.6. Types d'apprentissage 29
III.7. Construction d'un réseau de neurones 29
III.8. Applications des RNA dans le Génie Civil 30
III.9. Conclusion 31

Chapitre IV : Contrôle de la table vibrante par la méthode neuronale

IV.1. Introduction 32
IV.2. Principe et méthodologie 32
IV.2.1. Principe 32
IV.2.2. Méthodologie 34
IV.3. Modélisation de la table vibrante 35
IV.3.1. Modèle mathématique de la table vibrante 36
IV.3.2. Modèle par Eléments Finis de la table vibrante 37
IV.3.3. Modèle sur Matlab de la table et de ses équipements 38
IV.3.3.1. Fonction de transfert de la servovalve 40
IV.3.3.2. Fonction de transfert du système servovalve + vérin 40
IV.3.3.3. Fonction de transfert du système servo-hydraulique 41
IV.3.3.4. Fonction de transfert totale de la table non chargée (sans spécimen)..... 41
IV.3.3.5. Fonction de transfert du spécimen testé (spécimen à 1DDl) 42
IV.3.3.6. Fonction de transfert totale de la table chargée (avec spécimen à 1DDL) 43
IV.4. Construction du réseau neuronal 43
IV.4.1. Constitution des bases de données du réseau 44
IV.4.1.1. Analyses des données 44
IV.4.1.2. Séparation des bases de données 44
IV.4.1.3. Mise en forme des données pour un réseau de neurones 44
IV.4.2. Choix du type de réseau 45
IV.4.3. choix du nombre de couches cachées et le nombre de neurones dans chaque couche du réseau 49
IV.4.4. Choix de la fonction de transfert 50
IV.4.5. Choix de l'algorithme d'apprentissage 50
IV.4.6. Validation 52

IV.5. Conclusion 53

Chapitre V : Performance du RNA comme système de contrôle

V.1. Introduction 54

V.2. Performance du RNA pour un système sans contrôleur 54

V.2.1. Description des modèles numériques utilisés 54

V.2.2. Evaluation de la performance du système de contrôle 56

V.2.2.1. Modèle N° 1 : système linéaire sans spécimen 57

V.2.2.2. Modèle N°2 : système non linéaire sans spécimen 59

V.2.2.3. Modèle N° 3 : système linéaire avec spécimen rigide linéaire 62

V.2.2.4. Modèle N° 4 : système non linéaire avec spécimen rigide linéaire 64

V.2.2.5. Modèle N° 5 : système non linéaire avec spécimen rigide non linéaire 66

V.2.2.6. Modèle N° 6 : système non linéaire avec spécimen flexible non linéaire 68

V.2.3. Essai d'application du système de contrôle par RNA 71

V.3. Performance du RNA pour un système avec contrôleur 73

V.3.1. Description des modèles numériques utilisés 73

V.3.2. Evaluation de la performance du système de contrôle 73

V.3.2.1. Modèle N° 1 : table sans spécimen 73

V.3.2.2. Modèle N°2 : table avec spécimen à 1DDL 76

V.3.3. Essai d'application du système de contrôle par RNA 80

V.4. Conclusion 81

Chapitre VI : Conclusion générale

Conclusion générale 82

REFERENCES BIBLIOGRAPHIQUES 84

ANNEXES

ANNEXE 1 : Les résultats de test de validation du RNA pour la prédiction du signal de commande 90

ANNEXE 2 : Article paru dans IZIIS-50 Conference Proceedings, Mai 2015 142

Chapitre I : INTRODUCTION GENERALE

I.1. Généralités

L'expérimentation, qui peut être définie comme le processus d'acquisition d'informations à travers des observations physiques et des mesures, a été et doit impérativement être un aspect essentiel de la recherche dans le domaine du génie parasismique (Earthquake Engineering Research). Conventionnellement, l'expérimental a été utilisé pour étudier des phénomènes fondamentaux qui ont attiré l'attention, ou n'ont pas été suffisamment compris, ou pour développer ou vérifier des modèles analytiques, les valider, ou de tester les performances de nouvelles conceptions, ou pour extraire des détails empiriques concernant des modes de comportement dont le modèle analytique est complexe. Mais de nos jours l'expérimentation demande d'aller plus loin avec plus de développement et des techniques plus sophistiquées. Dans le domaine du génie parasismique, celle-ci se base principalement sur des tests quasi-statiques cycliques, des tests pseudo-dynamiques et des tests sur table vibrante [43] ou des combinaisons de ces types de tests appelées essais hybrides. Il est clair que ces différents types de tests seront plus appropriés dans certaines circonstances que dans d'autres, en fonction du type de recherche qu'on veut mener et des résultats qu'on veut trouver. Par exemple, si la recherche nécessite l'étude du comportement total d'un système structural alors le test quasi-statique cyclique ou le PSD peuvent être appropriés. L'étude d'un comportement dynamique non linéaire peut être difficile surtout dans le cas d'un test à grandeur réelle, à cause de la difficulté de fournir suffisamment d'énergie à la structure pour qu'elle produise une réponse non linéaire. Ceci donne le choix entre un test sur table vibrante ou le test centrifuge car les deux donnent la possibilité de reproduire le chargement auquel le prototype sera soumis. Le choix entre ces deux types de tests sera alors dépendant de la facilité de fabrication du spécimen et les moyens disponibles. Pendant que les tests sur des structures réelles paraissent idéaux, c'est presque toujours impossible de les réaliser pour des raisons financières et pratiques. Le test sur table vibrante quant à lui présente un très bon compromis entre les modèles testés qui sont des modèles réduits ou des structures à l'échelle réelle adaptées aux dimensions et la capacité de la table vibrante et l'importance de la nature des forces qui les sollicitent qui peuvent être des mouvements sismiques réels [21].

Les tables vibrantes sont un outil critique dans la recherche du génie parasismique et une importante qualification pour les laboratoires de par le monde [17, 18]. Ce sont de larges plateformes rectangulaires à six (06) degrés de liberté (parfois moins) commandés par des vérins hydrauliques ou autres. Elles sont le moyen le plus efficace de soumettre des prototypes (spécimens) à des excitations dynamiques similaires aux séismes réels. Leur utilisation dans l'évaluation du comportement dynamique des structures du génie civil est effective depuis les années 60 et elle s'étend de manière importante dans la recherche sismique, puisqu'elles fournissent les meilleurs moyens d'exciter différentes structures, de les assujettir à des forces d'inertie représentatives des mouvements sismiques réels.

I.2. Problématique

L'objectif principal de la table vibrante est de reproduire des enregistrements temporels (accélérogrammes par exemple) sur une plateforme avec une précision raisonnable afin de conserver les forces d'inertie et les effets de l'amortissement réels.

Cela dit, la reproduction d'un accélérogramme (accélérogramme correspondant à un séisme donné) par un tel système complexe demeure imparfaite et nombreux travaux dans la littérature le montrent [4, 18, 19, 21, 22, 45, 62, 87, 95]. L'idéal serait que le système soit un seul corps parfaitement rigide afin qu'il suive exactement les mouvements imposés (signal de commande). Mais l'importance et la complexité de la table vibrante et ses équipements, leur interaction, l'interaction table-spécimen, les nombreuses flexibilités du système (par exemple la flexibilité de la plateforme, les flexibilités locales entre les appuis des vérins hydrauliques et le bloc de réaction, etc.) [21], les fortes non-linéarités internes du système, les limitations des systèmes servo-hydrauliques-vérins qui causent des temps de retard [70] font en sorte que la table soit un système dynamique à elle seule avec sa propre réponse au signal de commande et de ce fait, cet idéal ne sera jamais atteint [18], c'est-à-dire le mouvement sera distordu et donc exige l'utilisation d'un contrôleur robuste capable de forcer la plateforme à suivre un mouvement prescrit désiré (a target motion). Le contrôleur est un équipement électronique muni d'un algorithme digital qui fournit la commande appropriée aux servovalves dans le but de conduire la table selon la bonne trajectoire (le mouvement désiré) [62].

Aujourd'hui, l'un des challenges clé de l'ingénieur est de réaliser le test avec précision et de conduire la table à reproduire le plus fidèlement possible le mouvement désiré [60]. Car malgré les nombreux progrès dans les stratégies de contrôle, les signaux sont toujours simulés avec un certain degré de distorsion.

La présente étude se concentre sur le développement d'un outil de contrôle offline basé sur les réseaux de neurones artificiels (RNA) permettant l'amélioration du taux de fidélité de reproduction du mouvement de la table même quand le spécimen exhibe un comportement non linéaire.

Le contrôleur adapté est un contrôleur intelligent basé sur un réseau de neurones multicouche récurrent à rétropropagation, qui établit les relations entre les couples de données (input/output) obtenues par deux modèles de la table : un premier modèle simplifié en éléments finis de la table qui représente les composants mécaniques de la table et puis un second modèle simplifié mais plus détaillé de la table ainsi que tous ses équipements implémenté dans Matlab/Simulink.

La caractéristique la plus intéressante des RNA est la généralisation, donc la possibilité d'utiliser le même réseau avec de nouvelles données.

I.3. Motifs et objectifs

Etant donné la complexité du système tout entier, la problématique principale de la table vibrante est la fidélité dans la reproduction du signal désiré au niveau de la plateforme (à la base du spécimen) car celui-ci est fortement distordu.

Plusieurs méthodologies de contrôle ont été développées, comparées, validées dans le but d'améliorer au maximum la précision du mouvement de la table. Des résultats très satisfaisants ont été trouvés dans la littérature mais malheureusement, ils ne s'étendent principalement que sur les systèmes linéaires ou localement linéarisés dont les paramètres sont connus et ne changent pas avec le temps au cours du test [19, 87]. Dans ce cas les systèmes de contrôle linéaires peuvent être réglés avec un très haut degré de précision. Cela dit, ce n'est plus valable quand le système suit un régime non linéaire et ses paramètres changent [62]. Plusieurs travaux antérieurs montrent les non linéarités internes du système et les nombreuses interactions notamment l'interaction non linéaire table-spécimen qui

perturbent de manière importante le contrôle actuel des tables vibrantes. Afin de remédier aux imprécisions et aux faiblesses des systèmes de contrôle linéaires, des techniques de contrôle itératives (online ou offline) sont apparues et devenues les plus répandues et les plus utilisées.

Une procédure itérative online typique peut être décrite comme suit :

- Avant toute excitation du système, mesurer les paramètres du spécimen à partir de sa réponse à des vibrations libres. Ces paramètres sont un état initial considéré comme un point de référence du spécimen avant le test ;
- commencer le réglage du contrôleur dont les termes (gain, lead, notch filter) sont mis à zéro ; des vibrations aléatoires à faibles amplitudes sont réalisées sur la table et la FRF est mesurée. Puis les termes du contrôleur sont modifiés au fur et à mesure que le niveau d'excitation augmente. Cela dit, changer un terme peut avoir un effet bénéfique sur une certaine région de fréquences mais aussi un effet indésirable sur une autre région ;
- une fois qu'une réponse proche du gain unité est obtenue, le signal d'input peut être appliqué (à amplitude réduite : échelle d'amplitude). Le spécimen et les données mesurées sont analysés puis un autre signal d'input à des amplitudes plus grandes est appliqué, ainsi de suite… jusqu'à à ce que la performance désirée soit atteinte.

Néanmoins, le besoin de préserver le spécimen d'éventuelles fatigues prématurées a mené à des techniques de contrôle de la table sans chargement (sans spécimen) qui, pourtant change lors du chargement de la table et contourne le plus important des problèmes causés qui est l'interaction table-spécimen.

Pour éviter ce compromis, des techniques de contrôle offline ont été développées afin de réaliser ces opérations théoriquement jusqu'à obtention du signal désiré au niveau de la plateforme [90]. A partir de tout cela est venu le besoin grandissant de développer une méthode de contrôle capable de pallier toutes ces insuffisances.

A cet effet, un contrôle intelligent offline basé sur les réseaux de neurones artificiels, ayant fait ses preuves dans nombreuses applications industriels, sera développé. Une architecture adéquate du réseau de neurones sera définie puis utilisée sur une base de données d'apprentissage réalisée à partir d'un modèle numérique par éléments finis de la table vibrante puis un autre modèle implémenté dans Matlab. Ainsi, le réseau sera capable de prédire le signal de commande qui devrait être envoyé à la table afin de recueillir le signal désiré au niveau de la plateforme (en terme d'accélération, de vitesse et de déplacement) qui excitera réellement la base du spécimen.

I.4. Méthodologie de l'étude

La banque de données servant à l'apprentissage du réseau de neurones est construite à partir de deux modèles représentatifs de la table vibrante et de ses équipements. Un premier modèle par éléments finis de la table, représentant uniquement les composants mécaniques de la table (la plateforme, les vérins et le spécimen), a été réalisé par le logiciel SAP2000. Le second représentant tous les composants de la table (contrôleur online, servovalves, vérins, plateforme, spécimen et bloc de réaction), a été implémenté dans Matlab sous forme de fonctions de transfert. Ensuite, une série d'analyses dynamiques a été faite pour construire une banque de données, c'est-à-dire un ensemble de couples (input/output) qui serviront à l'apprentissage puis au test et à la validation du réseau.

L'input est le signal de réponse de la plateforme, un accélérogramme recueilli au CDG de la plateforme de la table, et l'output du réseau est le signal de commande, signal recueilli au niveau de l'extrémité fixe des vérins. Ce dernier est comparé au signal désiré et il devient possible de calculer le signal d'erreur à minimiser pour corriger le signal de commande.

Une fois l'apprentissage du réseau de neurones fait sur un signal donné, le réseau sera testé puis validé pour plusieurs signaux (accélérogrammes réels). Il sera alors capable de généraliser. De ce fait, pour n'importe quel autre signal présenté à lui, le réseau produira le signal de commande à envoyer afin atteindre le signal désiré sur la plateforme.

I.5. Plan de travail

Cette présente étude est organisée comme suit :

- Le chapitre I est une introduction générale il parle des tables vibrantes et leur principal problème de reproduction des signaux désirés. Les motifs et objectifs que vise l'étude ainsi que la méthodologie suivie ont été présentés dans cette même partie.
- Le chapitre II détaille les composants des tables vibrantes ainsi que les spécifications de la table étudiée qui est la table du CGS (Sebbala, Alger). Il présente les nombreux problèmes que posent les tests réalisés sur table vibrante, leurs sources, et les différents travaux dans le contrôle visant à pallier ces problèmes et à améliorer les performances des tables vibrantes. Il présente également le contrôle actuel de la table vibrante du CGS et le choix de l'améliorer par un contrôleur intelligent.
- Le chapitre III présente le modèle d'un RNA et son fonctionnement, les différents types et architectures que les RNA peuvent avoir, les différentes fonctions de transfert, les différents types d'apprentissage et les différentes étapes de construction d'un RNA. Il cite aussi leurs différentes applications dans le Génie Civil.
- Le chapitre IV cerne le vif du sujet qui est l'application du RNA comme système de contrôle de la table vibrante. Il présente le principe et la méthodologie utilisés pour l'élaboration de l'algorithme de contrôle par RNA, ainsi que les modèles de la table servant à la construction de la banque de données input/output pour l'apprentissage, le test et la validation du réseau. Il détaille également la procédure du choix de l'architecture du réseau utilisé, ainsi que de son algorithme d'apprentissage.
- Le chapitre V est celui où sont décrits les modèles numériques de la table vibrante et présente surtout les résultats de l'évaluation de la performance du RNA comme système de contrôle.
- Le chapitre VI est une conclusion générale qui résume l'objectif principal de ce travail de recherche ainsi que la démarche suivie pour l'atteindre. Elle annonce également les perspectives principales qui pourront être traitées dans un travail futur.

Chapitre II : CONTROLE DE LA TABLE VIBRANTE

II.1. Introduction

La table vibrante est une des techniques expérimentales qui suscite actuellement de plus en plus d'intérêt. Elle permet d'exciter des modèles de structures et d'éléments structuraux avec une large gamme de mouvements sismiques simulés dont des séismes réels reproduits. Utilisant les données enregistrées il est alors possible d'interpréter et d'étudier le comportement dynamique de la structure testée.

Généralement, une table vibrante se compose de trois (03) systèmes : un système mécanique, un système hydraulique et un système électronique.

- Le système mécanique comporte principalement :
- Des accéléromètres pour fournir les informations au système de contrôle et l'accélération des différents endroits de la plateforme ;
- Un spécimen linéaire/non linéaire attaché à la plateforme ;
- Une plateforme mobile généralement en métal ;
- Appuis verticaux et latéraux ;
- Un bloc de réaction ou masse de réaction construit en béton armé qui fournit une masse très rigide contre laquelle le vérin peut pousser. Idéalement, elle doit faire 30 à 50 fois la masse de la plateforme et de la masse maximale du spécimen.
- Sol adjacent.
- Le système hydraulique comporte :
- Des pompes hydrauliques qui fournissent la grande pression du flux d'huile afin de mettre en marche la table vibrante ;
- Des accumulateurs hydrauliques. Il existe plusieurs accumulateurs d'huile, le plus petit sert à réguler le flux d'huile entrant et sortant des servovalves des vérins ;
- Des servovalves. Ce sont des valves qui contrôlent le flux d'huile dans les vérins d'où le mouvement de la plateforme ;
- Des vérins servo-hydrauliques (ou électromécaniques) qui actionnent la plateforme. La plupart des tables vibrantes comportent des vérins servo-hydrauliques ; cependant de petites tables vibrantes ayant des vérins électrodynamiques sont généralement utilisées pour des tests de fatigue sur des petites structures ou éléments structuraux ;
- Le système électronique comporte :
- Un contrôleur ;
- Des capteurs de mouvements ;
- Un système d'acquisition de données.

Un schéma simplifié ainsi que des images réelles de la table vibrante du laboratoire du CGS sont présentés ci-dessous.

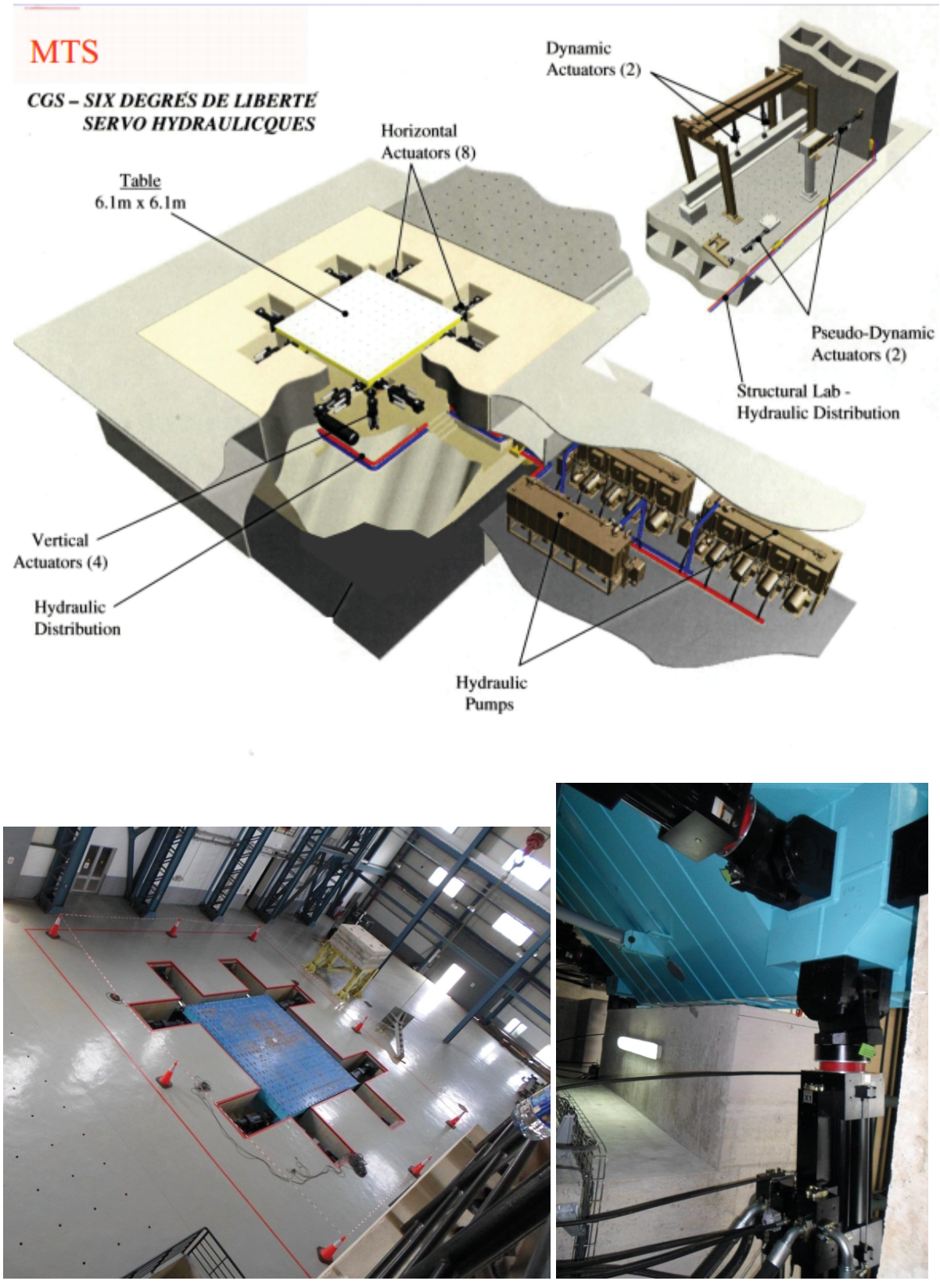


Figure.2.1. Schéma simplifié et photos réelles de la table vibrante du CGS.

Il existe quatre (04) principaux types de tables vibrantes dans la filière de recherche du génie parasismique. Ces types diffèrent de par le nombre d'axes possibles suivant lesquels le mouvement peut être contrôlé, allant de tables à seulement un (01) degré de liberté appelées « single-axis tables » (tables pouvant se mouvoir selon un seul axe) à des tables à six (06) degrés de liberté appelés « multi-axis tables » (pouvant être contrôlées suivant les six (06) axes, c'est-à-dire trois (03) axes de translation et trois (03) autres de rotation).

La table à un seul axe, mécaniquement la plus simple, excite le spécimen horizontalement. Pour plusieurs cas il est désirable d'exciter le spécimen suivant uniquement un seul axe car ceci simplifie considérablement la réalisation du test (le contrôle de la table) et l'interprétation des résultats. L'inconvénient principal de celle-ci est l'incapacité d'étudier des comportements plus complexes sous un chargement selon plus d'un axe. La table de six (06) DDL représente la plus polyvalente des tables qui permet un mouvement de la plateforme selon toutes les directions transversales et notamment les composantes rotationnelles. Elle permet d'étudier le comportement torsionnel d'une structure irrégulière en plan par exemple [21].

Le tableau qui suit présente les principales tables vibrantes à travers le monde.

Tableau 2.1. Principales tables vibrantes dans le monde.

Region	country	Location	Size(m)	Payload (metric tonnes)	N° DOF	Max freq (Hz)
Africa	South Africa	University of Witwatersrand	4x4	10	1	40
Asia	China	Guangzhou University	3x3	20	6	50
Asia	Singapore	Nanyang Technological University	3x3	10	1	50
Asia	India	IISc, Bangalore	1x1	0.5	6	50
Asia	Iran	Sharif University of Technology (SUT))	4x4	30	3	50
Asia	Japan	NIED 'E-Defence' Laboratory, Miki City	20x15	1200	6	50
Asia	Korea	Pusan National University	4x4	300	6	60
Asia	Taiwan	National Center for Research in Earthquake Engineering	5x5	50	6	50
Australia	Australia	University of Technology, Sydney	3x3	10	1	50
Europe	France	Commissariat à l'Energie Atomique et aux Energies Alternatives (CEA), AZALEE	6x6	100	6	50
Europe	France	FCBA Institut technologique	6x6	10	1	50
Europe	Greece	National Technical University of Athens	4x4	10	6	100
Europe	Italy	European Centre for Training & Research in Earthquake Engineering (EUCENTRE) – Shake Table	5.6x7	140	1	50
Europe	Turkey	Bogazici University	3x3	10	1	50
Europe	UK	University of Bristol (EERC)	3x3	17	6	100
USA	California	University of California at Berkeley	6.1x6.1	45	6	20
USA	California	University of California at San Diego	12.2x7.6	2000	1	20
USA	New York	University at Buffalo (State University of New York)	3.6x3.6	50	6	100

Les spécifications de la table vibrante du CGS détaillés sont données dans le tableau suivant.

Tableau 2.2. Spécifications de la table vibrante du laboratoire du CGS.

Patform	6.1mx6.1m
Payload	60t
Overtuning moment	120t
Torsion moment	90t
Active degrees of freedom	6
stroke	300mm
Max velocity	1100mm/s
Mac acceleration (20t)	1.7g
Max acceleration (payload)	1.0g
Frequency range	0.1-100Hz

La conception et la construction initiales de la table ont une importance significative sur son comportement final et sur les tests dynamiques qui y seront réalisés. Nombreux travaux visant à décrire les composantes de la table, sa conception, son fonctionnement et le déroulement de tests sur table vibrante peuvent être trouvés dans la littérature.

L'une des premières publications qui décrivent la conception de deux (02) tables vibrantes est celle de [13]. A partir de cette publication, nombreux travaux ont vu le jour notamment ceux de [17], [6] qui décrivent les principes généraux qui doivent être pris en compte dans la conception du système de la table vibrante. En particulier, ils montrent que pour atteindre la réponse désirée du système à un coût moindre, il est nécessaire de considérer la table comme un système dynamique.

Dans [61], les auteurs décrivent la conception, la fabrication et la construction de la table du NRCDP, Japon, et montrent les problèmes concernant la fabrication des appuis hydrostatiques et des servovalves.

Les auteurs de [73], [74] décrivent le développement de la table de Buffalo et particulièrement la conception de la plateforme elle-même.

Après la création de tables vibrantes plus grandes et plus puissantes, [54] montrent les améliorations nécessaires à la mécanique de la table et discutent comment améliorer les caractéristiques dynamiques du bloc de réaction en augmentant la masse de la fondation.

[96] décrivent l'amélioration de la performance de la table de l'université de British Columbia, résultant de l'ajout de quatre (04) vérins à la table. Des vérins ont été ajoutés à la table vibrante de Berkeley également (EERC, 1997).

Tous ces travaux, décrits et résumés par Crewe [21] montrent que les tables vibrantes sont opérationnelles depuis plus de 30 ans, que les ingénieurs et chercheurs élargissent les tables autant que leur champ de recherche dans le but d'améliorer le fonctionnement de celles-ci. Mais il existe depuis autant de travaux publiés visant à montrer les nombreux problèmes que rencontrent les tests dynamiques sur table vibrante.

II.2. Problèmes des tests sur table vibrante

L'arrangement des vérins crée trois (03) principaux problèmes du point de vue du contrôle du mouvement de la plateforme d'une table à six (06) DDL. D'abord, le fait qu'il y ait plus de vérins (huit (08) vérins) que de degrés de liberté (six (06) DDL), des techniques doivent être utilisées afin d'éviter un entrechoquement probable des vérins. Aussi, un système de contrôle a besoin de convertir le mouvement désiré de la plateforme en huit (08) différents signaux afin d'actionner les vérins. Enfin, dans le cas où le spécimen sera testé selon un nombre d'axes inférieur à six (06), le mouvement selon les autres axes sera empêché et cette restriction de mouvement et ce contrôle spécifique sur le mouvement de chaque vérins provient généralement du hardware control system. Cela dit, des recherches ont montré que le hardware control system ne peut pas toujours restreindre le mouvement des axes non sollicités convenablement, et dans ce cas un software control system additionnel doit être utilisé afin de compenser toute insuffisance dans le hardware.

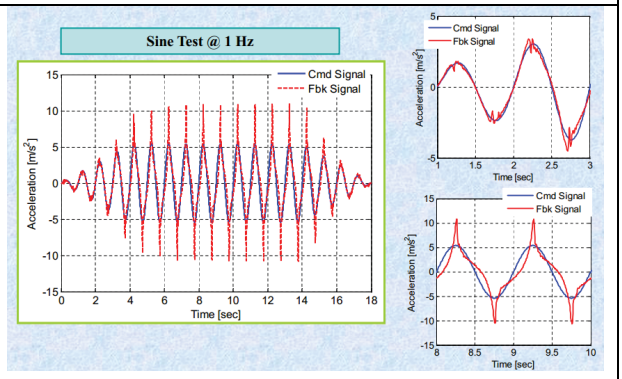
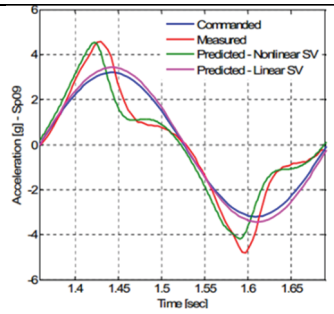
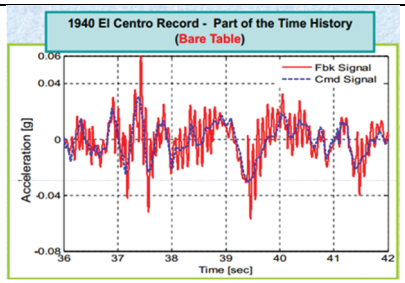
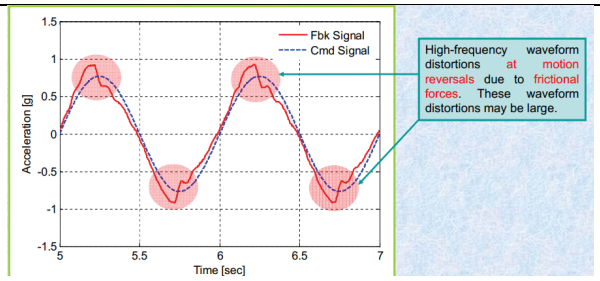
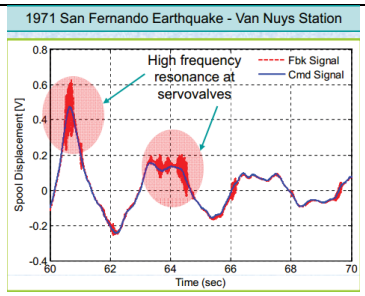
La conception de la table vibrante et les différents systèmes qui la constituent ont une grande influence sur son comportement général, dû à leurs diverses interactions. En effet, les paramètres dynamiques du système ne sont pas seulement déterminés par la plateforme de la table et ses vérins, mais aussi par les servovalves, la viscosité du fluide non linéaire, les sources dissipatives de frottement, ainsi que le spécimen testé. Aussi, les non linéarités du système, l'interaction table-spécimen, les nombreuses flexibilités dans le système, vont faire en sorte que le signal émis par la source va être modifié à maintes reprises par la table et ses équipements, jusqu'à ce qu'au niveau de la plateforme, le spécimen sera excité par un tout autre signal : c'est la distorsion de signal provoquée par la table elle-même.

Plusieurs chercheurs se sont intéressés à l'identification de ces nombreuses sources de distorsion, par exemple [45] ou encore plus récemment [20]. Ces derniers ont résumé les différentes sources de distorsion (différentes non linéarités contenues dans le système [18, 109, 110]) qui font obstacle à la précision du contrôle linéaire de la table, qu'ils disent interdépendantes, comme suit :

- Des sources hydrauliques :
    - Les servovalves qui contiennent des non linéarités internes;
    - La fréquence de résonance de la colonne d'huile dans les vérins ;
    - Le changement de pression (offre/demande) ;
    - Les limitations des débits et des forces des vérins ;
    - Le débit de fuite dans les servovalves et les vérins ;
- Des sources mécaniques :
    - Les forces dissipatives/de frottement ;
    - Les interactions linéaires ou non linéaires entre table-spécimen ;
    - La résonance entre l'étage pilote et le 3ème étage des servovalves ;

Des exemples de ces distorsions sont résumés dans le tableau suivant [Conte et al., 2010] :

Tableau 2.3. Différentes sources de distorsion du signal de commande et des exemples d'enregistrements dus à ces sources.

Distortion sources	
Example of signal distortion (from UNAM Shake table- with sine test at 1Hz):	
Linearized servovalve	
Oil column resonance	
Dissipative forces (friction):	
Resonance between the pilot stage and the 3rd stage of the servovalve:	

Il est à noter que d'après [20] la plus importante source de distorsion à éliminer reste l'interaction table-spécimen.

Quelques études analytiques anciennes détaillent l'effet de cette interaction, par exemple l'ancienne étude de [89] qui a montré qu'un spécimen flexible monté sur la table peut avoir un effet important sur les caractéristiques dynamiques du système entier, mais n'a pas pu démontrer que la capacité de la table à reproduire des accélérogrammes spécifiques peut être très affectée.

Les premiers résultats publiés montrant l'interaction table-spécimen ont été faits par [72]. Dans [14], les auteurs montrent que le tuning des quatre (04) tables vibrantes (Athens, ISMES, Bristol and LNEC) peut provoquer plusieurs résonances et instabilités une fois le spécimen monté et que la réponse désirée de la plateforme devient plus laborieuse à atteindre quand le spécimen est flexible.

Dans [11] la première analyse numérique des effets de l'interaction table-spécimen pouvant se produire lors des tests a été présentée. Les auteurs ont noté que l'interaction maximale se produisait avec une atténuation de la réponse de la plateforme à la fréquence propre du spécimen testé et ceci cause un réel inconvénient car l'un des buts des tests dynamiques est de solliciter le spécimen à sa fréquence propre afin de causer des dommages. Ils ont aussi prouvé que plus le rapport entre la masse du spécimen et la masse de la plateforme est grand plus l'interaction devient importante. Idéalement, une table vibrante doit avoir une fonction de transfert entre le signal de référence et le signal produit (une fonction de transfert totale) caractérisé par un gain unité et un déphasage nul à travers tout l'intervalle de fréquences opérationnelles sous table chargée, c'est-à-dire avec le spécimen monté sur la table [62, 63, 64]. L'influence de ce dernier sur la fonction de transfert de la table est montrée sur la figure qui suit [20]:

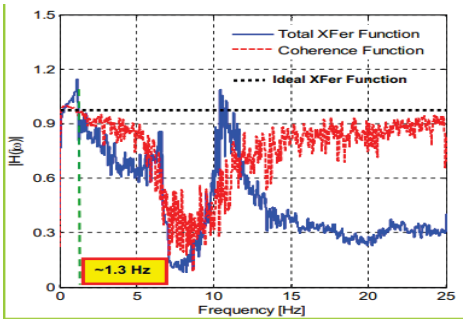


Figure 2.2. Effet de l'interaction table-spécimen sur la fonction de transfert du système

Afin de corriger ces distorsions, un système de contrôle robuste et fiable est alors plus que nécessaire, avec le principal but de reproduire au mieux le signal émis par la source. Conventionnellement, toutes les tables vibrantes comportent deux (02) types de systèmes distincts utilisés pour contrôler le mouvement de la table : un système de contrôle « hardware » et un autre « software » et les deux constituent ce qui s'appelle un système de contrôle.

- Le système hardware est un système électronique qui contrôle chaque vérin et y incorpore les boucles des feedbacks [98]. Ces boucles fonctionnent en soustrayant l'actuelle position du vérin de la position désirée puis utilisent ce nouveau signal pour actionner les servovalves. Ces boucles sont appelées « correction proportionnelle » car la magnitude de ce signal qui contrôle les servovalves est proportionnelle à la différence entre la position où le vérin doit être et sa position actuelle. La vitesse avec laquelle le vérin répond peut être variée en variant le « gain »

ou l'amplification qui est appliquée au signal de contrôle. Il est possible d'ajouter d'autre feedback, chacun avec son gain pour améliorer la précision de la réponse du vérin à travers un large intervalle de fréquences. Le processus d'ajustement des gains est appelé « tuning ».

- Le système software compense les imprécisions dans le tuning du système hardware. Généralement, le software qui contient la méthode de contrôle que le chercheur désire appliquée réalise le contrôle à l'aide d'un processus itératif. Cette procédure ne se réalise pas en temps réel ne peut alors compenser aucune réponse non linéaire pouvant se produire lors du test.

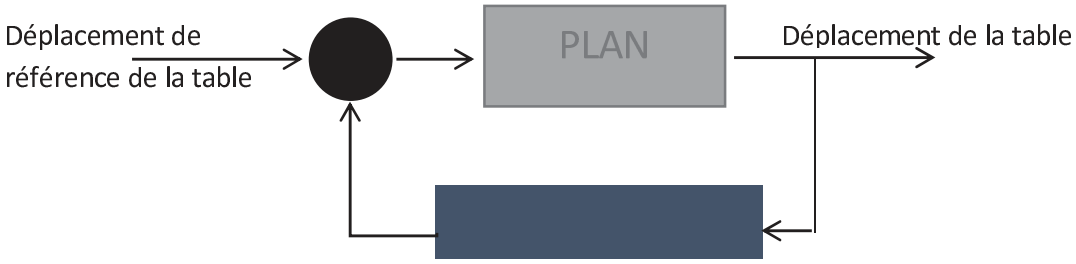


Figure 2.3. Schéma représentant le principe de contrôle d'une table vibrante.

Les premières techniques de contrôle de la réponse de la table vibrante, qui visent à supprimer les distorsions de signal, ont été décrites et comparées dans la fin des années 70 par [1]. Les auteurs de [82] ont comparé le mouvement atteint par la plateforme d'une table vibrante d'un DDL résultant d'un contrôle de réponse en déplacement et en accélération et ont conclu que contrôler la réponse de la plateforme en déplacement produisait la meilleure réponse. A partir de cela, la majorité des contrôles se font en mode déplacement.

Les systèmes de contrôle des tables vibrantes sont généralement basés sur des algorithmes de contrôle linéaires dans lesquels on suppose que le système à contrôler (appelé le plan) peut être modélisé par une série d'équations différentielles linéaires et les paramètres dynamiques du plan sont connus et inchangés durant les tests [87]. En effet, les contrôleurs linéaires fonctionnent en supposant que le plan sous contrôle modélisé par des équations linéaires non fonction du temps. Ces contrôleurs ont l'avantage d'être ajustés avec grande précision tant que les paramètres du plan restent linéaires. Ceci n'est pas vrai car le but même de ce genre d'essai est de pousser le spécimen à avoir un comportement non linéaire afin, et aussi à cause des multiples sources de distorsion citées auparavant, qui font obstacle à la précision de ce genre de contrôleur.

Afin de pallier l'important obstacle qu'est l'interaction table-spécimen et de minimiser l'erreur dans le mouvement de la plateforme, les constructeurs ont amélioré le matériel mais aussi les techniques de contrôle ont due beaucoup se développer. Plusieurs approches et stratégies de contrôle s'améliorent à travers des années de recherche comme par exemple les techniques itératives (iterative time history matching technique or transfer function iterative algorithm) [85]. Ces techniques enregistrent le mouvement de la plateforme atteint durant un test sismique puis corrigent le signal de commande de telle sorte que le mouvement de la plateforme durant le test suivant soit plus proche du mouvement désiré. Le software control system réalise cette procédure itérative en trois (03) phases [21]:

- envoyer un signal de commande à la table via le hardware control system ;

- enregistrer le mouvement de la plateforme ;
- et calculer le nouveau signal de commande.

Ces trois (03) phases constituent une itération et ces itérations continuent jusqu'à satisfaction, c'est-à-dire, jusqu'à obtention d'un mouvement très proche du désiré (erreur inférieure à une erreur tolérée).

Ces itérations peuvent se faire sur un signal de commande à amplitudes réelles sans spécimen ; ce dernier est monté sur la plateforme puis testé avec la dernière itération. Cependant, une erreur additionnelle dans la reproduction du signal apparaît alors due à l'interaction table-spécimen qui ne disparait pas. En effet, des expériences comparatives entre les performances des tables vibrantes ont montré qu'il fallait considérer la table et le spécimen comme étant un seul et même système dont la fréquence varie en fonction de la fréquence du spécimen [21]. Elles peuvent alors se faire sur un signal de commande à amplitudes réduites avec le spécimen attaché à la plateforme, et à la dernière itération, le signal est amplifié pour le test. L'inconvénient est l'apparition d'erreur dans la reproduction du signal aussi, due au fait que le spécimen se comporte quasi-linéairement durant les itérations à basses amplitudes, mais ce n'est pas le cas lors du test où les amplitudes sont élevées. Aussi, il a été prouvé que le niveau de fidélité de reproduction d'un signal atteint pour une amplitude donnée du signal de commande ne peut être maintenu en reproduisant le même signal à des amplitudes différentes.

Par ailleurs, le risque de dommages prématurés du spécimen est élevé, il sera sollicité par des cycles de mouvements de faibles amplitudes mais qui ils pourront causer un faible niveau de dommage par fatigue [90].

Les procédures itératives, fonctionnant bien avec les non linéarités récurrentes (comme celles des servovalves), ne suppriment pas complètement l'interaction table-spécimen pour autant et peuvent endommager le spécimen avant le test. Il demeure alors un compromis entre la précision dans le signal reproduit par la table (RMS minimal) et le risque d'endommagement prématuré du spécimen lors de mouvements forts.

II.3. Revue des travaux sur le contrôle et l'amélioration des performances des tables vibrantes

En 1995, B. Widrow et al., [10] utilisent un contrôleur en temps réel qui s'adapte aux changements des caractéristiques dynamiques au cours du temps, c'est le contrôle adaptif inverse (AIC). Il couvre une série de techniques qui fournissent une approche systématique pour un ajustement automatique des paramètres du contrôleur en temps réel, dans le but d'atteindre ou de maintenir un niveau acceptable (désiré) de contrôle lorsque les paramètres dynamiques du plan (du système) sont inconnus et/ou variables dans le temps.

Avec Plett [98], ils proposent un contrôle optimal des perturbations présentes dans le plan dynamique en utilisant un circuit spécial : ils obtiennent les perturbations à l'output du système, le filtrent puis le rajoutent au signal de commande et le renvoient en input.

En 1996, Filiatrault et al., [29] comparent cinq (05) techniques différentes utilisées pour tester un spécimen à deux (02) DDL sur une table à un DDL et montrent qu'une technique de contrôle itérative

offline ne peut entièrement contrôler la table et concluent que le développement d'un contrôleur en temps réel est nécessaire pour contrôler l'interaction non linéaire entre la table et le spécimen.

En 1998, Xie et al., [103] affirment que la différence entre les enregistrements recueillis lors des tests et ceux des simulations peut être due à deux (02) imperfections lors de la modélisation du processus réel : des différences paramétriques (i.e, la structure du modèle est identique au processus réel mais avec des paramètres différents) et des différences dans la structure du modèle (i.e, structure différente de celle du processus réel). Plusieurs méthodes ont été développées afin d'éliminer ces imperfections modèle/processus. Les auteurs utilisent l'AGMC (Adaptive Generic Model Control) comme algorithme de contrôle dans lequel un modèle non linéaire du processus peut être intégré dans la stratégie de contrôle. Ils développent deux (02) schémas: le premier est basé sur une estimation online des paramètres variant dans le temps du modèle non linéaire du GMC utilisant un filtre dit Strong Tracking Filter (STF). Le second schéma consiste en l'implémentation d'un concept appelé Input Equivalent Disturbances (IED) dans lequel les imperfections modèle/processus sont localisées. Ils montrent alors que l'AGMC peut maîtriser alors l'influence de la variation des paramètres au cours du temps mais aussi les perturbations non mesurables, et ce, efficacement.

Dans la même année, Crewe et al., [22] utilisent un système de contrôle des feedback et des algorithmes de contrôle qui itérativement, règlent les inputs de référence pour minimiser l'erreur.

En 1998, Spencer et Yang [85] utilisent une technique d'itération basée sur les fonctions de transfert pour régler les inputs de commande. Ces tests ont été réalisés sur une table non chargée.

En 1995, Newell et al., [57] développent un contrôleur basé sur le filtre de Kalman qui utilise une trajectoire de référence optimale obtenue en minimisant l'erreur. L'optimisation incorpore une réalisation non linéaire de la dynamique de la table. Le modèle non linéaire du système est linéarisé par rapport à la trajectoire de référence et le système obtenu est utilisé pour développer des gains de feedbacks variant dans le temps.

En 1999, Kuehn et al., [44] développent un contrôleur hautement performant et peu coûteux, utilisant une approximation linéaire du système dynamique, afin d'améliorer la performance de la table. Le contrôleur utilisé est un feedback/feedforward RHC (Receding Horizon Control or Model Preditive Control). Les résultats expérimentaux sont meilleurs en utilisant ce contrôleur à la place d'un contrôleur traditionnel (LQR contrôleur).

En 2000, Nowak et al., [59] disent qu'un contrôle précis de la table est un contrôleur de type closed-loop en temps réel (qui doit être suffisamment robuste pour faire face aux problèmes de précision et de stabilité même en présence d'importantes sources de perturbations) utilisé avec un processus de compensation itératif offline. Il crée une erreur minimale (cycle d'itérations court) mais afin d'atteindre un appariement parfait, une importante énergie d'excitation est nécessaire, les spécimens sont alors incapables de résister à ces excitations répétées sans s'endommager. Le processus itératif offline évite cela. Pour compenser les variations de fréquence du système pendant le test, la méthode de compensation utilisée est l'Adaptive Inverse Control (AIC) où un filtre basé sur des algorithmes de minimisation de l'erreur est automatiquement appliqué (inverse de la réponse fréquentielle).

Dans la même année, Igarashi et el., [38] développent une méthode expérimentale des sous-structures de la table vibrante dans laquelle une partie de la structure de la table est testée pendant que la

réponse du reste de la structure est numériquement calculée. Ces tests ont été développés sur une série de tables vibrantes hydrauliques et un DSP-based contrôleur.

Conte et Trombetti, [19] et [93] développent un modèle analytique linéaire d'une table vibrante uni-axiale utilisant une théorie de contrôle linéaire. Le modèle contient des gains proportionnels-dérivatifs-intégraux (PID), feedforward et de pression différentielle (ΔP). Leur analyse, basée sur une fonction de transfert totale de la table entière, montre l'importante interaction dynamique entre la colonne d'huile dans le vérin et le spécimen.

En 2001, Stoten et Gomez [87] proposent un contrôle adaptif utilisant l'algorithme MCS (Minimal Control Synthesis). Ce dernier améliore considérablement le contrôle des tables vibrantes en utilisant la stratégie de contrôle du déplacement pour des fréquences basses à moyennes. L'algorithme MCS, déjà utilisé pour contrôler chacun des huit (08) vérins de la table vibrante par Stoten (1993), fournit des moyens de contourner l'effet des variations des paramètres du système au cours du temps et les perturbations extérieures autant que les régimes non linéaires.

En 2004, Thoen et Laplace [90] présentent un modèle numérique d'une table vibrante de taille moyenne et ses sous-systèmes. Le modèle contient le logiciel de contrôle en temps réel de la table dans la simulation (le contrôleur de la table Controller GUI (MTS 469D) est lié au modèle Simulink par User Program Protocol). Ceci éloigne toute incertitude dans la modélisation du contrôleur et fournit à l'opérateur la même familiarité avec la simulation qu'avec le contrôleur du système réel.

En 2005, Chase et el., [15] présentent le problème du contrôle comme un compromis entre les gains des feedback et la précision de la reproduction des mouvements et développent une méthode de contrôle qui empêche les pics d'accélération et la sursaturation de la vitesse en modifiant l'accélérogramme ou le signal de référence conformément aux paramètres de la table une fois opérationnelle (limitation de la vitesse de l'huile due à la limitation du débit à travers les servovalves).

En 2008, Sivaselvan et el., [83] utilisent un contrôle de la force d'inertie au lieu d'un contrôle du feedback de la force, dans lequel un ressort a été intentionnellement introduit entre le vérin et la plateforme et une compensation du feedforward en déplacement est utilisée. Cette méthode de contrôle de la force a été utilisée avec succès dans des simulations dynamiques hybrides en temps-réel.

En 2008, Ozcelik et al., [63] et [64] construisent un modèle virtuel de la table vibrante UCSD-NEES et comparent les data réels et simulés et montrent la distorsion du signal du modèle et les mêmes gains du contrôleur que ceux utilisés dans le système réel. Ils affirment que la distorsion du signal est due à la non-linéarité du débit à travers l'orifice des servovalves. Alors, ils développent un modèle plus élaboré afin d'améliorer les performances du contrôleur existant (MTS 469D) en se basant sur une connaissance plus approfondie des caractéristiques physiques de la table.

Dans la même année, Airouche et al., [4] développent deux (02) modèles Simulink de la table uni-axiale EUCENTRE-TREESLab, l'un représente un système simplifié servovalve-vérin linéarisé, et le second représente un système non linéaire réaliste et détaillé. En 2014, ils développent aussi un modèle mathématique pour les différents composants de la table vibrante du CGS.

Traditionnellement, le bon fonctionnement de la table vibrante se fait grâce à des contrôleurs linéaires, qui sont conçus pour contrôler des systèmes linéaires. Généralement un vérin hydraulique est contrôlé

en mode de déplacement avec un contrôleur de type PID, sachant que les déplacements de référence sont à priori déterminés en intégrant l'accélérogramme deux (02) fois. Cependant, le PID a montré plusieurs faiblesses devant la complexité du système, les interactions de ses composants et les fortes non linéarités comme l'affirment [5].

Le PID, suscitant toujours un grand intérêt malgré ses limitations pousse Nakata en 2010 [55] à ajouter au contrôle du déplacement des feedback d'accélération et/ou de vitesse car, généralement, le contrôle du déplacement produit une faible précision dans la reproduction d'accélérations. En effet, utiliser des feedback d'accélération et de déplacement limiterait les pics d'accélération (apparus lors des mouvements forts quand la sursaturation de vitesse se produit).

En 2012, O'Hagan et Ma [60] essaient d'améliorer encore le PID. Ils font une estimation expérimentale des paramètres (gains) du PID contrôleur, pour améliorer la performance de ce type de contrôleur sur des tables vibrantes uni-axiales, et leur étude a montré cependant que des gains optimaux du contrôleur ne sont pas suffisants pour assurer l'achèvement du mouvement désiré et que le signal de commande doit passer par une mise à jour itérative.

Une stratégie de contrôle de l'accélération a été proposée par Stehman et Nakata [86] à la place du contrôle du déplacement : c'est une nouvelle approche utilisant uniquement les feedback d'accélération sans les feedback du déplacement pour reproduire précisément un accélérogramme de référence, mais en ajoutant un feedback en terme de force dans le but de fournir la stabilité du contrôleur.

Récemment, Shen et al., [81] proposent un contrôle hybride combinant une compensation offline et un AIC. Des études comparatives avec différentes stratégies de contrôle ont montré que ce contrôle proposé une meilleure convergence et une meilleure précision dans la TWR (Time Waveform Replication) qu'un autre contrôleur traditionnel seul (PID par exemple).

En 2013, Yang et al., [104] développent un contrôle non linéaire basé sur l'algorithme SMC (Sliding Mode Control). Ils utilisent la fonction de type Lyapunov pour conduire l'erreur asymptotiquement au zéro, ce qui mène à la performance désirée. Cet algorithme converge même le modèle présente des incertitudes et/ou des perturbations ainsi que les non linéarités.

Quand un modèle non linéaire est indispensable, la tâche d'obtenir un modèle hautement fidèle au processus réel devient plus difficile pour des processus non linéaires.

Un contrôleur linéaire, des perturbations non estimées de manière exacte, une linéarisation des composantes de la table, la complexité du système tout entier et ses interactions non prise en compte de manière explicite, laissent encore plusieurs raisons de chercher d'autres techniques de contrôle plus avancées.

Récemment, les réseaux de neurones artificiels (RNA) sont devenus un outil très intéressant dans la construction de modèles de systèmes non linéaires complexes, grâce à leur capacité intrinsèque d'apprendre, de généraliser et d'approximer n'importe quelle fonction non linéaire. Un grand nombre de systèmes de contrôle et d'identification basé sur les RNA ont été présentés dans la littérature, on peut en citer quelque uns : identification de système non linéaire utilisant les RNA, proposée par Chen et al., [16], identification et contrôle de systèmes dynamiques utilisant les RNA, proposée par Narenda [56], une investigation des RNA dans les systèmes de contrôle par Fabri et Kadirkamanathan [27], les

RNA dynamiques pour un contrôle prédictif de systèmes non linéaires par Liu et al., [51], et nombreux autres articles sur le contrôle en utilisant les RNA.

Les algorithmes de contrôle prédictif sont des plus adéquats pour l'utilisation des réseaux de neurones. En effet, la plupart des algorithmes non linéaires impliquent la minimisation d'une fonction coût en utilisant des méthodes de calcul pour obtenir les commandes optimales qui seront appliquées au processus. Cela dit, l'implémentation des algorithmes non linéaires de contrôle prédictif devient très difficile dans le contrôle réel car la minimisation de la fonction coût doit converger au moins vers une solution quasi-optimale et les opérations engendrées doivent être réalisées dans un temps très court (correspondant au pas de temps).

En 2002, Lazar et Pastravanu [47] analysent un contrôleur non linéaire prédictif qui élimine les obstacles les plus importants que connait l'implémentation du NMPC (Nonlinear Model Predicitve Controller) en développant un modèle non linéaire, en concevant un prédicteur neuronal et en fournissant une solution rapide et fiable pour les algorithmes de contrôle.

Un modèle de contrôleur prédictif linéaire et non linéaire a été développé en 2008 par Tricaud et Chen [92] dans lequel ils utilisent un solveur (RIOTS_95 dans Matlab) comme outil permettant la suppression de nombreux problèmes causés par le MPC et le NMPC.

En 2012, Mahajerin [52] développe un système d'identification et de contrôle de processus dépendant du temps utilisant des réseaux de neurones récurrents.

II.4. Contrôle actuel de la table vibrante du CGS

Dans le but d'améliorer la stabilité des systèmes de contrôle de la table vibrante, le Three Variable Controller (TVC) est largement employé, connu aussi sous le nom de « state variable controller ». C'est une partie intégrale des logiciels du MTS Control 469D.

La table vibrante du CGS est munie de ce type de contrôleur. Les fonctions de transfert entre tous les signaux d'input et d'output du TVC sont fournies par MTS Systems Corporation [90].

Le TVC contient des équipements additionnels qui lui permettent de compenser les distorsions des systèmes linéaires et non linéaires pour des signaux de commande (exemple : amplitude/phase control, AIC, OLI and notch filters). L'AIC par exemple est une technique de compensation de la réponse fréquentielle : il calcule l'erreur entre le profile désiré et actuel et applique un filtre (linéaire) de compensation pour réduire l'erreur après quelques itérations. L'algorithme OLI quant à lui vise à minimiser l'erreur entre la réponse désirée et actuelle quand celle-ci est significative. OLI compense le signal d'erreur en appliquant un filtre qui représente l'inverse de la fonction de transfert du système [28]. L'AIC et OLI sont un moyen très efficace pour fournir la stabilité nécessaire au système et la précision dans la reproduction des signaux en anticipant les perturbations au bout de quelques itérations.

Le TVC peut être ajusté de telle sorte à fonctionner simultanément en mode de déplacement, vitesse et accélération. Le processus d'ajustement de plusieurs paramètres de contrôle (par exemple, feedback et feedforward gains) et de pré conditionnement du mouvement d'entrée (input motion) pour un signal de reproduction optimal du système de table vibrante est appelé le réglage de la table (tuning).

Généralement, il se fait en trois (03) étapes (appelé aussi le three-step process) :

- Etape 1 : la première étape utilise un processus itératif dans lequel les paramètres de contrôle du TVC sont (manuellement) ajustés itérativement à petits pas pendant que la table chargée (avec spécimen) est en mouvement (cette étape se réalise sous des inputs en accélérations représentant un bruit blanc) ;
- Etape 2 : la deuxième étape consiste à obtenir une estimation du modèle inverse du plan. Ce modèle inverse est obtenu en utilisant la technique de l'AIC control [90]. La qualité du modèle inverse ainsi estimé dépend de l'amplitude du bruit blanc et des non linéarités présentes dans le système (AIC seul ne peut compenser les non linéarités si elles existent);
- Etape 3 : la troisième étape utilise la technique itérative online (OLI). En principe, OLI peut corriger n'importe quel déficit persistant après le réglage du TVC et plus encore, elle peut compenser des sources (linéaires ou non linéaires) de distorsion du signal dans le système (frottement, servovalves, interaction table-spécimen, etc.). la technique OLI est réalisée avec un mouvement cible plus faible pour éviter les dommages que pourra subir le spécimen.

Cela dit, tous ces processus de réglage sont laborieux et les résultats dépendent fortement des capacités et du niveau d'expérience de l'opérateur.

II.5. Nouveau contrôle : le Contrôle Intelligent

Auparavant, le contrôle traditionnel (impliquant le classique feedback control par exemple) rencontrait multiples difficultés. D'abord, la conception et l'analyse des systèmes de contrôle traditionnels se basaient sur leur modèle mathématique précis qui est très difficile à atteindre contenu de la complexité des systèmes, le degré élevé de non-linéarité, etc. Aussi, les hypothèses émises avant l'étude et la modélisation du système de contrôle qui étaient difficiles à respecter dans la pratique. Finalement, dans le but d'augmenter les performances du contrôle, la complexité du système de contrôle augmente également et leur fiabilité diminue alors.

Le degré d'aptitude du système de contrôle à faire face à ces multiples difficultés dépend du niveau d'intelligence du système. Pour ces raisons, une large utilisation du contrôle automatique dans la production industrielle a connu un grand succès. Ce dernier atteint largement par l'automatisation du contrôle et la supervision des tâches. Les systèmes automatisés peuvent ainsi réaliser leurs tâches de façon meilleure que l'homme. Malgré ce franc succès, il est cru que cette automatisation des systèmes peut être améliorée par l'implémentation d'équipements dits « human-like » comme la capacité d'apprentissage, d'adaptation, de raisonnement et d'abstraction. Ce n'est cependant pas évident de concevoir des systèmes avec de tels équipements.

De ce fait, il y a eu un grand intérêt pour la recherche dans le domaine de l'Intelligence Artificielle. Le Professeur Lu a écrit un livre qui résume quelques idées qui pourraient rendre meilleur le contrôle industriel et montre que la solution réside dans la combinaison des techniques de contrôle traditionnel des modèles dynamiques et le contrôle avec des approches nouvelles. Et c'est ce type de contrôle qu'est appelé « contrôle intelligent ».

Le terme « Intelligent Control » a été utilisé initialement et officiellement en 1967 par Leondes et a été utilisé de plusieurs façons, la plus adéquate étant celle où l'IC doit impliquer l'intelligence et la théorie de contrôle. Il doit être basé sur une importante tentative de comprendre et de copier le phénomène

qui a toujours été nommé « intelligence », qui signifie cette aptitude de généralisation, d'adaptation et de flexibilité qu'ont les hommes et de l'implémenter dans la théorie de contrôle dans le plus large domaine d'utilisation possible.

C'est un challenge dans les communautés du contrôle et des ordinateurs que d'explorer de nouvelles stratégies et philosophies de contrôle pour des processus industriels complexes. Les conditions d'établir cette nouvelle discipline qu'est le contrôle intelligent ont grandi graduellement. L'IEEE First Symposium sur le contrôle intelligent a eu lieu à Troy, New York en 1985. Durant ce meeting, les principes de l'IC comme étant une discipline et la structure d'un système de contrôle intelligent ont été discutés. Après ce premier meeting, l'IEEE Technical Committee sur le contrôle intelligent a été organisé dans l'IEEE Control System Society afin de discuter de la définition de ce terme (IC). Le second meeting a été la première conférence internationale sur l'IC dans laquelle plus de 150 participants venus des USA, d'Europe, de Japon, de Chine et autre pays y étaient. Ceci a prouvé l'importance grandissante de cette nouvelle discipline. A. Meystel, G.N Saridis, J.Y.S Luh, H.E Stephanou et A.H Levis ont réalisé d'excellents travaux pour le développement de l'IC. Des milliers de chercheurs, parmi lesquels des professeurs, des étudiants, des ingénieurs et des chercheurs scientifiques de différentes professions se sont engagés dans des travaux de recherche dans le domaine de l'IC. Plusieurs publications sur l'IC ainsi que plusieurs livres ont vu le jour, comme le livre intitulé « Intelligent Control » écrit par Cai, publié en 1990, « Intelligent Control : aspects of fuzzy logic and Neural Nets » écrit par Harris et al., publié en 1993 et beaucoup d'autres ne cessent de naitre jusqu'à ce jour.

Toutes ces stratégies développées par ces pionniers de l'IC montre un but commun de concevoir un système de contrôle robuste et adaptif (learning control system) dans un environnement certain ou incertain avec des connaissances (mathématiques) très limitée des principes du processus et des informations imprécises et/ou incomplètes.

Il est cependant dur de définir de manière exacte et unique des systèmes intelligents, depuis que l'intelligence avait plusieurs niveaux, et plusieurs sens conceptuels. Cela dit, généralement, un système intelligent est définit comme une machine avec des comportements semblables à ceux de l'homme et doit être capable de fonctionner efficacement dans un environnement déterministe, stochastique ou incertain. Aussi, la caractéristique la plus importante d'un système intelligent c'est l'auto-apprentissage et l'adaptation, mesurée par la performance de « généralisation ». Un principe important et pratique : c'est qu'une bonne généralisation est une condition nécessaire dans un système intelligent. Généralement, elle dépend de l'architecture du système, ses paramètres, l'algorithme d'apprentissage et la différence entre l'environnement d'apprentissage et l'environnement d'exécution.

L'IC est l'un des plus récents et importants domaines dans lesquels les machines sont utilisées pour simuler l'intelligence humaine. Les premiers systèmes de contrôle intelligent appliquaient quelques méthodes primaires et se développaient très lentement. Dans les récentes années, l'IC a émergé rapidement dans le courant intellectuel de l'IA. La théorie et les techniques du contrôle moderne ont été rapidement développées contenues des exigences du domaine de l'industrie notamment ainsi que les nombreuses investigations réalisées et ont été appliqués avec succès dans des systèmes d'identification, d'estimation, d'optimisation, et dans des systèmes linéaires particulièrement. Ces avancements ont fourni aux ingénieurs de contrôle plus de précision, plus d'outils mathématiques pour la conception et l'analyse des systèmes de contrôle. Par-dessus ceci, il y a eu le progrès étonnant

dans la science de l'ordinateur et dans la technologie du matériel en général. Il a été reconnu par la communauté industrielle que les ordinateurs d'aujourd'hui ont une très large mémoire, une très grande vitesse de performance, des coûts relativement faibles, et un environnement d'interface homme-machine très sophistiqué. De ce fait, les ordinateurs et les micro-processeurs ont été largement utilisés comme des outils basiques dans le contrôle industriel. En effet, le principal challenge aujourd'hui n'est pas le manque de fiabilité dans les ordinateurs et leurs logiciels pour supporter le contrôle sophistiqué, mais le manque de logiciels réalistes et possibles applicables au contrôle [106].

En plus de la technologie du contrôle moderne, l'application des systèmes intelligents et la combinaison du contrôle moderne avec les systèmes intelligents a été reconnue comme l'un des plus importants courants de recherche dans le développement d'une génération nouvelle du contrôle dans l'industrie. De récentes recherches faites sur l'IA incluant le savoir de l'ingénierie, la reconnaissance de formes, la logique floue, et les réseaux de neurones artificiels a fourni de grandes opportunités à la résolution de problèmes complexes de contrôle. Certes, l'IC n'a pas été conçu de manière à représenter une discipline systématique dans les aspects théoriques, mais d'un point de vue pratique, il a été reconnu par chacune des communautés du contrôle et de l'intelligence artificielle comme une variété d'approches comme la logique floue et les réseaux de neurones et pouvant être appliqué efficacement dans la résolution de nombreux problèmes de contrôle.

Actuellement, l'IC connait un énorme progrès et est de plus en plus adopté suite à l'énorme chance que connaissent les chercheurs d'aujourd'hui qu'est la puissance des ordinateurs actuels, leur performance et leur développement incessant. En effet, afin de mener à bien une technique d'intelligence artificielle, il suffit d'avoir de l'intelligence et un artifice : l'ordinateur est un excellent artifice de calcul qui démontre parfaitement bien l'intelligence.

II.6. Conclusion

L'ajout d'un contrôleur offline peut augmenter considérablement la performance de la table et limiter la procédure itérative online qui peut être laborieuse et surtout qui peut pré-endommager le spécimen. De ce fait, un contrôleur basé sur les RNA s'avère un moyen très efficace de prédire le signal de commande qui, une fois envoyé par la source, mènera la table selon le mouvement désiré.

Chapitre III : Les Réseaux de Neurones Artificiels

III.1. Introduction

Un système (ou machine) intelligente doit être capable de prendre une décision raisonnable (ou satisfaisante) ou d'agir dans un environnement variable ou incertain en terme de capacités de traitement d'informations en parallèle, d'apprentissage, de généralisation et de créativité. Avec les récents développements dans les technologies de l'ordinateur et de la recherche dans le domaine des algorithmes et architectures des systèmes neurologiques, la technologie des réseaux de neurones artificiels a été appliquée avec succès dans la reconnaissance des formes, le traitement de l'image, le traitement de texte, la robotique, la commande de processus, les systèmes de contrôle, etc.

L'étude des réseaux de neurones artificiels (RNA) a fait du chemin depuis que McCulloch et pitts ont établi le modèle cerveau dans leur publication. Les RNA ont été reconnus comme étant l'un des plus puissants outils dans le développement des systèmes intelligents, particulièrement dans la classification, le traitement de signal (filtrer les bruits dans un signal), mais aussi, la modélisation des systèmes non-linéaires (un modèle de RNA peut être établi à travers un apprentissage adaptif basé sur les données du système : ses inputs et ses outputs), le contrôle adaptif et l'apprentissage, le contrôle de systèmes dont la connaissance reste incomplète ou imprécise ou décrite avec la présence de multiples incertitudes.

Les contrôleurs, basés sur les RNA sont alors dotés de plusieurs propriétés importantes telles que la rapidité de l'exécution, la robustesse dans le contrôle et la forte adaptabilité de leurs applications.

Les aspects les plus importants dans le développement d'un RNA sont de concevoir leur architecture (appelée topologie), l'algorithme d'apprentissage et la sélection des exemples d'apprentissage.

III.2. Présentation générale

III.2.1. Les réseaux de neurones biologiques

Les réseaux de neurones biologiques réalisent facilement un certain nombre d'applications telles que la reconnaissance de formes, le traitement du signal, l'apprentissage, la mémorisation et surtout la généralisation. Ces applications sont pourtant, malgré tous les efforts déployés en algorithmique et en intelligence artificielle, à la limite des possibilités actuelles.

C'est à partir de l'hypothèse que le comportement intelligent émerge de la structure complexe et du comportement actif des éléments de base du cerveau que les réseaux de neurones artificiels se sont développés.
L'utilisation des réseaux de neurones passe d'abord par une compréhension du principe de fonctionnement du constituant élémentaire des réseaux de neurones qui est la cellule nerveuse. Elle est composée de trois (03) parties : le corps cellulaire les dendrites et les synapses comme montré sur la figure suivante :

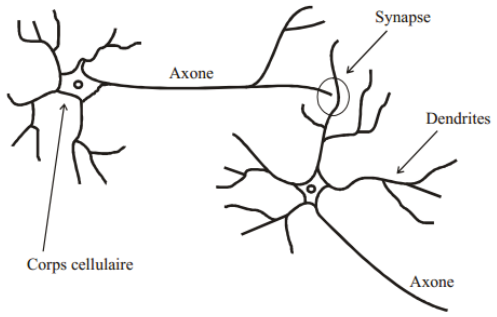


Figure 3.1. Cellule nerveuse et ses composantes.

Le corps cellulaire joue le rôle d'un intégrateur. Les dendrites quant à elles forment un maillage et font fonction de récepteurs principaux. Elles permettent d'acheminer vers le corps du neurone les signaux électriques provenant d'autres neurones. L'axone est la fibre nerveuse qui transporte les charges électriques (signal) émises par le neurone. Il se ramifie à son extrémité pour communiquer avec les autres neurones dans des zones appelées les synapses. Lorsque le neurone devient suffisamment excité (lorsque la charge accumulée dépasse un certain seuil), par un processus électrochimique, il engendre un potentiel électrique qui se propage à travers son axone pour éventuellement exciter d'autres neurones.

Il semble que c'est l'arrangement spatial des neurones et de leur axone ainsi que la qualité des connexions synaptiques individuelles qui détermine la fonction précise d'un réseau de neurones biologiques. C'est en se basant sur ces connaissances que le modèle mathématique décrit ci-dessus a été défini.

III.2.2. Les réseaux de neurones formels

Par analogie et d'une manière simplifiée, le neurone formel réalise trois (03) opérations sur ses entrées :

- Pondération : multiplication de chaque entrée « P_i » par un paramètre appelé poids de connexion W (efficacité du synapse biologique : si le poids est élevé alors le synapse transfert le signal vers un autre neurone sinon non),
- Sommation : une sommation des entrées pondérées est effectuée,
- Activation : passage de cette somme par une fonction, appelée fonction d'activation ou de transfert « f ». Elle calcule la valeur produite par le neurone appelée sortie du neurone « a » qui est transmise aux neurones suivants.

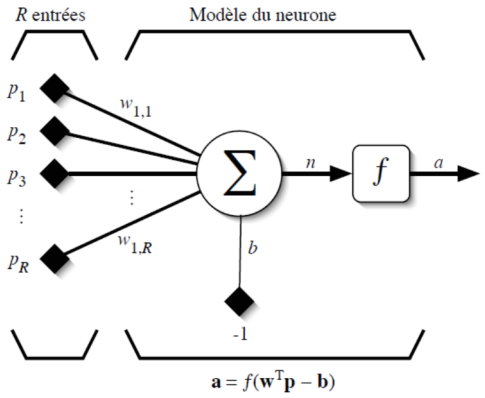


Figure 3.2. Modèle mathématique d'un neurone formel

La sortie de l'intégrateur est le net input n appelé aussi le niveau d'activation du neurone, qui est égal à la différence entre la somme des entrées pondérées W_iP_i et le biais b. le net input n peut s'écrire ainsi :

$n=\sum W_iP_i - b$, i=1,R.

Il existe plusieurs fonctions d'activation, elles sont présentées dans le tableau suivant [K. VIRIYAMETANONT] :

Tableau 3.1. Exemples de fonctions de transfert a=f(n) et nomenclature Matlab

Nom de la fonction	Relation d'entrée/sortie	Icône	Nom Matlab
seuil	$a = 0$ si $n < 0$ $a = 1$ si $n \geq 0$		hardlim
seuil symétrique	$a = -1$ si $n < 0$ $a = 1$ si $n \geq 0$		hardlims
linéaire	$a = n$		purelin
linéaire saturée	$a = 0$ si $n < 0$ $a = n$ si $0 \leq n \leq 1$ $a = 1$ si $n > 1$		satlin
linéaire saturée symétrique	$a = -1$ si $n < -1$ $a = n$ si $-1 \leq n \leq 1$ $a = 1$ si $n > 1$		satlins
linéaire positive	$a = 0$ si $n < 0$ $a = n$ si $n \geq 0$		poslin
sigmoïde	$a = \frac{1}{1+\exp^{-n}}$		logsig
tangente hyperbolique	$a = \frac{e^n - e^{-n}}{e^n + e^{-n}}$		tansig
compétitive	$a = 1$ si n maximum $a = 0$ autrement		compet

III.3. Types de réseaux

Les premiers modèles de RNA étaient les feedforward networks. Ils sont maintenant reconnus comme des outils très puissants pour approximer les fonctions, pour exécuter des tâches de classification ou pour agir comme une mémoire associative (associative memory). Leur apprentissage, basé sur l'algorithme de rétropropagation d'erreur, a été proposé par nombreux chercheurs. Plus encore, il a été prouvé que n'importe quelle fonction continue peut être approximée selon un certain degré de précision par un réseau feedforward s'il y a suffisamment de neurones cachés [36].

Parmi les feedforward networks :

a- Perceptron :
Le plus utilisé est le perceptron multicouche (PML ou MLP). C'est un classificateur linéaire inventé en 1957 par F. Rosenblatt. Il s'agit d'un réseau très simple, où les neurones sont disposés en trois (03) couches de neurones (une couche d'entrée, une couche de sortie et entre les deux un couche cachée), les connexions se font uniquement vers l'avant et dont la topographie ne peut pas évoluer. En revanche, les poids synaptiques peuvent être modifiés afin de parvenir au résultat souhaité.
Ce réseau est capable de réaliser les mêmes opérations qu'un classificateur linéaire. Les réseaux perceptron doivent subir un apprentissage supervisé basé sur l'algorithme de rétropropagation de l'erreur, par la commande « adapt », qui présente les vecteurs d'entrée au réseau un à un et fait les corrections du réseau en se basant sur les résultats de chaque présentation.
Les limitations de ce type de réseau : essentiellement le fait que les sorties ne peuvent avoir que les valeurs 0 ou 1 à cause de la fonction de transfert du type « hardlimit ». Ces réseaux ne peuvent que classer des séries de vecteurs linéairement séparables.

b- Les filtres linéaires :
Similaires au perceptron, sauf que leur fonction de transfert est linéaire et non hardlimited. Et donc, les sorties peuvent prendre n'importe quelle valeur. Mais comme les perceptrons, ils ne peuvent que résoudre des problèmes linéairement séparables.
Limitations des réseaux linéaires : ils ne peuvent apprendre uniquement des relations linéaires entre les vecteurs d'entrée et les vecteurs de sortie.

Ces récents théorèmes d'apprentissage donnent une importante base théorique pour les applications des réseaux feedforward dans nombreux problèmes, cela dit, ces problèmes restent restreints aux tâches statiques : reconnaissance des formes, des caractères, des images, etc. Ce sont des réseaux statiques.

Suivant le développement des réseaux feedforward, quelques modèles ont été introduits présentant une généralisation de l'algorithme basique de rétropropagation de l'erreur traitant des connexions de retour (feedback connections). Plusieurs réseaux dynamiques ayant des connexions récurrentes ont d'importantes capacités jusqu'ici non obtenues dans les réseaux statiques et ont prouvé qu'ils étaient beaucoup plus puissants. Leur particularité est que leur activité neuronale varie au cours du temps : l'information des signaux (phase, fréquence, amplitude) est importante et les différentes parties d'un signal ne sont pas interchangeables.

Parmi les réseaux dynamiques, le focused time-delay (FTDNN). On le créée similairement au réseau feedforward mais en ajoutant comme paramètre d'apprentissage le « tapped delay line vector ». Il existe aussi le distributed time-delay network (DTDNN), le réseau NARX. Les deux réseaux dynamiques les plus utilisés sont les réseaux récurrents : les plus répandus sont le réseau d'Elman, et le réseau de Hopfield.

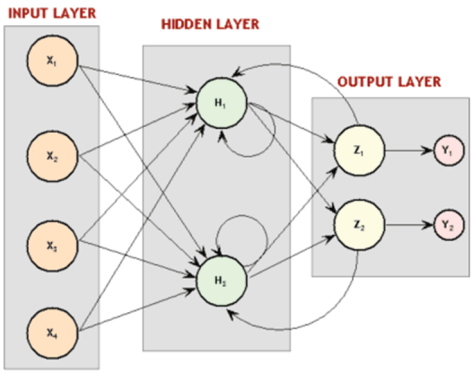


Figure 3.3. Schéma d'un réseau de neurones récurrent.

- Définition d'un réseau récurrent :
  Les réseaux précédents (les feedforward networks par exemple) sont des réseaux où les connections entre les neurones allaient que dans un seul sens, de l'entrée vers la sortie, et n'allaient que d'une couche vers la suivante. Dans les années 80 fut développé un autre type de réseau où existaient des connections entre les neurones dans les deux sens, ce sont les réseaux récurrents.
  Un réseau de neurones est dit récurrent s'il existe au moins un cycle dans la structure des connections entre neurones. Le fonctionnement d'un neurone en lui-même ne diffère pas nécessairement de celui des neurones des modèles antérieurs. L'existence de liens à l'envers permet de créer dans le réseau un comportement dynamique, mais lui rajoute aussi une fonction de mémorisation. En effet, ils sont très utilisés pour leur mémoire associative car ils sont capables de reconstituer une donnée à partir d'informations incomplètes ou partielles et/ou bruitées. Ils peuvent également résoudre des problèmes dont on n connait pas la solution (recherche opérationnelle). De tels réseaux sont donc plus complexes, mais également beaucoup plus puissants.
  Exemples de réseaux récurrents : le réseau d'Hopfield, le réseau d'Elman.

III.4. Architecture du réseau

L'association de plusieurs neurones formels forme un réseau de neurones artificiels. Ils diffèrent selon le type de connections entre les neurones, une cinquantaine de types peut être dénombrée. Par exemple : le perceptron de Rosemblat (classificateur linéaire), les réseaux de Hopfield etc.

La topologie d'un réseau de neurone est définie par son architecture (ou structure) et la nature de ses connexions.

Les réseaux de neurones sont définis par une topologie sous forme de couches. Les neurones de deux couches adjacentes sont interconnectés par des poids. L'information dans le réseau se propage d'une couche à l'autre, on dit qu'ils sont de type «feedforward ». Le réseau le plus connu appartenant à ce type de famille dite « feedforward » est le perceptron multicouche (Multi Layer Perceptron MLP).

Nous distinguons trois types de couches :

- Couche d'entrée : c'est la première couche du réseau. Les neurones de cette couche reçoivent les valeurs d'entrée du réseau (données source utilisées pour l'analyse) et les transmettent aux neurones cachés (les neurones de la seconde couche). Chaque neurone reçoit une valeur, il ne fait pas donc de sommation.
- Couches cachées : chaque neurone de cette couche reçoit l'information de plusieurs couches précédentes, effectue la sommation pondérée par les poids, puis la transforme selon sa fonction d'activation. Par la suite, il envoie cette réponse aux neurones de la couche suivante.
- Couche de sortie : elle joue le même rôle que les couches cachées, la seule différence entre ces deux types de couches est que la sortie des neurones de la couche de sortie n'est liée à aucun autre neurone.

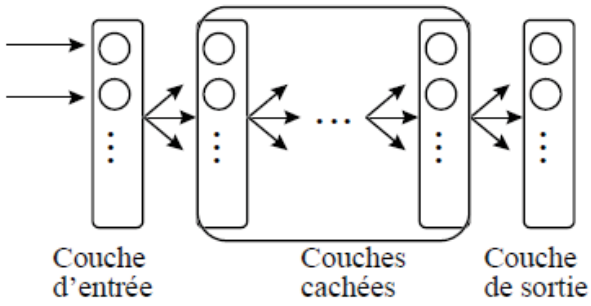


Figure 3.4. Schéma d'un réseau de neurones à couches

Les couches des réseaux de neurones sont généralement ordonnées dans le sens croissant de la couche d'entrée vers la couche de sortie.

Le Modèle de connexion définit la manière dont sont interconnectés les neurones d'un réseau. En se basant sur la structure à couches, on distingue différents types de connexions :

- les connexions intercouches (interconnexion entre neurones de couches voisines),
- les connexions supracouche (lorsque les couches ne sont pas adjacentes),
- les connexions intracouches (connexions entre neurones d'une même couche) et l'autoconnexion (un neurone avec lui-même).

De manière générale, le sens de transfert de l'information dans un réseau est défini par la nature des connexions: directes ou récurrentes. Les connexions directes sont celles qui sont dirigées d'une couche d'indice inférieur vers une couche d'indice supérieur. Les connexions sont dites récurrentes lorsque des sorties de neurones d'une couche sont connectées aux entrées d'une couche d'indice inférieur.



Figure 3.5. Schéma de (a) connexion directe et (b) connexions récurrentes d'un réseau de neurones

III.5. Apprentissage du réseau

L'une des propriétés les plus avantageuses d'un RNA est sa capacité à améliorer sa performance à travers un processus d'apprentissage. C'est un processus dynamique et itératif qui modifie les paramètres du réseau (les poids) en réaction avec les stimuli qu'il reçoit ; ces modifications vont faire que le réseau présente une réponse nouvelle : l'erreur entre les sorties produites par le réseau et les sorties désirées, appelée « fonction coût », diminue et donc le réseau s'améliore.

Le réseau s'adapte, en effet, par comparaison entre le résultat qu'il a calculé en fonction des entrées fournies et la réponse attendue en sortie. Dans ce cas, la correction des poids calculée sur la base de l'erreur prédiction/cible est rétropropagée vers les entrées pour mettre à jour les poids des neurones. L'apprentissage est réalisé selon des algorithmes d'optimisation itératifs plus ou moins classiques. La fonction de performance couramment utilisée est l'erreur quadratique moyenne (EQM) mesurée entre les valeurs prédites et les valeurs visées. L'EQM calculée à l'itération n est notée En.

L'algorithme d'apprentissage le plus utilisé jusqu'à présent est la technique de rétropropagation du gradient de l'erreur. Il s'agit, à l'itération n, de calculer la matrice ∂Wn/∂En définie comme le gradient de l'erreur En par rapport aux poids Wn, de la dernière couche vers la première. On calcule alors la matrice des poids actualisée Wn+1 selon la relation suivante où η constitue le pas de descente :

$$W_{n+1} = W_n - \eta . \partial W_n / \partial E_n$$

Cette procédure est itérée afin de réduire progressivement l'erreur commise par le RNA. Elle est interrompue lorsque le critère d'arrêt de l'algorithme d'apprentissage est atteint. Ce principe général fonde les méthodes de type « descente de gradient ».

Ces algorithmes sont sensibles à l'existence de minima locaux. Ils présentent également l'inconvénient d'être lent, car la convergence nécessite un pas assez faible.
Ces limitations ont conduit au développement d'autres algorithmes plus élaborés : les méthodes de Quasi-Newton et Levenberg-Maquardt. Ces algorithmes font appel aux dérivées secondes de la fonction coût par rapport aux poids du réseau. Ils améliorent nettement la vitesse de convergence par rapport aux algorithmes de rétropropagation d'erreur.

Concernant l'arrêt du processus d'apprentissage, le critère le plus simple consiste à fixer un nombre maximum d'itérations, définissant ainsi une limite supérieure sur la durée de l'apprentissage. Un deuxième critère commun consiste à fixer une borne inférieure sur l'EQM ou encore sur sa racine carrée. Ces deux critères comportent cependant des limitations. En effet, fixer la durée de l'apprentissage n'est aucunement liée à la performance du réseau (l'EQM n'est pas garantie minimale) et fixer une limite inférieure à l'EQM peut créer un autre problème, celui du sur-apprentissage.

Afin d'éviter le sur-apprentissage, on préconise de mettre en œuvre la méthode dite de validation croisée (cross validation). Celle-ci consiste à diviser les données en deux (02) ensembles : un ensemble pour l'apprentissage et un autre pour la validation où l'indice de performance sera calculé). Le seuil de sur-apprentissage est détecté lorsque la performance du réseau sur l'ensemble des données de validation commence à se dégrader, alors que la performance sur la base d'apprentissage continue de s'améliorer. La détection de ce seuil constitue le critère d'apprentissage optimal et les poids retenus à la fin sont ceux calculés lors de l'itération correspondante.

III.6. Types d'apprentissage

Selon la définition donnée pour le processus d'apprentissage d'un réseau de neurones, il est possible de dire que l'apprentissage peut être divisé en deux (02) catégories : apprentissage supervisé ou non supervisé.

- Apprentissage supervisé : est dit lorsque le réseau connaît la réponse désirée qu'il doit produire, et ajuste ses paramètres de telle sorte à ce qu'il produise une réponse aussi proche que possible de celle désirée (la différence entre elles est minimale) ;
- Apprentissage non supervisé (ou auto-organisationnel): comme par exemple le réseau de Kohonen, est dit lorsque la réponse désirée du réseau est inconnue. De ce fait, à partir d'un ensemble d'exemples, le réseau essaie par lui-même de se stabiliser autour d'un état de réponse après quelques itérations. Dans ce cas, l'apprentissage est basé sur des probabilités. Le réseau modifie ses paramètres en fonction des régularités statistiques de l'entrée et établir des catégories, en attribuant et en optimisant une valeur de qualité, aux catégories reconnues.

III.7. Construction d'un réseau de neurones

L'architecture d'un RNA se définit selon les étapes suivantes :

- Choisir le type de réseau ;
- Choisir le nombre de couches cachées et le nombre de neurones dans chaque couche cachée;
- Le nombre d'entrée et le nombre de neurones de la couche de sortie sont fixés par les spécifications du problème ou par tâtonnement lors de problèmes plus complexes ;
- Choisir le type de fonction d'activation de chaque neurone;
- Fixer l'algorithme d'apprentissage.

Trois problèmes doivent être signalés:

- La sensibilité de l'algorithme de rétropropagation (convergence) à l'ordre des présentations de la base d'exemples et au prétraitement effectué;
- L'importance du temps de calcul;
- L'indétermination liée au tirage aléatoire des poids des connexions lors de la création d'un réseau.

Il n'existe pas de procédure rigoureuse permettant de définir l'architecture d'un RNA pour résoudre un problème donné. Plusieurs architectures peuvent mener à des performances comparables pour un même problème. Habituellement, l'approche la plus utilisée est celle dite «test-et-erreur » : créer plusieurs réseaux et choisir le plus performant.

III.8. Applications des RNA dans le Génie Civil

L'application des RNA en génie civil a débuté dans les années 1990. Parmi les premiers articles publiés dans le domaine du GC, celui d'Adeli et Yeh [2] décrivant l'utilisation et l'apprentissage du perceptron dans la conception de l'ingénierie, et depuis plusieurs équipes se sont orientées vers l'application des RNA aux problèmes de mécanique des structures, modélisation des matériaux, contrôle des vibrations, prédiction des fissures, détection des dommages [71, 102, 108].

La technique de modélisation par RNA des comportements des matériaux a été introduite par J.Ghaboussi [31], qui utilise la capacité d'apprentissage des RNA pour simuler le comportement des matériaux. Aussi, la prédiction de la résistance en compression des bétons ordinaires et à haute performance a été élaborée par Yeh [105], utilisant un Perceptron multicouches pour estimer l'évolution des résistances en compression des bétons à haute performance. En 2006, Öztaş et al., [65] utilisent les RNA dans la prédiction de la résistance en compression et l'affaissement des bétons à hautes performances.

Les RNA ont aussi été utilisés pour estimer la capacité portante (moment de flexion, cisaillement, adhérence,…) de structures en béton armé ou précontraint par Mansour et al., [53]. Ils ont utilisé un PMC pour prédire la résistance ultime au cisaillement de poutres en béton armé. Les auteurs ont confirmé la faisabilité de la mise en œuvre des RNA pour résoudre les problèmes de prédiction des charges de ruine. En 2005, Sakla et Ashour [78] ont aussi utilisé un PMC pour estimer l'ancrage des armatures dans le béton. Inel en 2007 [39] a également utilisé un PMC pour estimer la capacité de chargement de poteaux en béton armé.

Dans le domaine du contrôle non destructif (CND), les RNA sont de plus en plus utilisés du fait de leur capacité à résoudre des problèmes d'optimisation, de modélisation, d'aide à la décision, de prédiction et de détection de défauts. Dans ce contexte, plusieurs chercheurs mettent en œuvre les RNA notamment pour le traitement de données de CND. Par exemple, l'utilisation des réseaux de neurones a permis à Liu et al [50] d'identifier et de classifier de la fissuration par une technique ultrasonore. En 2004, Tsuchimoto et al., [94] ont développé un système de diagnostic des dommages structurels et de le dégradation en utilisant les RNA comme système d'identification des dommages. Dans la même année, Zaid et al., [107] quant à eux, ont prouvé la performance des RNA une fois couplés aux techniques de CND pour la caractérisation géométrique des structures en béton armé (profondeur et diamètres des armatures).

En 2005, Shaw et al., [80] ont utilisé les RNA dans l'interprétation des données RADAR et l'extraction d'informations sur la position des armatures, se basant sur une banque de données expérimentale réalisée sur un matériau simulé.

L'apprentissage de leur PMC est réalisé sur la base des résultats de simulations numériques (éléments finis) de propagation d'ultrasons en milieux fissurés. Par ailleurs, Hola et Schabowicz [34] fusionnent trois méthodes non destructives (ultrasons, marteau de Schmidt, impact echo) pour la prédiction de la résistance en compression du béton en laboratoire et in situ. Dans une autre étude [40], la vitesse d'impulsions d'ultrasonores est utilisée en entrée d'un PMC pour prédire la résistance en compression de bétons.

En hydraulique également, l'efficacité des RNA fait ses preuves. En 1999, Vazquez [95] les utilise pour appliquer le modèle de Muskingum aux réseaux d'assainissement. La performance des RNA face aux problèmes de prédiction fait que Kouki [42] utilise l'apprentissage multi-objectif de ces derniers pour la prévision des crues. En 2011, Line Kong [49] utilise les RNA pour la modélisation des crues des bassins karstiques. Quand en 2013, Kouassi et al., [41] appliquent et comparent deux modèles : un modèle conceptuel et un modèle de RNA à la simulation des débits annuels dans le bassin versant du N'zi-Bandama en côte d'Ivoire.

Les larges applications des RNA dans le GC ne font pas abstraction au génie-parasismique : la génération des accélérogrammes compatibles à des spectres de réponses [30], la détection des périodes propres des bâtiments préfabriqués à moyenne et grande hauteur ainsi que la simulation de leurs réponses dynamiques [46], l'estimation du risque lié à l'effet de site1-D [9] ou 2-D [67], l'estimation d'un séisme artificiel et la réponse spectrale [79], évaluation des dommages [108], la détermination de l'accélération maximale sismique du sol [91], l'évaluation de la liquéfaction [84], la prédiction de la période dynamique et du facteur d'amplification d'un sol en zone urbaine [76], l'estimation de la vitesse maximale du sol [24], la prédiction de la vulnérabilité des constructions [58], l'évaluation de l'excentricité accidentelle à partir d'une réponse dynamique [23].

III.9. Conclusion

Les réseaux de neurones artificiels se répandent de plus en plus et leurs applications s'étendent sur divers domaines notamment le génie civil mais aussi le contrôle. Etant donné leur puissante capacité de généralisation à partir de quelques exemples d'apprentissage, leur utilisation dans le cas de cette étude s'avère appropriée et leur performance sera testée dans ce qui suit.

Chapitre IV : Contrôle de la table vibrante par la méthode neuronale

IV.1. Introduction

La table vibrante consiste généralement en une plateforme qui peut se déplacer suivant un ou plusieurs DDL actionnée par des vérins hydrauliques. Sa fonction primaire étant de reproduire au niveau du laboratoire la nature réelle de l'accélérogramme qui testera le spécimen monté sur la plateforme. Ceci se réalise à travers un système de contrôle qui est responsable de l'envoi des signaux de commande aux servovalves des vérins dans lesquels la pression de l'huile causera le déplacement de leurs bras, et donc le déplacement de la plateforme.

Le signal envoyé est un accélérogramme qui, dans son parcours de la source vers la base du spécimen monté sur la table, sera modifié par nombreux effets du système entier. Il se produit alors une erreur entre le signal mesuré au niveau de la plateforme et le signal envoyé. Elle peut être mesurée et une correction au signal envoyé sera appliquée de manière itérative jusqu'à ce que le signal désiré soit produit par le système. C'est la stratégie conventionnelle du contrôle de la table vibrante.

Des travaux antérieurs sur les performances des tables vibrantes ont révélé leurs fortes non linéarités (servovalves, forces dissipatives, variation de pression et de volume dans les vérins, etc.). Mais aussi, le spécimen peut avoir, lors de la plupart des tests faits, un comportement non linéaire, et de ce fait ses caractéristiques peuvent brusquement changer au cours du test et surtout si une rupture se produit. De ce fait, les caractéristiques dynamiques du système entier que forment la table et ses équipements changent et les paramètres du tuning ne sont alors plus adéquats. La convergence du contrôleur, c'est-à-dire le signal contrôlé converge vers le signal désiré nécessite que les propriétés du spécimen test soient inchangés donc, tout test durant lequel le comportement non linéaire du spécimen est atteint, est clairement empêché. Ceci représente un certain paradoxe.

Les procédures itératives online, comme décrites précédemment, tentent de remédier à ces insuffisances mais ont toutes les deux l'inconvénient majeur qu'est le dommage prématuré du spécimen.

Durant des années, plusieurs insuffisances dans le contrôle des tables vibrantes ont montré que celles-ci ne peuvent suivre le mouvement désiré pendant un comportement non linéaire des spécimens testés, nécessitant ainsi de revoir les algorithmes et méthodes de contrôle utilisés.

Cette présente étude propose un contrôle offline basé sur la méthode neuronale où un RNA sera construit sur Matlab. Son apprentissage basé sur des ensembles input/output pourra fournir au réseau une capacité à généraliser et donc à prédire le signal qui doit être émis par la source afin de recueillir au niveau de la plateforme le signal désiré.

IV.2. Principe et méthodologie

IV.2.1. Principe

La distorsion du signal peut être mise en évidence en comparant le signal envoyé par la source et le signal recueilli au niveau du CDG de la plateforme qui excitera le spécimen. Idéalement, ces signaux sont identiques, mais ce n'est pas le cas. Conte et al. (2010) montrent la différence entre le signal de commande (désiré) et le signal de retour mesuré.

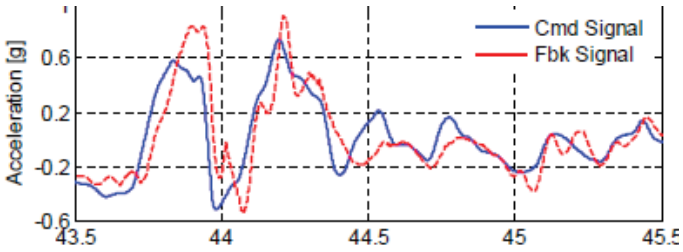
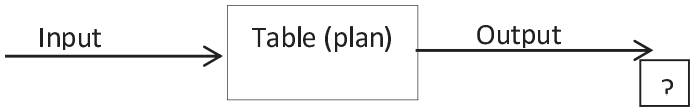


Figure 4.1. Mise en évidence d'une distorsion de signal [20].



Dans cette présente étude, un réseau de neurones artificiels sera construit dans le but de déterminer un signal appliqué au système de commande qui, une fois envoyé, sera modifié et donnera le signal de réponse désiré au niveau de la plateforme.

L'apprentissage du réseau se fera sur une base de données construite par des ensembles input/output recueillis d'un modèle de la table et de ses équipements. L'input représente le signal au niveau du CDG de la plateforme ; l'output représente le signal appliqué au système de commande de la table. En l'absence d'un système de contrôle online, ce dernier est le signal recueilli dans les vérins.

Une fois l'apprentissage validé, le réseau sera capable de généraliser et donc, pour tout input, il produira l'output (signal traité ou modifié) qui donnera le signal le plus proche possible du signal désiré au niveau de la plateforme.

Ceci constitue un contrôleur robuste de la table s'opérant offline : il permet ainsi de reproduire plus fidèlement le signal désiré.

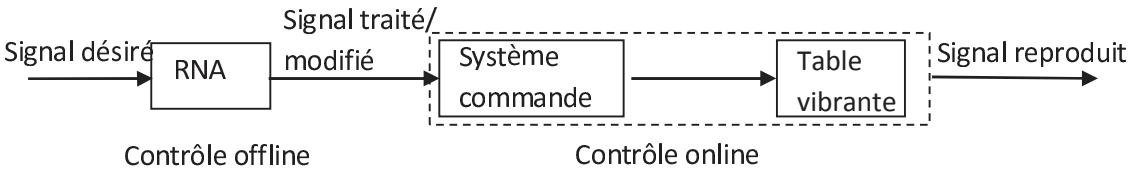


Figure 4.2. Schéma de la procédure d'ajustement du signal par RNA

IV.2.2. Méthodologie

Le réseau de neurones est construit selon les étapes citées dans le chapitre précédent et constitue le cœur du système de contrôle de la table vibrante proposé dans cette étude. Les données qui serviront à l'apprentissage puis au test et à la validation du réseau ont été tirées de deux modèles du système de la table et de ses équipements. Le premier modèle de la table utilisé est un modèle en éléments finis représentant les composants mécaniques de la table, c'est-à-dire le spécimen, la plateforme et les vérins hydrauliques. Le second modèle de la table est sous forme de fonctions de transfert implémentées dans Matlab dans lequel les composants mécaniques, hydrauliques et électroniques sont modélisés par leurs fonctions de transferts, c'est-à-dire le spécimen, la plateforme, le bloc de réaction, les vérins hydrauliques, les servovalves et le contrôleur online de la table.

Ces données d'apprentissage sont des couples input-output (appelés exemples) où l'input est le signal de réponse de la plateforme, un accélérogramme recueilli au CDG de la plateforme de la table (modèle EF), ou l'accélérogramme de réponse de la fonction de transfert (modèle Matlab), et l'output du réseau est le signal de commande, signal recueilli au niveau de l'extrémité fixe des vérins (modèle EF) ou le signal de commande envoyé (modèle Matlab). Ce signal est alors comparé au signal désiré et il devient possible de calculer le signal d'erreur à minimiser pour corriger le signal de commande.

Dans ce cas, c'est un des accélérogrammes réels enregistrés lors du séisme de Boumerdes qui servira à l'apprentissage du réseau.

Les signaux qui serviront au test et à la validation du réseau sont des accélérogrammes (de Boumerdes) enregistrés à des stations différentes, mais aussi l'accélérogramme d'El Centro, de Kobe et un signal sinusoïdal.

Chaque signal contient plus de 4000 points, ce qui est suffisant pour un bon apprentissage.

Une fois l'apprentissage fini, le réseau sera alors capable de généraliser.

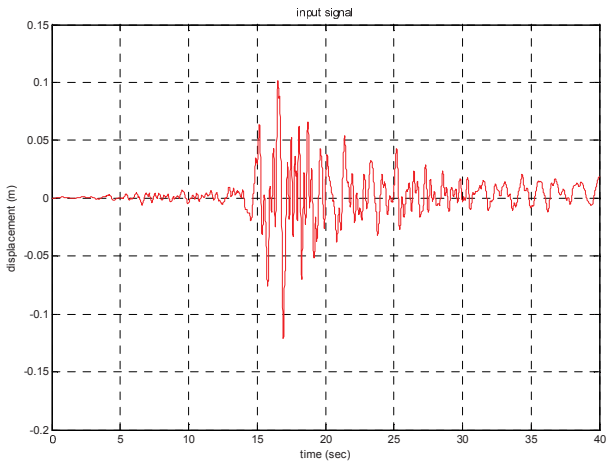


Figure 4.3. Signal d'entrée du réseau.

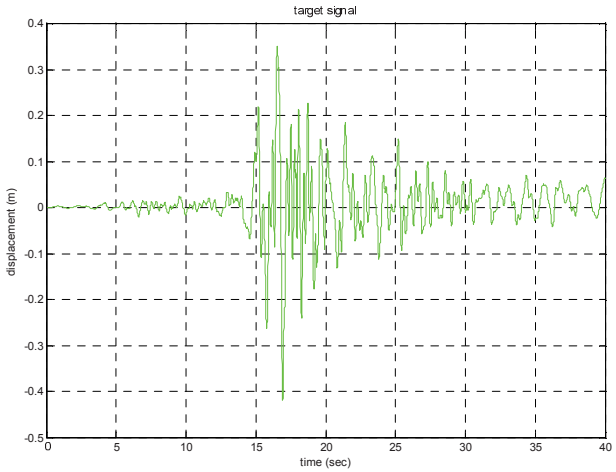


Figure 4.4 : signal désiré du réseau.

IV.3. Modélisation de la table vibrante

La modélisation de la table constitue une étape cruciale nécessaire pour un bon apprentissage du réseau et donc une bonne généralisation. Les données doivent être représentatives du processus et c'est pour cela qu'une précision dans le modèle est nécessaire.

Compte tenu de la complexité du système tout entier de la table et de ses équipements, mais aussi les fortes non linéarités internes de la table, les interactions dynamiques entre ses différents sous-systèmes, rendent le modèle mathématique beaucoup plus complexe et remettent en question sa précision.

Plusieurs travaux se sont focalisés sur la modélisation de la table vibrante et de ses équipements afin d'améliorer sa performance. D'après ces études, plus le modèle est réaliste et plus la distorsion de signal pourrait être minimisée. Parmi ces travaux :

En 1995, Dyke et al., [25] développent un modèle linéaire du système servovalve/vérin dans lequel ils utilisent les feedback en accélération comme stratégie de contrôle.

En 2000, Conte et Trombetti [19] développent un modèle analytique linéaire d'une table uni-axiale de taille petite à moyenne, utilisant un contrôleur linéaire. Leur modèle permet de représenter mathématiquement la relation input-output entre le mouvement désiré et reproduit sur la table. Chaque composant de la table est alors représenté par sa fonction de transfert (servovalve, servovalve/vérin, le système servo-hydraulique et le système de la table entier c'est-à-dire la fonction de transfert totale).

En 2001, Williams et al., [101] développent un modèle numérique réaliste du système servovalve/vérin avec un spécimen linéaire élastique.

En 2004, Thoen et Laplace [90] présentent un modèle numérique d'une table vibrante de taille moyenne incluant uniquement une modélisation des composants mécanique et hydraulique du

système : un modèle non linéaire du système servovalve/vérin avec un spécimen linéaire est développé.

Ozcelik et al. [63] développent un modèle virtuel de la grande table NEES-UCSD, incluant les modèles des servovalves, des vérins, des accumulateurs, un modèle en 2D du système mécanique de la table et un modèle linéaire/non linéaire du spécimen.

En 2008, Airouche et al. [4] développent 2 modèles d'une table uni axiale (EUCENTRE TREESLab high performance uni axial shaking table), l'un représente une linéarisation du système servovalve/vérin (modèle simplifié) et l'autre un système réaliste non linéaire plus représentatif. En 2013 [3], ils réalisent une identification paramétrique de la table vibrante du CGS.

En 2013, Pagano et al. [66] développent un modèle mathématique non linéaire en identifiant ses paramètres par une méthode basée sur la minimisation de l'erreur quadratique entre les data expérimentales et simulées.

IV.3.1. Modèle mathématique de la table vibrante

Un modèle mathématique simple représentant le système mécanique de la table est donné comme suit :

$$M_e \ddot{u}_x(t) + K_e\, u_x(t) + (C_e\, |\dot{u}x(t)^{\alpha}| + \mu_e\, F_v)\, \mathrm{sign}(\dot{u}x(t)) = F_{act}(t) \qquad (4.1)$$

Où :

- M_e : est la masse effective ;
- K_e : est la rigidité effective ;
- C_e : est l'amortissement visqueux effectif ;
- μ_e : est le coefficient de frottement de Coulomb effectif ;
- F_v : la force verticale agissant sur les appuis verticaux générée par les parties servo-hydrauliques de la table;
- F_{act} : la force du vérin du côté considéré de la plateforme ;

Les composants mécaniques comme les appuis verticaux et les hold-down strusts sont représentés comme des ressorts élastiques et le mécanisme de dissipation (c'est-à-dire le terme (Ce $|\dot{u}x(t)^{\alpha}|$+μe Fv) sign($\dot{u}x(t)$)) est représenté par des amortisseurs visqueux non linéaires et des éléments de frottement de type Coulomb. La constante des ressorts pour les appuis verticaux sont déterminés en considérant les caractéristiques physiques des appuis verticaux c'est-à-dire l'aire effective de l'appui, la largeur de la colonne d'huile (oil column length) et le module de compressibilité de l'huile (bulk modulus of the oil).

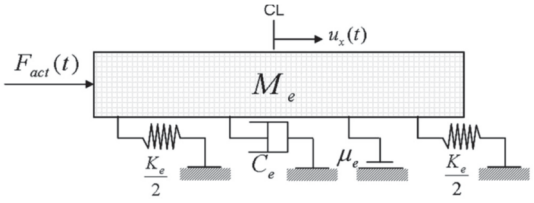


Figure 4.5. Schéma mathématique de la table vibrante [64].

IV.3.2. Modèle par Eléments Finis de la table vibrante

Une première modélisation de la table concernant uniquement l'aspect physique de celle-ci a été réalisé : c'est une modélisation par éléments finis. Le logiciel utilisé est le logiciel de modélisation par EF qu'est SAP2000.

Les éléments à modéliser sont :

- Le spécimen ;
- la plateforme ;
- Les vérins hydrauliques.

La masse de réaction et le sol entourant la table sont supposés être rigide et les observations expérimentales ont prouvé que ceci était suffisant [64].

Deux types de spécimens seront utilisés, un spécimen rigide et un spécimen flexible d'une fréquence fondamentale de 13Hz et 5Hz respectivement.

L'assemblement de la plateforme rigide, le spécimen et le bloc de réaction est schématisé d'après Ozcelik (2008) comme suit :

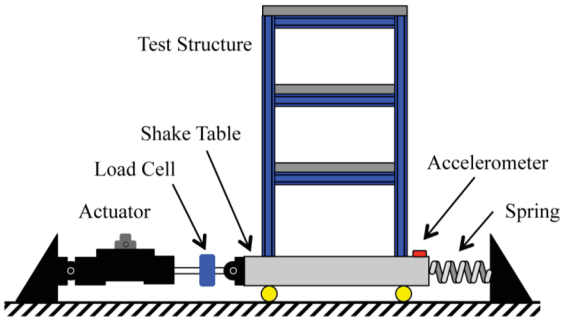


Figure 4.6. Schéma représentatif de la table vibrante et de ses équipements.

➢ Spécifications de la table vibrante du CGS :

La table vibrante du CGS, équipée par MTS System Corporation consiste en une plateforme en acier de 6.1mx6.1m actionnée par douze (12) vérins servo-hydrauliques : quatre (04) longitudinaux, quatre (04) verticaux et quatre (04) verticaux. La table a six (06) degrés de liberté : les déplacements longitudinaux, latéraux, verticaux, les rotations autour de l'axe longitudinal x (roll), autour de l'axe latéral y (pitch) et autour de l'axe vertical z (yaw) [3].

La table peut simuler des séismes et autres mouvements vibratoires avec un déplacement maximal de ±250mm et ±150mm dans les directions horizontales. Le déplacement dans les directions verticales est de ±100mm. La vitesse maximale de 1.1m/s pour chacune des deux directions horizontales et de 0.8m/s dans la direction verticale peuvent être atteintes. Les

accélérations maximales sont de ±1.0g et de 0.8g dans les directions horizontales et verticales, respectivement, atteintes pour un spécimen de masse maximale de 60t. La bande de fréquences opérationnelles est de 0-50Hz [3].

La table vibrante est supportée par une masse de réaction conçue pour transmettre les vibrations induites par le test dans le sol. Cette masse est construite en béton armé et a une totale masse de 6400t.

La pression hydraulique fournie aux servovalves est alimentée par cinq (05) pompes à haute pression qui peuvent délivrer un débit total de 3600litres/min à une pression de 21 MPa.

La figure suivante représente un modèle en éléments finis de table chargée d'un spécimen rigide :

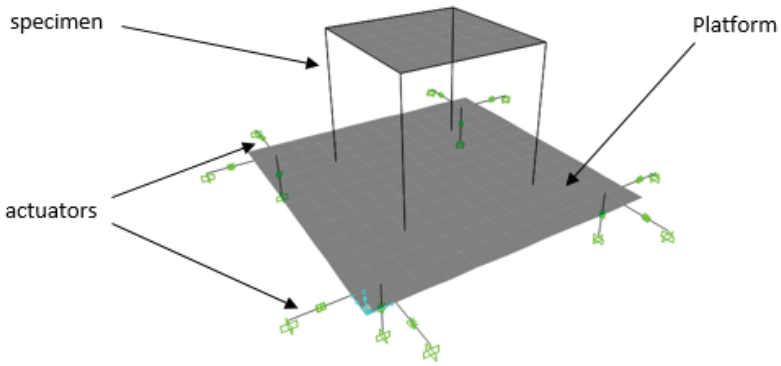


Figure 4.7. Modèle par Eléments Finis de la table avec le spécimen.

IV.3.3. Modèle sur Matlab de la table et de ses équipements

Afin de modéliser le comportement global de la table vibrante plus fidèlement incluant le contrôleur online, un modèle plus complet de celle-ci est nécessaire. Il inclue la modélisation de ses composants mécaniques, hydrauliques et électroniques comme montrés sur le schéma suivant [63] :

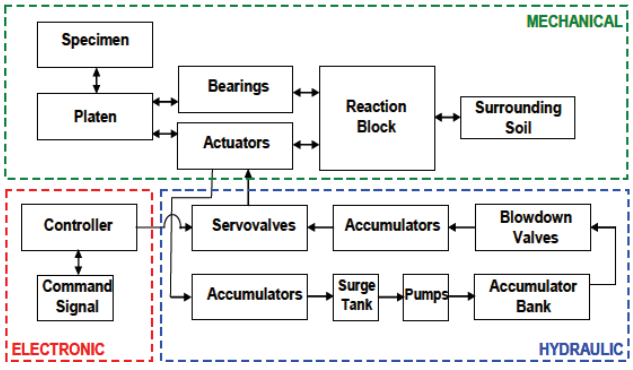


Figure 4.8. Composants et diagramme d'interaction de la table vibrante.

Dans cette présente étude, les composants modélisés sont : le bloc de réaction, la plateforme, le spécimen, les vérins hydrauliques, les servovalves, et le contrôleur. Concernant ce dernier, le TVC a été remplacé par un contrôleur linéaire de type PID uniquement pour des raisons de simplification compte tenu du fait que le modèle de la table ne fournit que les données d'apprentissage, du test et de la validation du RNA.

Donc le contrôleur utilisé est un PID incluant les différents gains du système de contrôle qui sont le gain proportionnel K_{pro}, le gain intégral K_{int}, le gain dérivatif K_{der}, le gain feedforward K_{ff} et le gain de différence de pression K_{dp} [4, 63, 93].

Dans un premier temps, la table sera modélisée sans spécimen puis avec un spécimen à 1DDL.

Le modèle utilisé sera sous forme d'une fonction de transfert totale de la table, c'est-à-dire entre le signal désiré et le signal reproduit, en suivant les étapes suivantes et selon le schéma ci-dessous:

- Modélisation de la servovalve : $H_t(s)$;
- Modélisation du système servovalve-vérin : $S_s(s)$
- Modélisation du système servo-hydraulique incluant le contrôleur, la servovalve et le vérin hydraulique : H(s) ;
- Modélisation de toute la table (sans spécimen) incluant le système servo-hydraulique et la base de la table (le bloc de réaction) : T(s) ;
- Modélisation de toute la table (avec spécimen à 1DDL) incluant le système servo-hydraulique, la base de la table (le bloc de réaction) et le spécimen : $T_{sdof}(s)$

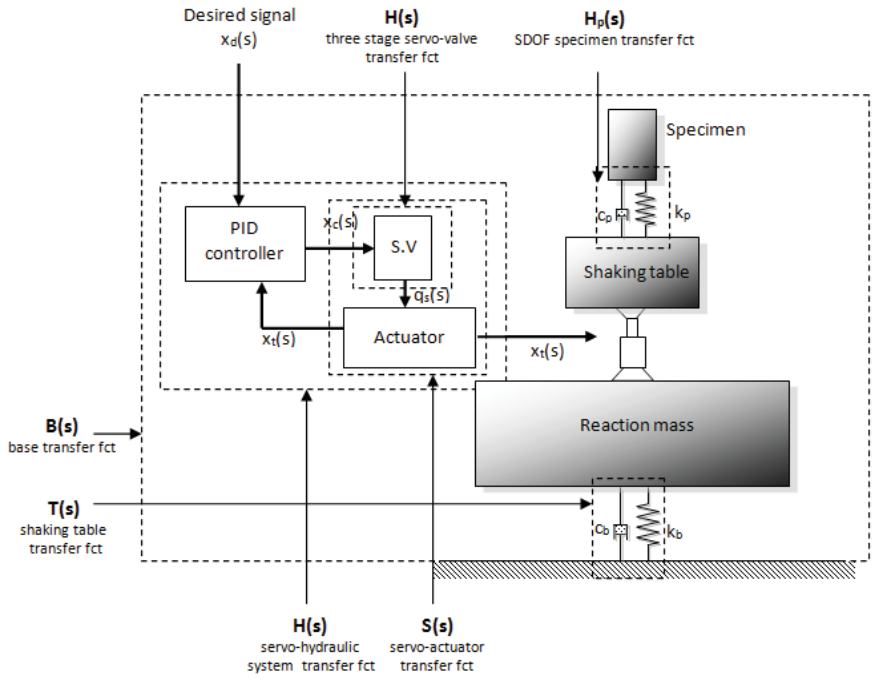


Figure 4.9. Schéma des fonctions de transfert du système [19].

IV.3.3.1. Fonction de transfert de la servovalve

Ce modèle de la table diffère du précédent principalement par l'ajout des servovalves aux vérins hydrauliques. En effet, Les vérins comportent une servovalve électrohydraulique qui est à la fois une interface et un amplificateur de puissance entre la commande électrique et la puissance hydraulique. Son rôle est de fournir le fluide hydraulique aux chambres des vérins à partir du signal électrique qu'elle reçoit du contrôleur.
Elle constitue donc un composant clé car ses caractéristiques affectent largement la performance globale de la table.
La servovalve électrohydraulique est un composant complexe à modéliser à cause des nombreuses caractéristiques non-linéaires qui affectent son fonctionnement. Par exemple, les non-linéarités du flux. Il en existe deux (02) non-linéarités principales : la non-linéarité entre la chute de pression et le débit à travers les orifices de la servovalve, et la non-linéarité du gain de flux due à la non-linéarité du coefficient de déchargement mais due aussi aux variations dans la surpression.

La modélisation de la servovalve se fera sous forme d'une fonction de transfert définie comme le ratio entre le taux de flux d'huile $q_s(s)$ fourni par le 3ème étage de la servovalve à la chambre de pression du vérin et le signal de commande électrique de la servovalve $x_c(s)$ [19]:

$$H_t(s)=\frac{qs(s)}{xc(s)} \quad (4.2)$$

$$H_t(s)=k_t\, e^{-\tau s} \quad (4.3)$$

Où : K_t représente le coefficient gain de la table [19].
La non-linéarité interne de la servovalve est obtenue grâce au temps de retard τ dans la réponse de la servovalve. Physiquement, ce temps de retard peut être interprété comme le temps nécessaire pour passer de la dynamique due à la mécanique à celle due à l'hydraulique dans la servovalve. C'est le temps écoulé entre l'instant où le signal électrique atteint le 1er étage de la servovalve (l'étage électrique) et l'instant où le 3ème étage (l'étage de puissance) répond en conséquence.
Ceci est dû aux effets dynamiques (effets d'inertie) observé expérimentalement par plusieurs auteurs [19, 3, 4].

IV.3.3.2. Fonction de transfert du système servovalve + vérin

La fonction de transfert du système servovalve + vérin entre le déplacement du bras du vérin $x_t(s)$ et le signal de commande de la servovalve $x_c(s)$ est donnée sous la forme suivante :

$$S(s)=\frac{xt(s)}{xc(s)} \quad (4.4)$$

$$S(s)=\frac{\mathrm{Ht(s)}}{\mathrm{sA+kle}\left(\frac{Fa(s)}{xt(s)}\right)+s\left(\frac{V}{4\beta A}\right)\left(\frac{Fa(s)}{xt(s)}\right)} \quad (4.5)$$

L'expression de la fonction de transfert S(s) est en fonction de $F_a(s)$ qui représente la force du vérin et où :

A : représente la section du piston ;

K_{le} : représente le coefficient de perte de pression dans la valve ;

V : représente le volume de la chambre du vérin ;

β: représente le module de compressibilité du fluide.

IV.3.3.3. Fonction de transfert du système servo-hydraulique

La fonction de transfert du système servo-hydraulique entre le déplacement du bras du vérin $x_t(s)$ et le déplacement désiré de la table $x_d(s)$ est donnée sous la forme suivante :

$$H(s)=\frac{\mathrm{xt(s)}}{\mathrm{xd(s)}} \quad (4.6)$$

$$H(s)=\frac{\mathrm{S(s)}[\frac{Kint}{s}+\mathrm{Kpro+s(Kff+Kder)}]}{1+\mathrm{S(s)}[\frac{Kint}{s}+\mathrm{Kpro+s\ Kder}-\left(\frac{kdp}{A}\right)\left(\frac{Fa(s)}{xt(s)}\right)]} \quad (4.7)$$

L'expression de la force du vérin Fa(s) sera développée ci-dessous.

IV.3.3.4. Fonction de transfert totale de la table non chargée (sans spécimen)

En réalité, afin de représenter fidèlement la table, il n'est possible de négliger les effets de flexibilité du bloc de réaction. Car même si le bloc de réaction est considéré comme infiniment rigide et fixe par rapport à un système de référence, il présente néanmoins quelques degrés de flexibilité représentés par un déplacement $x_b(s)$ du bloc de réaction par rapport au système de référence.

La fonction de transfert du bloc de réaction B(s) est obtenue entre le déplacement du bras du vérin $x_t(s)$ et le déplacement du bloc de réaction par rapport au système de référence $x_b(s)$:

$$B(s)=\frac{\mathrm{xb(s)}}{\mathrm{xt(s)}} \quad (4.8)$$

$$B(s)=-\frac{\mathrm{mt}}{\mathrm{mT}}\frac{s^2}{s^2+s.2\xi b\ \omega b+\omega b^2} \quad (4.9)$$

Où :

m_t : représente la masse de la table ;

m_T : représente la masse de la table + la masse du bloc de réaction ;

ξ_b et ω_b : représentent respectivement l'amortissement et la pulsation propre du bloc de réaction.

L'expression de la force du vérin Fa(s) est donnée comme suit :

$$F_a(s)=x_t(s)\ s^2\ m_t[1+B(s)] \quad (4.10)$$

En intégrant ces effets de flexibilité du bloc de réaction dans les expressions des fonctions de transfert du système servo-hydraulique H(s) et du système servovalve+vérin S(s), elles deviennent sous la forme suivante :

$$S'(s)=\frac{Ht(s)}{s^3\left[\frac{V\,mt}{4\beta A}\right][1+B(s)]+s^2 mt\ Kle[1+B(s)]+sA} \quad (4.11)$$

$$H'(s)=\frac{S\prime(s)[\frac{Kint}{s}+\mathrm{Kpro}+s(\mathrm{Kff}+\mathrm{Kder})]}{1+S\prime(s)[\frac{Kint}{s}+\mathrm{Kpro}+s\ \mathrm{Kder}-s^2(Kdp\,\frac{mt}{A})[1+\mathrm{B(s)}]]} \quad (4.12)$$

La fonction de transfert totale de la table non chargée T(s) est définie comme la fonction de transfert entre le déplacement absolu désiré de la table $x_d(s)$ et le déplacement absolu total de la table $x_{ta}(s)$ où $x_{ta}(s)= x_t(s)+ x_b(s)$:

$$T(s)=\frac{\mathrm{xta(s)}}{\mathrm{xd(s)}} \quad (4.13)$$

$$T(s)= H'(s)[1+B(s)] \quad (4.14)$$

L'allure de la fonction de transfert totale de la table non chargée obtenue est comme suit:

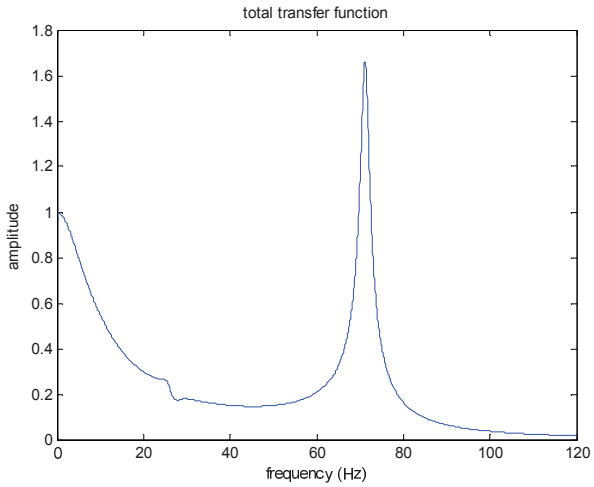


Figure 4.10. Fonction de transfert totale de la table vibrante sans spécimen.

Les paramètres de la table vibrante du CGS nécessaires dans l'élaboration de la fonction de transfert totale de la table ne sont pas tous disponibles, et pour cela, les données utilisées sont ceux de la table de Rice University (Texas, USA) [19].

IV.3.3.5. Fonction de transfert du spécimen testé (spécimen à 1DDl)

La fonction de transfert du spécimen entre le déplacement absolu à la base du spécimen $x_{pa}(s)$ et le déplacement absolu total de la table $x_{ta}(s)$:

$$H_p(s)=\frac{\mathrm{xpa(s)}}{\mathrm{xt(s)}} \quad (4.15)$$

$$H_p(s)=\frac{s\ 2\xi p\ \omega p+\ \omega p^2}{s^2+s.2\xi p\ \omega p+\ \omega p^2} \quad (4.16)$$

Où :

ξ_p et ω_p : représentent respectivement l'amortissement et la pulsation propre du spécimen. Dans notre cas les valeurs choisies sont arbitraires, respectant la masse maximale du spécimen que la table peut supporter ainsi que la plage de fréquences opérationnelles.

IV.3.3.6. Fonction de transfert totale de la table chargée (avec spécimen à 1DDL)

De même que pour la table non chargée, la prise en compte des effets de flexibilité du bloc de réaction est nécessaire.

La fonction de transfert du bloc de réaction B(s) est obtenue entre le déplacement du bras du vérin $x_t(s)$ et le déplacement du bloc de réaction par rapport au système de référence $x_b(s)$:

$$B_{SDOF}(s)= \frac{xb(s)}{xt(s)} \tag{4.17}$$

$$B_{SDOF}(s)=- \frac{s^2\left(\frac{mt}{mT}\right)[1+\left(\frac{mp}{mT}\right)Hp(s)]}{s^2[1+\left(\frac{mp}{mT}\right)Hp(s)]+s.2\xi b\ \omega b+ \omega b^2} \tag{4.18}$$

En intégrant ces effets de flexibilité du bloc de réaction dans les expressions des fonctions de transfert du système servo-hydraulique H(s) et du système servovalve+vérin S(s), elles deviennent sous la forme suivante :

$$S'_{SDOF}(s)=\frac{Ht(s)}{s^3\left[\frac{V\ mt}{4\beta A}\right][1+Bsdof(s)][1+\frac{mp}{mt}Hp(s)]+s^2 mt\ Kle[1+B(s)][1+\frac{mp}{mt}Hp(s)]+sA} \tag{4.19}$$

$$H'_{SDOF}(s)= \frac{S\prime(s)[\frac{Kint}{s}+Kpro+s(Kff+Kder)]}{1+S'SDOF(s)[\frac{Kint}{s}+Kpro+s\ Kder-s^2(Kdp\ \frac{mt}{A})[1+B(s)][1+\frac{mp}{mt}Hp(s)]]} \tag{4.20}$$

La fonction de transfert totale de la table chargée $T_{SDOF}(s)$ est définie comme la fonction de transfert entre le déplacement absolu désiré de la table $x_d(s)$ et le déplacement absolu total de la table $x_{ta}(s)$:

$$T_{SDOF}(s) = \frac{xta(s)}{xd(s)} \tag{4.21}$$

$$T(s)= H'_{SDOF}(s)[1+B_{SDOF}(s)] \tag{4.22}$$

IV.4. Construction du réseau neuronal

Des décennies auparavant, il était clair pour les ingénieurs de contrôle ainsi que pour les informaticiens que faire face à des tâches et des environnements complexes nécessitaient des moyens au-delà de la méthode traditionnelle de la modélisation qui utilisait leurs propriétés physiques. Donc dans la quête de trouver des méthodes et approches de contrôle de processus complexes et de prise de décision dans des environnements fortement non linéaires, les RNA ont reçu un intérêt croissant autant comme identificateur que contrôleur de processus complexes. Il y a essentiellement deux raisons à cela. En premier, le fait que les RNA soient des approximateurs universels. En second, l'origine d'où ils sont inspirés, c'est-à-dire les fascinantes (et très complexes) cellules nerveuses de l'homme :

continuellement, il y avait un espoir pour les chercheurs que cette direction pourrait un jour mener à un système qui se comporte, du moins partiellement, de manière similaire aux systèmes nerveux d l'homme.

Comme mentionné dans le chapitre précédent, la construction d'un réseau de neurones artificiels et le choix de son architecture se réalisent par étapes. Dans cette partie du chapitre, ces étapes seront plus détaillées.

IV.4.1. Constitution des bases de données du réseau

Cette étape se fait en dehors du cadre des RNA, elle a pour objectif de recueillir une série de données à partir du processus à modéliser, à la fois pour développer le réseau de neurones et pour le tester. Dans le cas d'applications sur des données réelles, l'objectif est de rassembler un nombre de données suffisant pour constituer une base représentative des données susceptibles d'intervenir en phase d'utilisation du système neuronal.
La fonction réalisée résultant d'un calcul statistique, le modèle qu'il constitue n'a de validité que dans le domaine où on l'a ajusté. En d'autres termes, la présentation de données très différentes de celles qui ont été utilisées lors de l'apprentissage peut entraîner une sortie totalement imprévisible.

IV.4.1.1. Analyses des données

Il est souvent préférable d'effectuer une analyse des données de manière à présenter au réseau de neurones des caractéristiques représentatives du phénomène. Cette détermination des caractéristiques a des conséquences à la fois sur la taille du réseau (et donc le temps de simulation), sur les performances du système (pouvoir de séparation, taux de détection), et sur le temps de développement (temps d'apprentissage).

IV.4.1.2. Séparation des bases de données

Afin de développer une application à base de réseaux de neurones, il est nécessaire de disposer de deux bases de données : une base pour effectuer l'apprentissage et une autre pour tester le réseau obtenu et déterminer ses performances. Afin de contrôler la phase d'apprentissage, il est souvent préférable de posséder une troisième base de données appelée « base de validation croisée ». Les avantages liés à l'utilisation de cette troisième base de données seront exposés dans les sections suivantes.
Il n'y a pas de règle pour déterminer ce partage de manière quantitatif. Il résulte souvent d'un compromis tenant compte du nombre de données dont on dispose et du temps imparti pour effectuer l'apprentissage. Chaque base doit cependant satisfaire aux contraintes de représentativité de chaque classe de données.

IV.4.1.3. Mise en forme des données pour un réseau de neurones

De manière générale, les bases de données doivent subir un prétraitement appelé normalisation des données dans le but de ramener les données dans un intervalle de valeurs restreintes afin d'améliorer la convergence du réseau. Un prétraitement courant consiste à effectuer une normalisation soit une simple réduction des variables

d'entrées dans un intervalle donné (min-max normalisation) appelé « rescaling », soit une standardisation (lorsque le min et le max sont inconnus) ou alors une normalisation par échelle décimale.

Les données de la présente étude ont été normalisées par la méthode du rescaling de manière à ce que leurs valeurs soient comprisses entre -1 et +1.

Notant que : rescaling : x'=(x-min) / (max-min)

IV.4.2. Choix du type de réseau

Après la collecte des données qui, dans cette étude, sont extraites des modèles de la table vibrante, la première étape est la création du réseau, c'est-à-dire le choix du type de réseau à utiliser.

Le choix d'un réseau peut dépendre :

- De la tâche à effectuer (classification, association, contrôle de processus (c'est notre cas)) ;
- De la nature des données (dans notre cas, des données présentant des variations au cours du temps),
- D'éventuelles contraintes d'utilisation temps-réel (certains types de réseaux de neurones, tels que la 'machine de Boltzmann', nécessitant des tirages aléatoires et un nombre de cycles de calculs indéfini avant stabilisation du résultat en sortie, présentent plus de contraintes que d'autres réseaux pour une utilisation temps-réel),
- Des différents types de réseaux de neurones disponibles dans le logiciel de simulation que l'on compte utiliser (à moins de le programmer).

Il est possible que le choix se fasse par élimination d'autres types étant au préalable inadéquats. Par exemple : Le perceptron multicouche du fait de ses nombreuses limitations. En effet, le PML donne une réponse de 0 ou 1 à cause de la fonction de transfert de type seuil alors que la réponse désiré du réseau de cette étude doit être un signal de déplacement de la plateforme dont les valeurs ne se limitent guerre à 0 ou à 1.

En se basant sur la direction du signal, les RNA peuvent être divisés en deux (02) catégories : les feedforward Neural Network (FFNN) et les Recurrent Neural Network (RNN) comme montré dans la figure suivante :

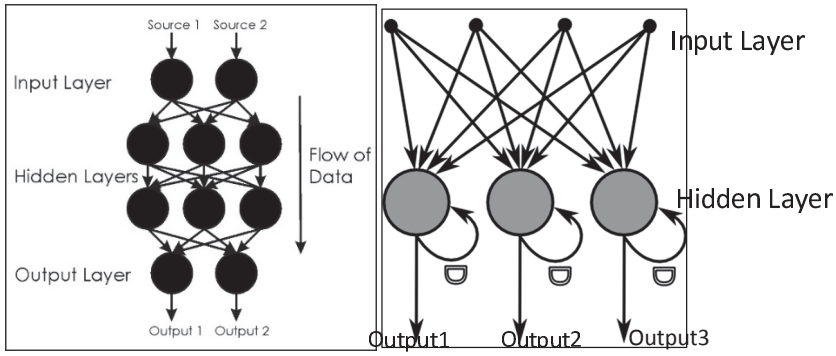


Figure 4.11. Réseau de neurones de type (a) Feed-Forward (b) Recurrent

Les FFNN sont les réseaux les plus courants où la direction des signaux se fait de l'input vers l'output avec aucun retour en arrière (sans feedback), c'est-à-dire chaque sortie de neurone représente l'entrée du neurone de la couche suivante. Introduire des feedback au FFNN fournit un caractère dynamique au réseau et devient alors un RNN dont le comportement est implicitement fonction du temps. En effet, dans les RNN, le neurone peut recevoir des entrées de n'importe quel neurone quel que soit sa position dans le réseau (précédent ou suivant le neurone en question).
Une différence essentielle entre les FFNN et les RNN est que les FFNN sont des réseaux statiques, c'est-à-dire chaque série d'entrée peut produire uniquement une série de sorties. Au contraire, les RNN sont des réseaux dynamiques, c'est-à-dire chaque série d'entrée peut produire différentes séries de sorties.
Il est aussi possible de noter que les FFNN n'ont pas de mémoire contrairement aux RNN qui temporairement mémorisent l'information qu'ils reçoivent. Ils peuvent alors faire face à des systèmes dont les paramètres sont connus partiellement ou varient au cours du temps.
Aussi, les FFNN ont souvent une ou plusieurs couches cachées de neurones sigmoïdes suivis (es) par une couche de sortie linéaire, ce qui est uniquement représentatif de systèmes dont les réponses sont linéaires, alors que dans le cas de présence de non linéarités, les RNN peuvent être utilisés : ils sont considérés comme des systèmes non linéaires [52].

Les réseaux dynamiques sont donc des outils beaucoup plus puissants que les réseaux statiques. Et l'une des principales applications des réseaux dynamiques est les systèmes de contrôle [Matlab].
Donc, les RNN paraissent meilleurs candidats pour cette étude. Le réseau le plus adéquat pour la prédiction de signaux est le réseau d'Elman [7, 23].
Le réseau choisi est le réseau récurrent d'Elman.

- **<u>Réseau récurrent : réseau d'Elman</u>**

  La connectivité des neurones dans les réseaux de neurones récurrents ne se limite pas, comme dans le cas des réseaux à propagation avant (feedforward), à des architectures dans lesquelles l'information se propage de l'entrée vers la sortie couche après couche. Tout type de connexion est admis, c'est à dire d'un neurone à n'importe quel autre, y compris lui-même. Ceci donne lieu à des comportements dynamiques qui peuvent être fort complexes. Parmi les architectures les plus connues, nous pouvons citer le modèle de Hopfield [35] basé sur le concept de mémoire adressée par son contenu (mémoire associative).
  D'autres algorithmes d'apprentissage supervisé ont été proposés pour des types de réseau récurrent à couches. Ce sont des adaptations de l'algorithme de rétro-propagation du gradient des réseaux feedforward [77, 48]. Une de ces adaptations est l'algorithme appelé rétropropagation récurrente [75]. Le principe de cet algorithme est de propager l'erreur de la couche de sortie vers la couche d'entrée, avec une certaine modification des poids et des fonctions d'activations des neurones. Les poids des connexions récurrentes sont ainsi mis à jour avec le même principe que la mise à

jour des poids dans l'algorithme de rétropropagation du gradient. Un autre type d'algorithme d'apprentissage pour les réseaux récurrents est appelé rétropropagation dans le temps. Le but de cet algorithme est d'obtenir une certaine réponse désirée pour certains neurones à certains instants. Williams et Zipser [100] ont proposé un algorithme pour l'apprentissage dans les réseaux entièrement connectés. Une version de cet algorithme appelée RTRL s'effectue en temps réel ce qui revient à réaliser l'apprentissage pendant que les données sont présentées au réseau au lieu d'avoir à attendre que la totalité des données soient produites.

Afin de ne pas trop compliquer le processus d'apprentissage, il existe des réseaux dynamiques partiellement récurrents. Les connexions récurrentes utilisées permettent au réseau de prendre en compte les informations d'un passé récent, et sont généralement fixes (ne font pas l'objet d'un apprentissage). L'architecture proposée par [26] utilise une couche, appelée couche de contexte, qui duplique les états des neurones de la couche cachée à l'instant précédent. Une autre architecture similaire à celle d'Elman est proposée par Jordan en 1986.

Le réseau d'Elman est la forme la plus simple des réseaux de neurones récurrents. Il a été cible d'un très grand intérêt dans la recherche et a été implémenté dans Matlab et appliqué aux systèmes d'identification [68]. Il contient des connections à partir des neurones de la couche cachée à des neurones spéciaux contenus dans une couche appelée couche de contexte. C'est ce qui ajoute au réseau la caractéristique de mémorisation car le réseau apprend non seulement des données actuelles mais aussi des données précédentes du réseau. A chaque instant, l'input est propagé vers l'avant similairement à un mode feed-forward standard puis l'algorithme d'apprentissage est normalement appliqué. Les connections récurrentes, elles, permettent aux neurones de contexte de copier les informations contenues dans les neurones cachés. Le réseau fournit son output en prenant compte ces informations mémorisées.

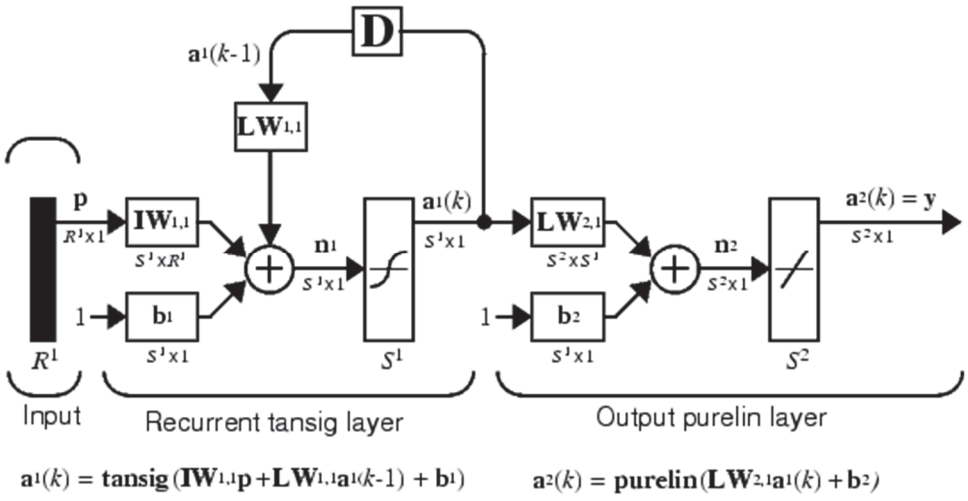


Figure 4.12. Réseau d'Elman.

- **Fonctionnement du réseau d'Elman**

Le réseau d'Elman diffère des réseaux conventionnels à deux couches dans le fait que la première couche a des connexions récurrentes. Le délai dans ces connexions mémorise (emmagasine) les valeurs à partir de l'étape précédente, qui peuvent être utilisées dans l'étape actuelle. Donc, même si deux réseaux avec le mêmes poids reçoivent les mêmes données à un instant donné, leurs sorties peuvent être différentes.

A un instant spécifique k, les activations précédentes des neurones cachés à l'instant k-1 et l'input actuel (à l'instant k) sont utilisés comme inputs au réseau. A ce stage, le réseau fonctionne comme un FFNN et propage ces inputs vers l'avant pour produire un output. La règle d'apprentissage standard de backpropagation peut alors être employée pour l'apprentissage du réseau. Après cette étape d'apprentissage, les activations des neurones cachées à l'instant k sont envoyés en arrière à travers les liaisons récurrentes vers les neurones de contexte et y sont enregistrés pour la prochaine étape d'apprentissage à l'instant k+1.

Si l'input est représenté par u(k) et l'output du réseau par y(k), l'input total entrant au $i^{ème}$ neurone caché est noté $v_i(k)$. L'output sortant du $j^{ème}$ neurone de contexte est noté $x_{jc}(k)$.

$$v_i(k)= \sum w^x{}_{i,j}\,(k-1)\,x_j(k-1)+ w^u{}_i\,(k-1)\,u(k-1),\ j=1\ldots n \qquad (4.23)$$

$$x_i(k)= v_i(k) \qquad (4.24)$$

$$y(k)= \sum w^y{}_i(k-1)\,x_i(k),\ i=1\ldots n \qquad (4.25)$$

où $w_{u\ i}()$,$w_{x\ i}$,j(), et $w_{y\ i}()$, i,j=1…n sont les poids des liaisons respectivement entre la couche d'entrée et la couche cachée, et entre la couche cachée et la couche de sortie. Ces équations décrivent un système dynamique linéaire d'ordre n où n est la dimension du vecteur x(k) qui est le nombre de neurones cachés/de contexte.
Il est aussi possible d'écrire :

$$y(k)= A_1\,y(k-1)+A_2\,y(k-2)+\ldots+A_n\,y(k-n)+B_1\,u(k-1)+B_2\,u(k-2)+\ldots+B_n\,u(k-n) \qquad (4.26)$$

Donc, théoriquement, un réseau d'Elman est capable de modéliser un système dynamique d'ordre n s'il est entrainé pour. Pour modéliser un système représenté par l'équation (4.1) utilisant des couples de données input-output, 2n de neurones d'entrée sont alors nécessaires si un FFNN est utilisé. Pour un réseau d'Elman, un nombre de (n+1) neurones d'entrée est nécessaire en comptant également les neurones de contexte. Ceci montre qu'un réseau d'Elman a une structure nettement plus petite qu'un FFNN, ce qui est très significatif surtout quand n est très grand. [68].

IV.4.3. choix du nombre de couches cachées et le nombre de neurones dans chaque couche du réseau

Il est possible de trouver quelques « règles » disant que le nombre de neurones dans la couche cachée doit être quelque part entre le nombre de neurones de la couche d'entrée et celui de la couche de sortie [12] ou plusieurs autres règles mais celle-ci n'ont pas de réel sens. Il n'y a absolument aucun moyen de construire un bon réseau convergeant uniquement à partir du nombre d'input ou d'output [37, 88]. Mais cela dépend fortement de la représentativité des données, la présence de bruits et la complexité de la fonction que le réseau essaie d'apprendre.

Le nombre de neurones et donc de poids ajustable est un des facteurs fondamentaux de la convergence du réseau : si le réseau possède un trop grand nombre de poids à ajuster, c'est-à-dire si le réseau est « souple », il risque de s'ajuster au bruit présent dans les données d'apprentissage et donc de présenter des oscillations non significatives entre les points d'apprentissage : le réseau aura alors de mauvaises propriétés de généralisation. Si au contraire le nombre de poids à ajuster est trop petit, c'est-à-dire si le réseau est « rigide », il risque de ne pas converger et donc de ne pas reproduire la réponse désirée.

Un choix intelligent du nombre optimal de neurones reste par la méthode dite essai-erreur : construire simplement plusieurs réseaux avec des nombres différents de neurones dans les couches cachées, puis estimer la qualité de chacun des réseaux en leur présentant des exemples qui n'ont pas fait partie des exemples d'apprentissage (estimer leur performance de généralisation par l'erreur) et enfin, choisir le réseau le plus performant (qui généralise le mieux).

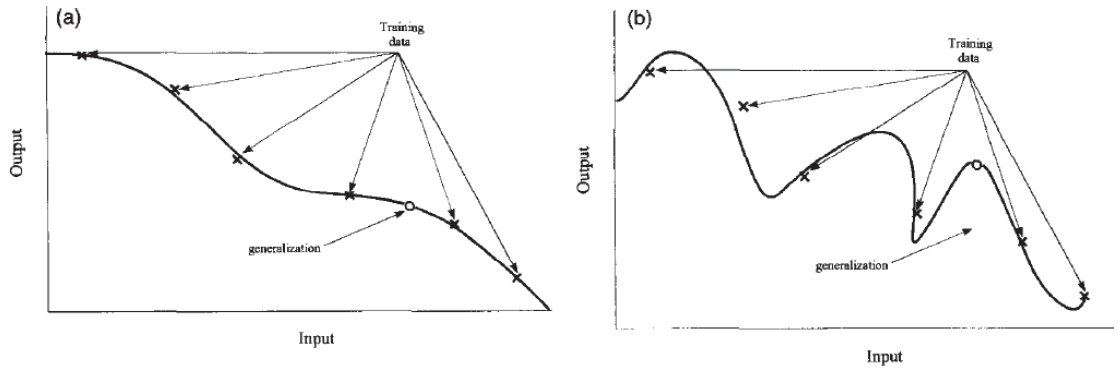


Figure 4.13. (a) bonne généralisation (b) mauvaise généralisation- réseau souple [88]

Concernant le nombre d'entrée et le nombre de neurones de la couche de sortie, ils sont déterminés par les spécifications du problème. Dans cette étude, ce nombre est le nombre de points de l'accélérogramme échantillonné.

IV.4.4. Choix de la fonction de transfert

La fonction considérée est la fonction logistique pour le passage de la couche d'entrée à la couche cachée. Le passage de cette dernière à la couche de sortie sera soit linéaire, soit sigmoïde selon nos types de variables et le résultat recherché.

Généralement, l'obtention des paramètres architecturaux optimaux se fera par comparaison des performances obtenues pour différentes architectures de réseaux de neurones.
Le choix du type de fonction d'activation et de l'algorithme d'apprentissage a été fixé par le choix du réseau utilisé qui est le réseau récurrent d'Elman. De ce fait, le réseau contient des couches cachées ayant une fonction de transfert de type tangeante-sigmoïde et une couche de sortie purement linéaire.

IV.4.5. Choix de l'algorithme d'apprentissage

L'apprentissage est vraisemblablement la propriété la plus intéressante des réseaux neuronaux. L'apprentissage est une phase du développement d'un réseau de neurones durant laquelle le comportement du réseau est modifié jusqu'à l'obtention du comportement désiré. L'apprentissage neuronal fait appel à des exemples de comportement (les stimuli).
La notion d'apprentissage implique mémorisation (capacité d'assimiler des exemples nombreux) et généralisation (capacité de traiter des cas non rencontrés mais similaires).
Plusieurs types d'apprentissages peuvent être adaptés à un même type de réseau de neurones. Les critères de choix sont souvent la rapidité de convergence ou les performances de généralisation.
Le critère d'arrêt de l'apprentissage est souvent calculé à partir d'une fonction de coût, caractérisant l'écart entre les valeurs de sortie obtenues et les valeurs de références (réponses souhaitées pour chaque exemple présenté). Cet écart est généralement décrit par l'erreur quadratique moyenne (MSE). Souvent une erreur de 10% caractérise un réseau incapable de généraliser convenablement.

L'apprentissage se fait à l'aide d'algorithmes de calcul appelés algorithmes d'apprentissage, il a été défini deux (02) grandes catégories selon que l'apprentissage est dit supervisé ou non supervisé. La différence entre ces deux (02) catégories d'apprentissage est la présence ou non d'un « teacher signal » (comportement de référence). Dans un apprentissage supervisé, le « superviseur » connait d'avance, au moins pour une série d'exemples, que doit être le signal de sortie pour un signal d'entrée donné (le comportement désiré en sortie est connu dans l'apprentissage supervisé mais ne l'est pas dans l'apprentissage non supervisé). Le réseau évolue alors en adaptant ses paramètres de telle sorte à approcher le plus le comportement attendu de lui.
Dans cette présente étude, le signal désiré au CDG de la plateforme de la table vibrante que le réseau de neurones est tenu à atteindre est connu, l'apprentissage est alors de type supervisé.

Les algorithmes d'apprentissage se basent généralement sur le calcul du gradient (gradient descent, and gradient descent with momentum) et la rétropropagation du gradient de l'erreur. Ceux-ci sont excessivement simples pour des problèmes plutôt complexes et prennent un temps extrêmement long pour converger (dans le cas où ils convergent). Il existe alors plusieurs algorithmes à plus haute performance qui converge de 10 à 100fois plus rapidement que les anciens.
Il est évident que vu la complexité de notre système, l'algorithme qui sera utilisé sera l'un de ces puissants algorithmes.
Ces algorithmes se divisent en deux (02) catégories :

- la première catégorie utilise des techniques heuristiques qui ont été développées sur l'analyse de la performance des algorithmes du gradient descendant standard (variable learning rate backpropagation algorithm and resilient backpropagation algorithm).
- La seconde catégorie utilise des techniques d'optimisation numérique standard (conjugate gradient algorithm, quasi-Newton algorithm and Levenberg-Marquardt algorithm).

L'algorithme de Levenberg-Marquardt est un standard pour l'optimisation de l'erreur quadratique due à ses propriétés de convergence rapide et de robustesse. Il permet de trouver numériquement une solution au problème de minimalisation de fonction souvent non linéaire et dépendant de plusieurs paramètres. Comme les méthodes quasi-Newtoniennes, l'algorithme de Levenberg-Marquardt a été conçu pour approcher la matrice Hessienne qui ne peut être calculée dans tous les cas, d'où le grand avantage de cette méthode.
L'algorithme d'apprentissage choisi est celui de Levenberg-Marquardt. La principale motivation du choix de l'algorithme de Levenberg-Marquardt (LM) repose sur la taille de la matrice Hessienne en fonction de la quantité de données de la base d'apprentissage, du coût moindre des calculs et de la garantie rapide de la convergence vers un minimum. En effet, l'algorithme de Levenberg-Marquardt avec deux couches cachées peut approximer n'importe quelle fonction (avec un nombre fini de discontinuités) avec l'exactitude désirée à condition que les couches cachées contiennent suffisamment de neurones.

- **Algorithme de L-M**

L'algorithme de rétropropagation de base [33] est fondé sur le principe de la descente du gradient selon lequel les paramètres du réseau tel les poids de pondération et les seuils d'activation (biais), sont ajustés dans la direction opposée de celle du gradient de la fonction d'erreur. Souvent et en pratique, l'utilisation de cette méthode ne suffit pas à elle seule pour garantir un bon apprentissage du réseau.
Plusieurs algorithmes d'apprentissage ont été développés dans le but d'améliorer les performances du processus d'entrainement du réseau, et ce en utilisant des techniques d'optimisation du deuxième ordre basées sur la méthode de Newton.
Ces algorithmes présentent au moins deux (02) avantages par rapport à celui de la descente du gradient, à savoir : une convergence plus rapide des paramètres du réseau

et une valeur minimale de la fonction d'erreur beaucoup plus faible. Parmi ces algorithmes on retrouve ceux de Levenberg-Marquardt.
Le principe de l'algorithme de rétropropagation consiste à propager l'information provenant de la couche d'entrée vers la sortie. Un vecteur de données est présenté au réseau et en le traversant, toutes les valeurs élémentaires subissent simultanément des transformations. En sortie, on obtient un vecteur qui est comparé à une sortie désirée. Cette comparaison permet de corriger et d'adapter les poids du réseau en propageant l'erreur de propagation obtenue de la couche de sortie vers la couche d'entrée. Le mécanisme de propagation et de rétropropagation est répété jusqu'à l'obtention d'un résultat satisfaisant. C'est à dire jusqu'à ce que l'erreur quadratique totale descende sous un seuil présélectionné.
La règle d'adaptation des poids dans l'algorithme standard est donnée par:

$$\omega ij\,(k+1) = \omega ij\,(k) + \Delta\omega ij\,(k) \qquad (4.27)$$

La règle d'adaptation des poids dans l'algorithme de Levenberg-Marquardt est donnée par:

$$\Delta\omega = (J^T J + \mu\, I)^{-1} J^T e \qquad (4.28)$$

Où J est la matrice Jacobienne des dérivées de l'erreur, μ est un scalaire et e le vecteur d'erreur.
L'algorithme de rétropropagation du gradient se résume aux étapes suivantes:

- Choix de l'architecture du réseau et initialiser les paramètres du réseau;
- Choix aléatoire d'un couple (input-output) dans la base d'apprentissage;
- Calcul des sorties des différentes couches pour le vecteur d'entrée choisi à l'aide de la sommation pondérée et d'une fonction d'activation appropriée.
- Mise à jour des poids:
    - de la dernière couche en utilisant la relation appropriée,
    - des couches précédentes en utilisant la relation appropriée,
- si le critère d'arrêt n'est pas satisfait retourner à l'étape 2.

Les deux autres algorithmes utilisent les mêmes étapes. Ils se distinguent seulement par leur façon d'adapter les poids du réseau.
Lorsqu'on constate que le RNA converge vers une erreur quadratique acceptable, les poids synaptiques Wij et les biais bj optimaux sont sauvegardés. La phase de généralisation est ensuite lancée. Si on constate que le réseau généralise mal, le processus d'apprentissage est repris avec une nouvelle architecture du réseau.

IV.4.6. Validation

Une fois le réseau de neurones entraîné (après apprentissage), il est nécessaire de le tester sur une base de données différente de celles utilisées pour l'apprentissage ou la validation croisée. Ce test permet à la fois d'apprécier les performances du système neuronal et de détecter le type de données qui pose problème. Si les performances ne sont pas

satisfaisantes, il faudra soit modifier l'architecture du réseau, soit modifier la base d'apprentissage (caractéristiques discriminantes ou représentativité des données de chaque classe).

Si le réseau n'est pas suffisamment précis et pour améliorer les résultats qu'il donne :

- Initialiser le réseau et lancer l'apprentissage une seconde fois ;
- Augmenter le nombre des neurones cachés (aux environs de vingt (20) neurones) ;
- Essayer un autre algorithme d'apprentissage ;
- Fournir plus de données au réseau.

IV.5. Conclusion :

Le réseau final choisi est un réseau d'Elman, ayant deux couches cachées de type tangente sigmoïde contenant chacune 21 neurones, et une couche de sortie de type linéaire. Son apprentissage est réalisé par l'algorithme de Lavenberg-Marquardt.

La fonction de performance du réseau est l'erreur quadratique moyenne (MSE) et l'erreur tolérée est de 10^{-5}.

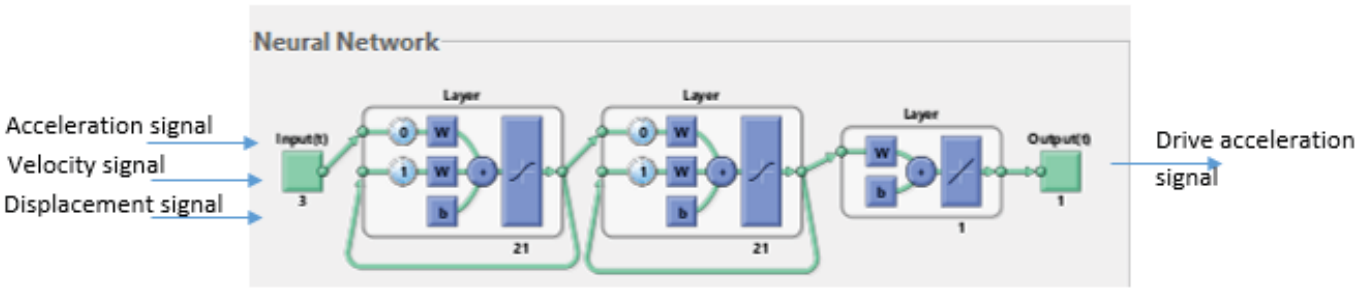


Figure 4.14. Structure du réseau de neurones récurrent utilisé.

Chapitre V : Performance du RNA comme système de contrôle

V.1. Introduction

L'utilisation des réseaux de neurones artificiels s'étend de plus en plus dans le domaine de contrôle des systèmes mécaniques. Dans ce présent chapitre, la performance du réseau de neurones choisi pour le contrôle de la table vibrante sera testée. Le RNA sera utilisé en premier lieu comme seul contrôleur de la table pour minimiser le signal de distorsion entre le signal de réponse de la table et le signal désiré. Ensuite, il sera utilisé pour aider à améliorer la performance d'un contrôleur online choisi dans ce cas comme un PID contrôleur pour des raisons pratiques et simplificatrices.

V.2. Performance du RNA pour un système sans contrôleur online

V.2.1. Description des modèles numériques utilisés :

Afin de simplifier le modèle de la table, tout en maintenant les caractéristiques dynamiques proches de la réalité, une première modélisation sera faite en éléments finis par le logiciel SAP2000. Les composants de la table modélisés sont la plateforme, les vérins et le spécimen.

La plateforme est modélisée par des éléments bidimensionnels plaques épaisses, en acier, d'une masse totale de 50 tonnes et une épaisseur d'1m, ayant des fréquences propres fondamentales comprise entre 90 et 100 Hz.

Dans le but de distinguer les sources de distorsion, six (06) modèles en éléments finis ont été construits :

- Le modèle N°1 représente la table non chargée. Il comporte des vérins linéaires qui ont une fréquence propre de 16.4 Hz, une rigidité de 60000KN/m équivalente à celle de la colonne d'huile comprimée et un coefficient d'amortissement ξ=15% [3].

  La fréquence propre fondamentale devient alors égale à 3.37Hz.

- Le modèle N°2 représente la table non chargée. Il comporte des vérins ayant un amortissement non linéaire (Coulomb et visqueux non linéaire) « damper » avec une rigidité de 60000 KN/m, un coefficient d'amortissement $C=40KN(m/s)^{-1/2}$. Concernant l'amortissement exponentiel α, il doit être positif et compris entre 0.1 et 2. S'il est égal à 1, l'amortissement devient alors visqueux linéaire, plus il s'éloigne de 1 et plus la non linéarité devient forte [8]. La valeur de α introduite dans le modèle est de 0.5 [64].

  La fréquence propre fondamentale devient alors égale à 5.81Hz.

- Le modèle N°3 représente la table chargée avec un spécimen ayant un comportement linéaire d'une masse de 20 tonnes et une fréquence propre de 13Hz. Il comporte des vérins linéaires avec les mêmes caractéristiques décrites dans le modèle N°1.

  La fréquence propre fondamentale devient alors égale à 2Hz.

- Le modèle N°4 représente également la table chargée avec un spécimen linéaire rigide, d'une fréquence fondamentale de 13Hz, mais les non linéarités des vérins hydrauliques seront prises en compte. Leurs caractéristiques sont les mêmes que dans le modèle N°2.

  La fréquence propre fondamentale devient alors égale à 2.44Hz.

- Le modèle N°5, comme le N°4 prend en compte les non linéarités des vérins avec les mêmes caractéristiques que dans le modèle N°2, mais représentera la table chargée avec un spécimen non linéaire rigide d'une fréquence fondamentale de 13 Hz. La non-linéarité introduite dans le spécimen est des rotules plastiques (modèle hystérésis plastic) à la base du spécimen.

  La fréquence propre fondamentale devient alors égale à 2.21Hz.

- Le modèle N°6, comme le N°5 prend en compte les non linéarités des vérins avec les mêmes caractéristiques que dans le modèle N°2, mais représentera la table chargée avec un spécimen non linéaire flexible d'une fréquence fondamentale de 4.4Hz. La non-linéarité introduite dans le spécimen est des rotules plastiques (modèle hystérésis plastic) à la base du spécimen.

  La fréquence propre fondamentale devient alors égale à 1.52Hz.

Une série d'analyses dynamiques (linéaires et non linéaires) sera faite sur ces modèles afin de construire la base de données qui servira à l'apprentissage du réseau mais aussi à son test et à sa validation.

Les signaux qui serviront d'input au réseau sont les réponses de la plateforme en termes d'accélération, de vitesse et de déplacement (signal recueillis au CDG de celle-ci) et le signal cible (target signal) du réseau sera le signal de commande appliqué au niveau des vérins (à l'extrémité fixe du vérin).

Les accélérogrammes utilisés sont l'accélérogramme de Kedara ou de Méliana pour l'apprentissage. Ils comportent 4000 points, ce qui représente un nombre suffisant pour l'apprentissage du RNA.

Quant au test et à la validation du RNA, différents types d'accélérogrammes du champ proche ou lointain seront utilisés comme ceux de Dar El Beida, Azazga, Blida, Hussein Dey, Meliana, El Centro, Kobe ainsi qu'une accélération sinusoïdale.

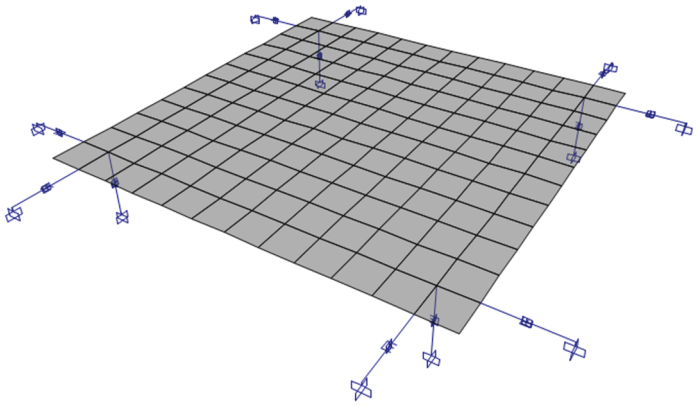


Figure 5.1. Table non chargée avec vérins linéaires ou non linéaires (linear/nonlinear damper).

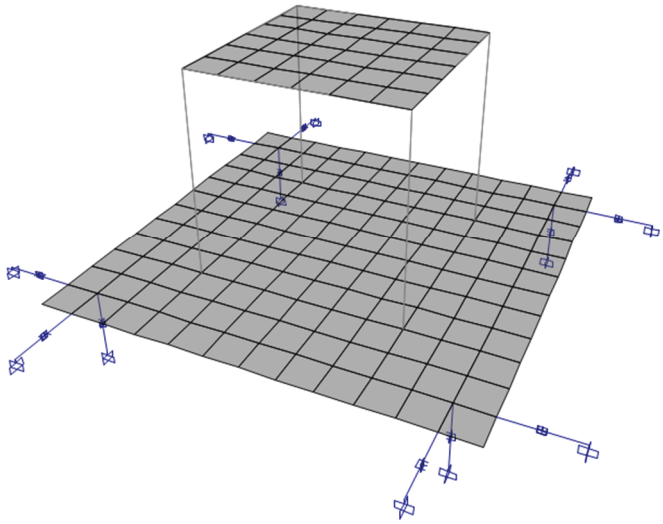


Figure 5.2. Table chargée (spécimen linéaire) avec vérins linéaires ou non linéaires.

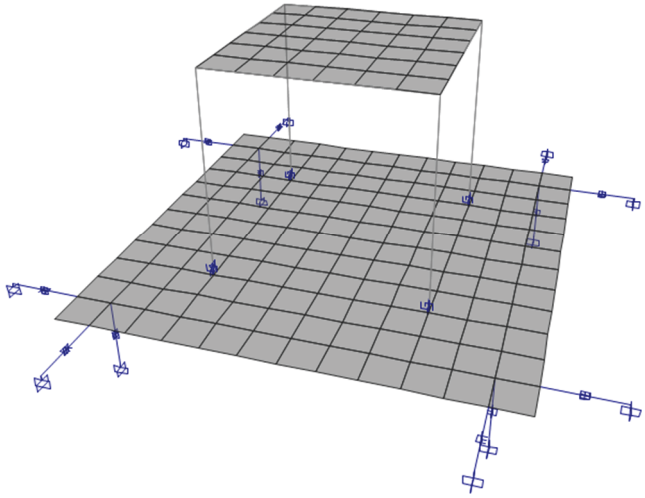


Figure 5.3. Table chargée avec spécimen non linéaire (plastic), vérins non linéaires (nonlinear damper).

V.2.2. Evaluation de la performance du système de contrôle

On rappelle que la base de données à constituer à partir du modèle en EF suppose que la table vibrante ne dispose d'aucun système de contrôle.

Le signal de commande est appliqué directement aux niveaux des points fixes des vérins et le signal de réponse est recueilli au CDG de la plateforme.

V.2.2.1. Modèle N° 1 : système linéaire sans spécimen

La distorsion de signal du modèle N°1 est mise en avant par la figure suivante pour l'accélérogramme d'El Centro:

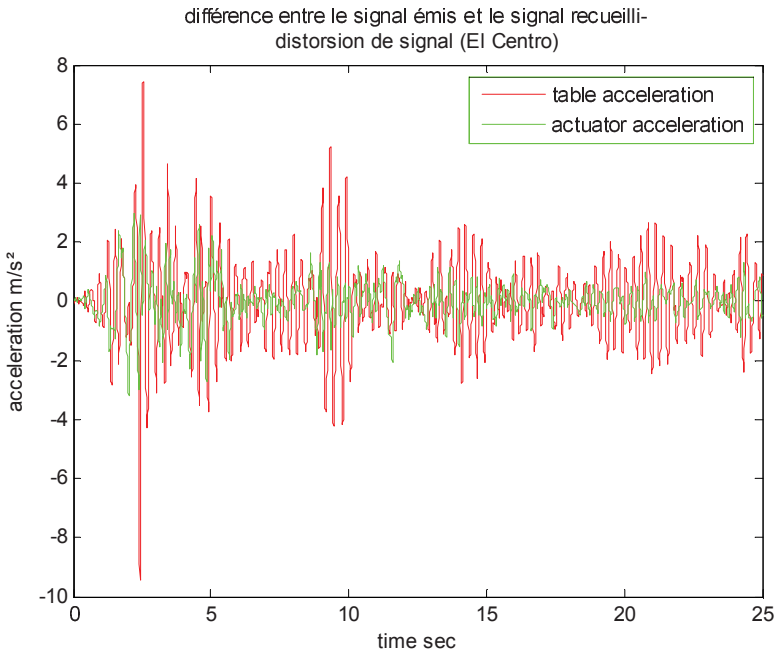


Figure 5.4. Mise en évidence de la distorsion de signal (différence entre le signal de réponse et le signal cible) du modèle N°1.

Le contrôle par RNA vise à supprimer cette distorsion qui fait obstacle à la performance de la table vibrante.

Afin de vérifier le bon fonctionnement des réseaux de neurones et leur capacité de généralisation, l'ensemble des données utilisées pour le test et la validation a été présenté au réseau pour effectuer une régression linéaire entre les sorties du réseau (réponses obtenus Y) et les cibles correspondantes (réponses exigées T (target)).

L'apprentissage du réseau est fait avec l'accélérogramme de Kedara.

La performance d'apprentissage du réseau montre qu'il prédit le signal désiré à une erreur quadratique moyenne de 8.53 10^{-7}, comme le montre la figure suivante :

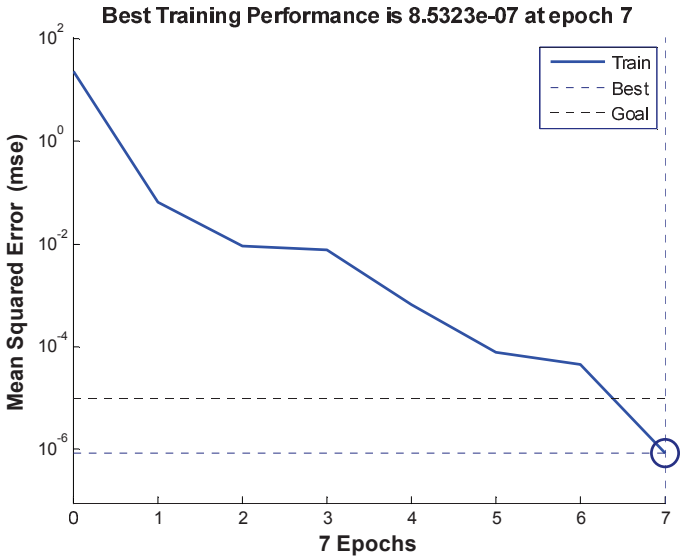


Figure 5.5. L'erreur quadratique moyenne entre le signal de réponse et le signal cible du modèle N°1.

La performance du réseau de neurones atteinte est estimée à partir du coefficient de corrélation obtenu R=0.99 indiquant une prédiction quasi parfaite du réseau (Y=T). Aussi, l'erreur quadratique moyenne qui est de l'ordre de 10^{-6} ce qui est plus qu'admissible.

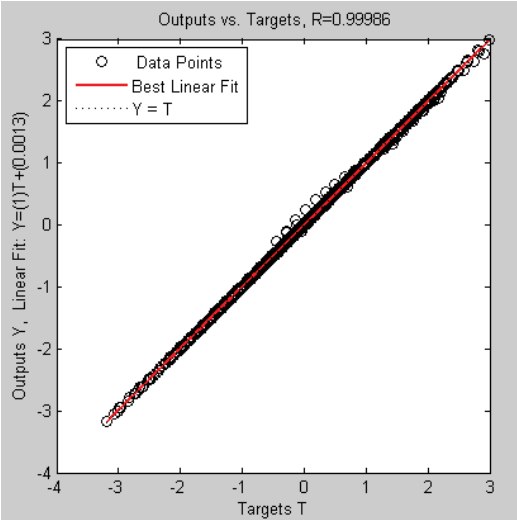


Figure5.6. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°1 (accélérogramme d'El Centro).

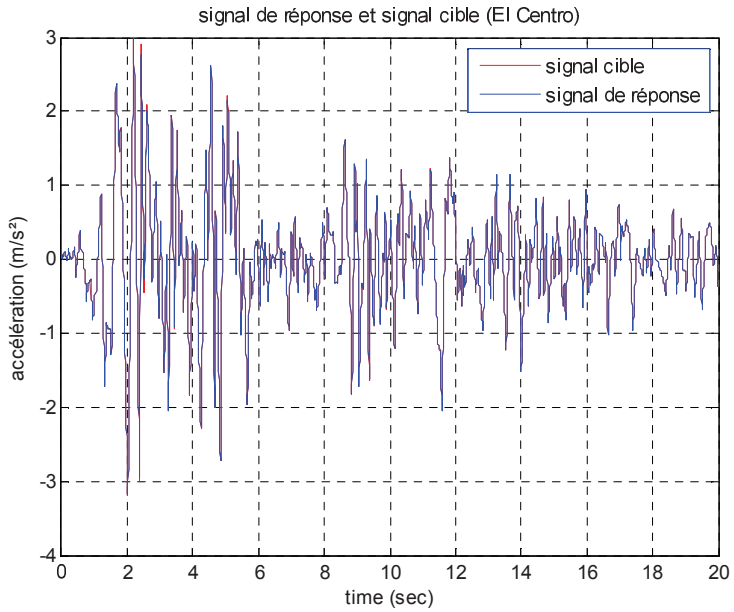


Figure 5.7. Signal de réponse et signal cible du modèle N°1.

V.2.2.2. Modèle N°2 : système non linéaire sans spécimen

La distorsion de signal du modèle N°2 est mise en avant par la figure suivante pour l'accélérogramme d'El Centro:

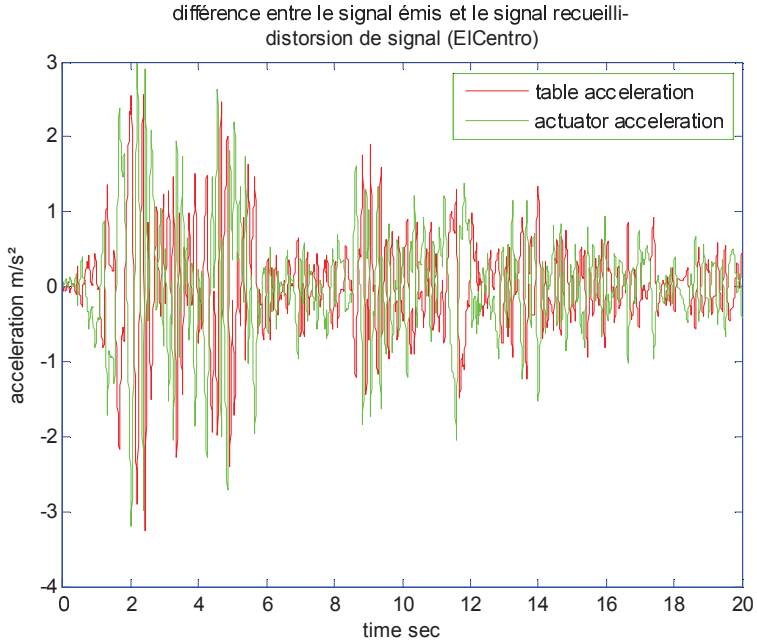


Figure 5.8. Mise en évidence de la distorsion de signal (différence entre le signal de réponse et le signal cible) du modèle N°2.

L'apprentissage du réseau se fait avec l'accélérogramme de Méliana. La performance d'apprentissage du réseau montre qu'il prédit le signal désiré à une erreur quadratique moyenne de 9.97 10^{-6}, comme le montre la figure suivante :

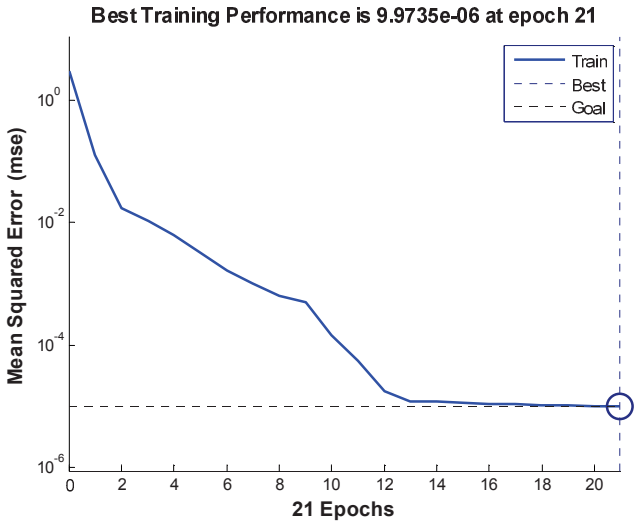


Figure 5.9. L'erreur quadratique moyenne entre le signal de réponse et le signal cible du modèle N°2.

La performance du réseau de neurones atteinte est estimée à partir du coefficient de corrélation obtenu R=0.99 indiquant une prédiction quasi parfaite du réseau (Y=T). Aussi, l'erreur quadratique moyenne qui est de l'ordre de 10^{-6} ce qui est plus qu'admissible.

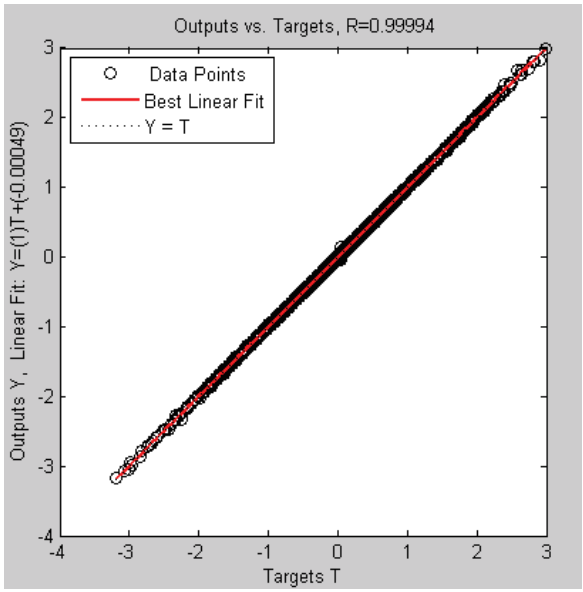


Figure 5.10. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°2 accélérogramme d'El Centro).

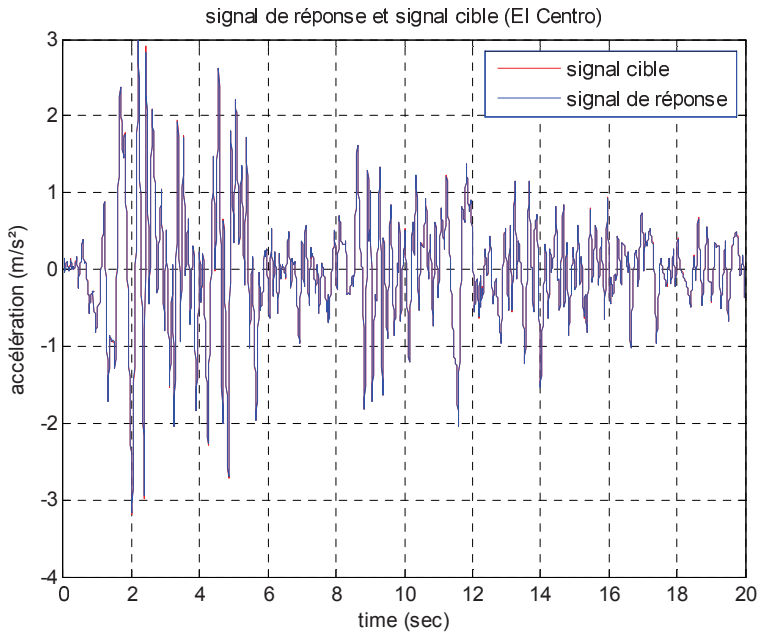


Figure 5.11. Signal de réponse et signal cible du modèle N°2.

La performance du réseau obtenue signifie que le réseau est capable de prédire un signal de réponse très proche du signal cible (avec une erreur quadratique moyenne de 9.66 10^{-6}) même lorsque les vérins exhibent un comportement non linéaire. L'incursion de ces derniers dans le domaine non linéaire est représentée sur la courbe d'hystérésis suivante :

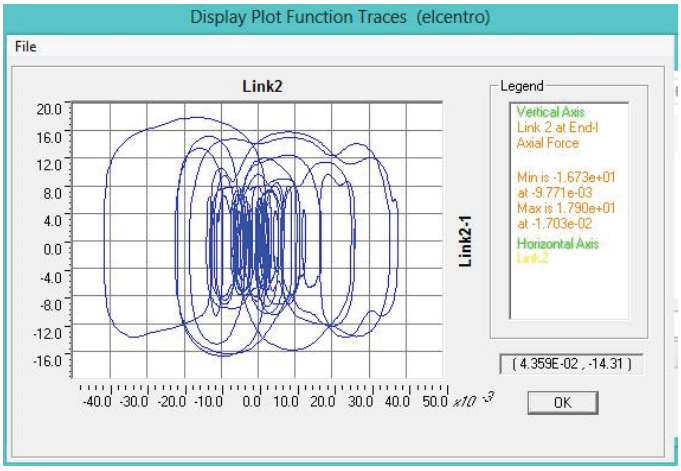


Figure 5.12. Courbe force-déplacement de vérin horizontal de la table vibrante.

L'allure de la courbe d'hystérésis des vérins est spécifique à une dissipation d'énergie par amortissement typique de Coulomb dans la colonne d'huile des vérins [3, 64].

V.2.2.3. Modèle N° 3 : système linéaire avec spécimen rigide linéaire

La distorsion de signal du modèle N°3 est mise en avant par la figure suivante pour l'accélérogramme d'El Centro :

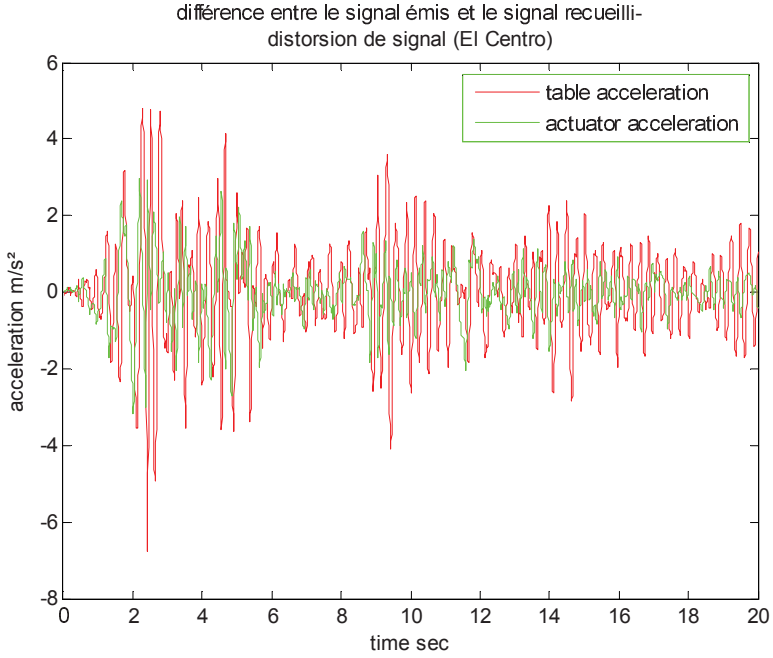


Figure 5.13. Mise en évidence de la distorsion de signal (différence entre le signal de réponse et le signal cible) du modèle N°3.

L'apprentissage du réseau est fait avec l'accélérogramme de Kedara.

La performance d'apprentissage du réseau montre qu'il prédit le signal désiré à une erreur quadratique moyenne de 9.66 10^{-6}, comme le montre la figure suivante :

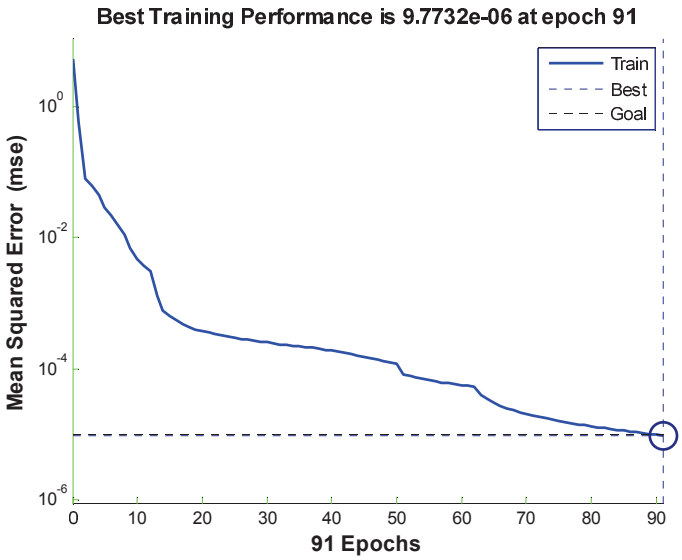


Figure 5.14. L'erreur quadratique moyenne entre le signal de réponse et le signal cible du modèle N°3.

La performance du réseau de neurones obtenue est estimée à partir du coefficient de corrélation obtenu R=0.99 indiquant une prédiction quasi parfaite du réseau. Aussi, l'erreur quadratique moyenne qui est de l'ordre de 10^{-6} ce qui est plus qu'admissible.

Ceci indique que le réseau de neurones est capable de pallier le problème de l'interaction table-spécimen, et ce dans le domaine linéaire.

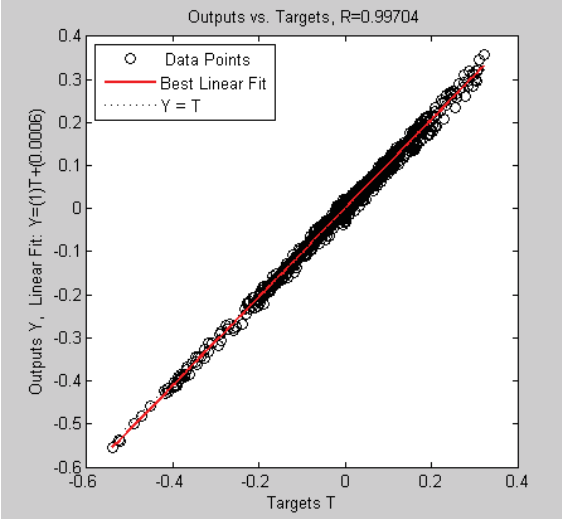


Figure 5.15. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°3 (accélérogramme d'El Centro).

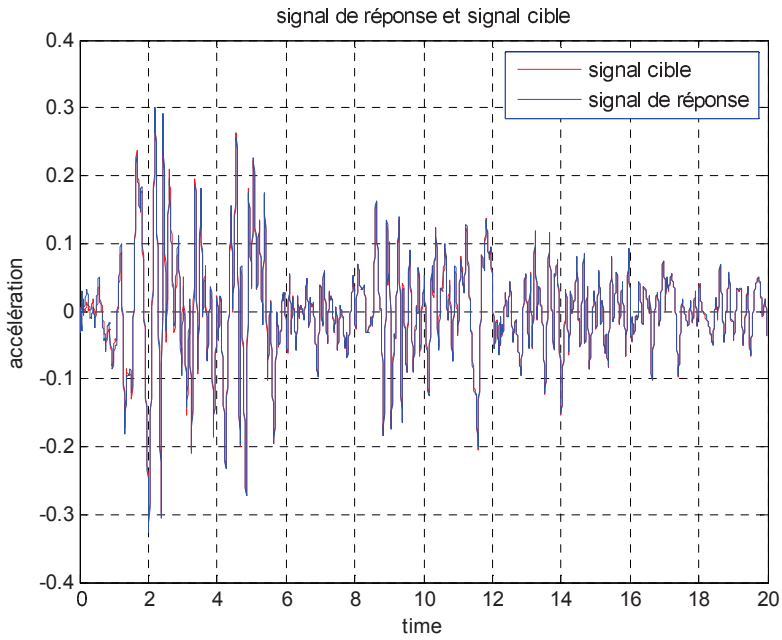


Figure 5.16. Signal de réponse et signal cible du modèle N°3.

V.2.2.4. Modèle N° 4 : système non linéaire avec spécimen rigide linéaire

La distorsion de signal du modèle N°4 est mise en avant par la figure suivante pour l'accélérogramme d'El Centro :

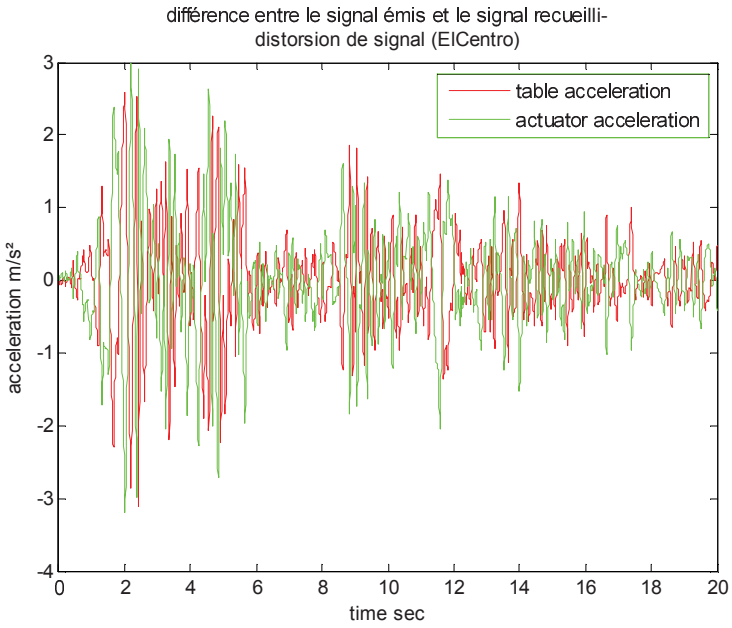


Figure 5.17. Mise en évidence de la distorsion de signal (différence entre le signal de réponse et le signal cible) du modèle N°4.

L'apprentissage du réseau se fait avec l'accélérogramme de Méliana.

La performance d'apprentissage du réseau montre qu'il prédit le signal désiré à une erreur quadratique moyenne de 9.76 10^{-5}, comme le montre la figure suivante :

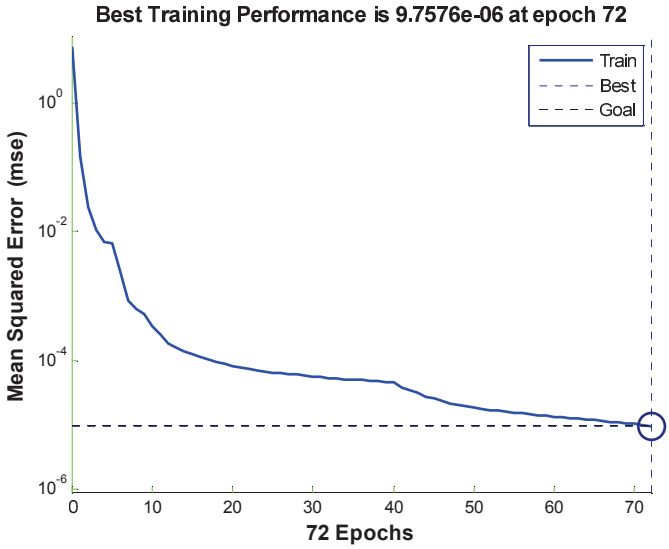


Figure 5.18. L'erreur quadratique moyenne entre le signal de réponse et le signal cible du modèle N°4.

La performance du réseau de neurones obtenue est estimée à partir du coefficient de corrélation obtenu R=0.99 indiquant une prédiction quasi parfaite du réseau. Aussi, l'erreur quadratique moyenne qui est de l'ordre de 10^{-6} ce qui est plus qu'admissible.

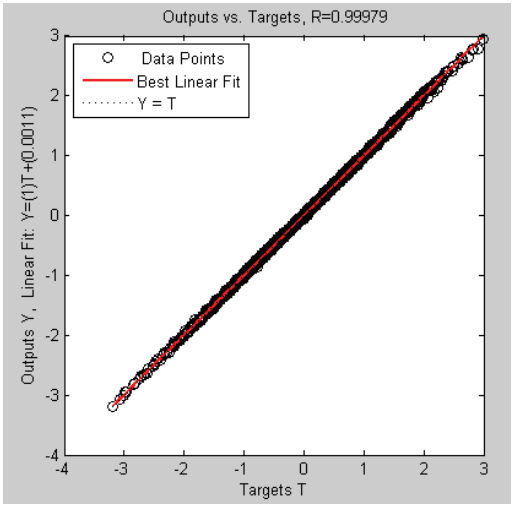


Figure 5.19. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°4 (accélérogramme d'El Centro).

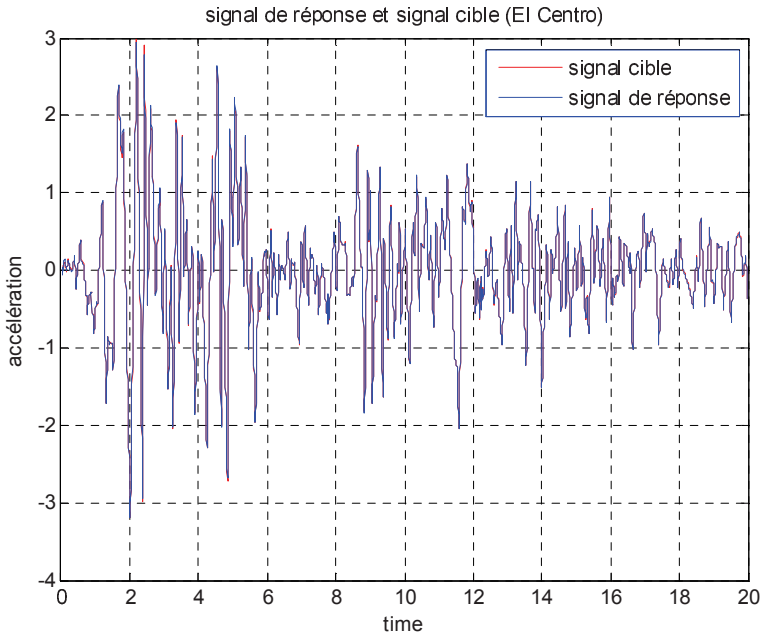


Figure 5.20. Signal de réponse et signal cible du modèle N°4.

La performance du réseau obtenue signifie que le réseau est capable de prédire un signal de réponse très proche du signal cible même pour une table chargée (palier le problème d'interaction table-spécimen) et ce même dans le cas où les vérins exhibent un comportement non linéaire.

L'incursion de ces derniers dans le domaine non linéaire est représentée sur la courbe d'hystérésis suivante :

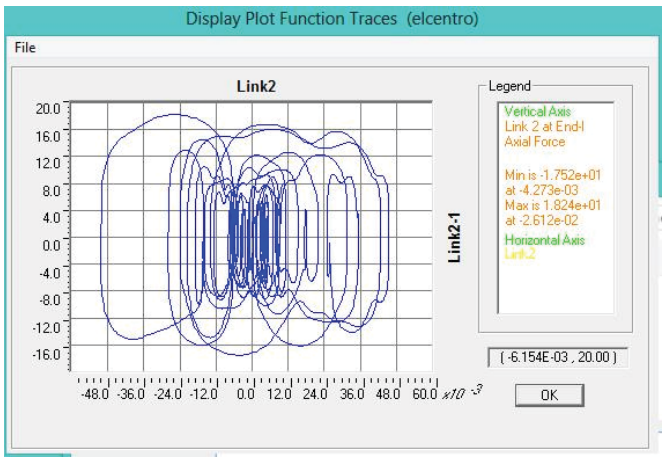


Figure 5.21. Courbe force-déplacement de vérin horizontal de la table vibrante.

V.2.2.5. Modèle N° 5 : système non linéaire avec spécimen rigide non linéaire

L'apprentissage du réseau se fait avec l'accélérogramme de Méliana.

La performance d'apprentissage du réseau montre qu'il prédit le signal désiré à une erreur quadratique moyenne de 9.97 10^{-6}, comme le montre la figure suivante :

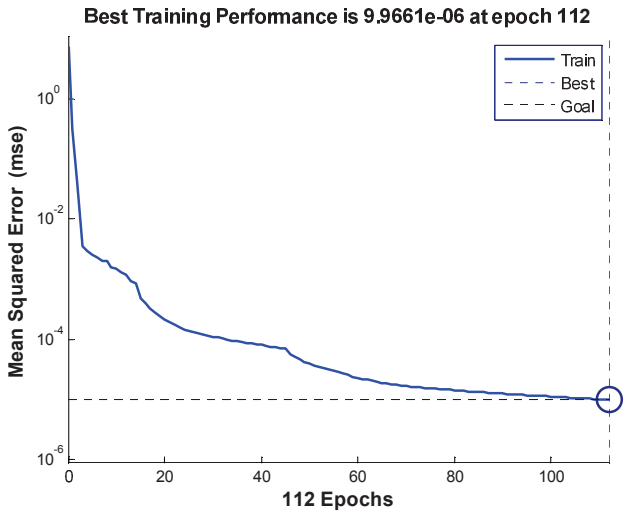


Figure 5.22. L'erreur quadratique moyenne entre le signal de réponse et le signal cible du modèle N°5.

La performance du réseau de neurones obtenue est estimée à partir du coefficient de corrélation obtenu R=0.99 indiquant une très bonne prédiction du réseau. Aussi, l'erreur quadratique moyenne qui est de l'ordre de 10^{-5} ce qui est admissible.

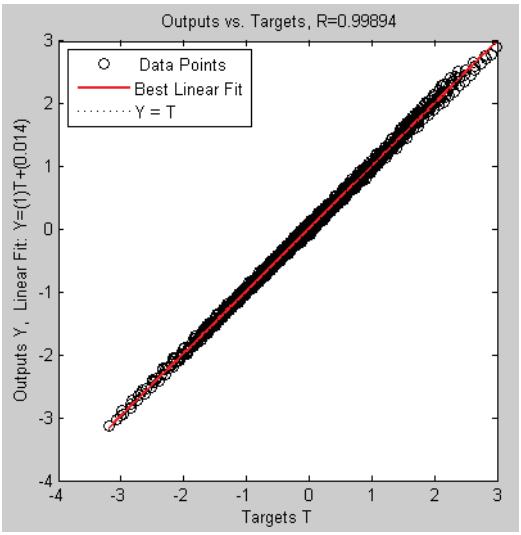


Figure 5.23. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°5.

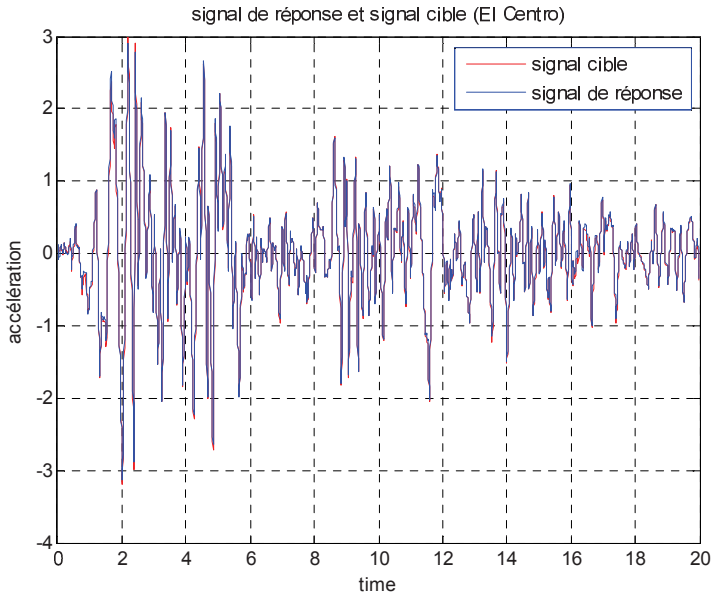


Figure 5.24. Signal de réponse et signal cible du modèle N°5.

La performance du réseau obtenue signifie que le réseau est capable de prédire un signal de réponse très proche du signal cible même pour une table chargée (palier le problème d'interaction table-spécimen) et ce même dans le cas où les vérins exhibent un comportement non linéaire. Aussi, le réseau prédit un signal désiré même quand le spécimen monté sur la table peut présenter une formation de rotules plastiques en sa base pendant le test.

L'incursion de ces derniers dans le domaine non linéaire est représentée sur la courbe d'hystérésis suivante :

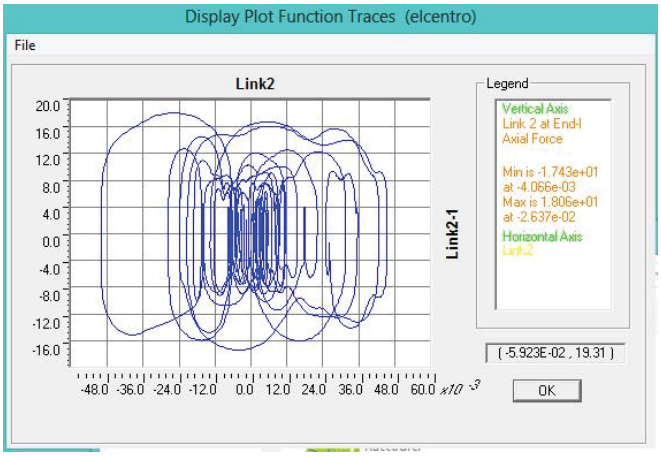


Figure 5.25. Courbe force-déplacement de vérin horizontal de la table vibrante.

L'incursion du spécimen dans le domaine non linéaire est représentée sur la courbe d'hystérésis suivante :

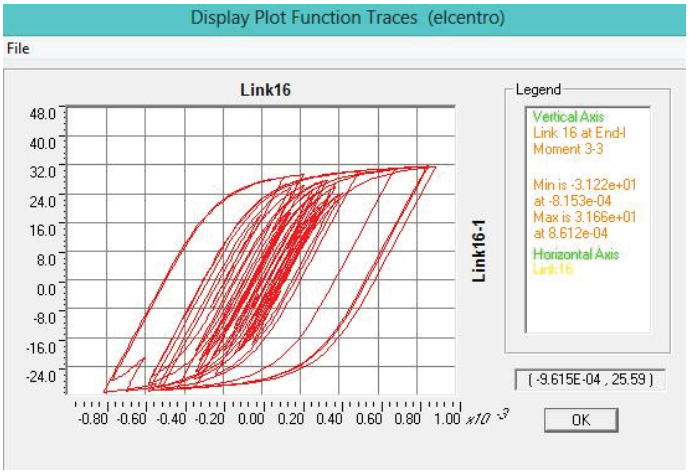


Figure 5.26. Courbe force-déplacement de la base du spécimen.

V.2.2.6. Modèle N° 6 : système non linéaire avec spécimen flexible non linéaire

L'apprentissage du réseau se fait avec l'accélérogramme de Méliana.

La performance d'apprentissage du réseau montre qu'il prédit le signal désiré à une erreur quadratique moyenne de 9.81 10^{-6}, comme le montre la figure suivante :

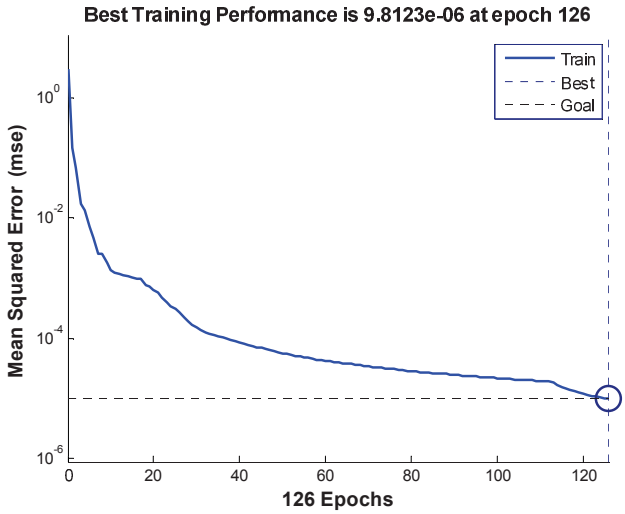


Figure 5.27. L'erreur quadratique moyenne entre le signal de réponse et le signal cible du modèle N°6.

La performance du réseau de neurones obtenue est estimée à partir du coefficient de corrélation obtenu R=0.99 indiquant une très bonne prédiction du réseau. Aussi, l'erreur quadratique moyenne qui est de l'ordre de 10^{-5} ce qui est admissible.

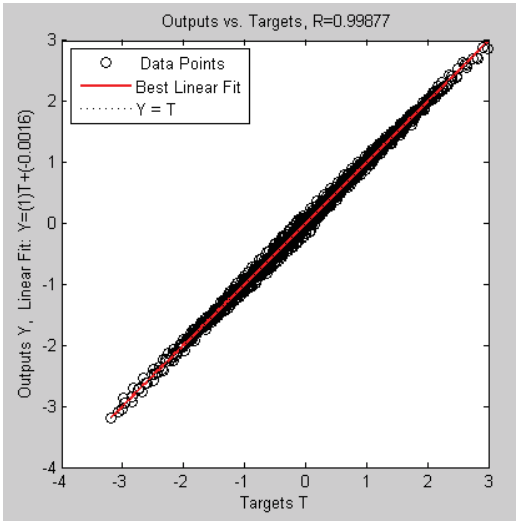


Figure 5.28. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°6.

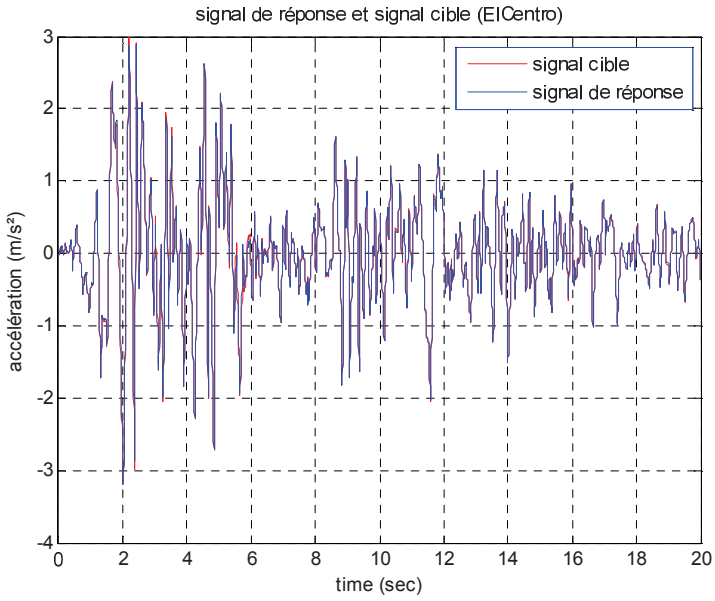


Figure 5.29. Signal de réponse et signal cible du modèle N°6.

La performance du réseau obtenue signifie que le réseau est capable de prédire un signal de réponse très proche du signal cible même pour une table chargée avec un spécimen flexible (palier le problème d'interaction table-spécimen) et ce même dans le cas où les vérins exhibent un comportement non linéaire. Aussi, le réseau prédit un signal désiré même quand le spécimen monté sur la table peut présenter une formation de rotules plastiques en sa base pendant le test.

L'incursion de ces derniers dans le domaine non linéaire est représentée sur la courbe d'hystérésis suivante :

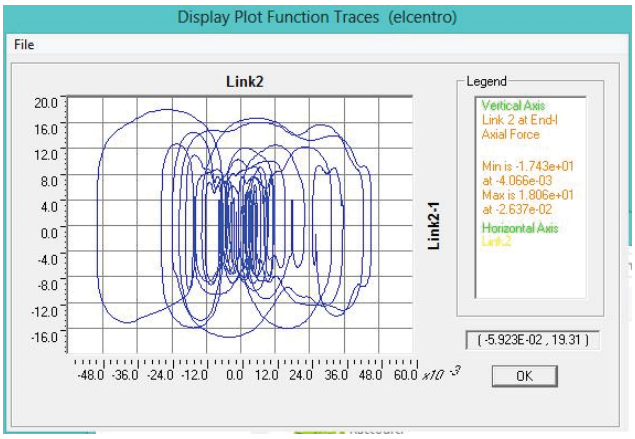


Figure 5.30. Courbe force-déplacement de vérin horizontal de la table vibrante.

L'incursion du spécimen dans le domaine non linéaire est représentée sur la courbe d'hystérésis suivante :

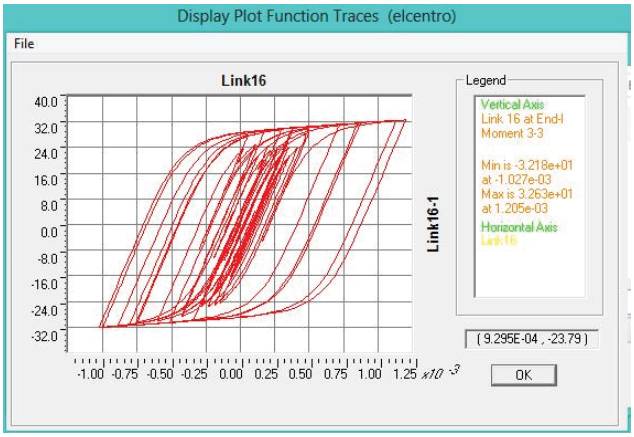


Figure 5.31. Courbe force-déplacement de la base du spécimen.

Les résultats de test et de validation du réseau pour la prédiction du signal de commande sont représentés en Annexe 1.

V.2.3. Essai d'application du système de contrôle par RNA

Le principe de cette stratégie de contrôle est d'introduire le signal prédit par le RNA comme signal d'excitation de la table (signal de commande) de telle sorte à ce que le signal de réponse de celle-ci soit égal au signal désiré.

Pour ce faire, un input sera présenté au réseau comportant l'accélération, la vitesse et le déplacement désiré. L'output du réseau sera le signal qui excitera la table (signal de commande appliqué aux vérins). À la suite d'une analyse dynamique, le signal de réponse de la table en accélération sera comparé au signal désiré (accélérogramme réel).

Le test a été fait sur le modèle N°5 comprenant un spécimen non linéaire avec une modélisation non linéaire des vérins de la table. Le signal utilisé est l'accélérogramme d'ElCentro.

La procédure de correction des signaux par RNA est illustrée dans le schéma suivant :

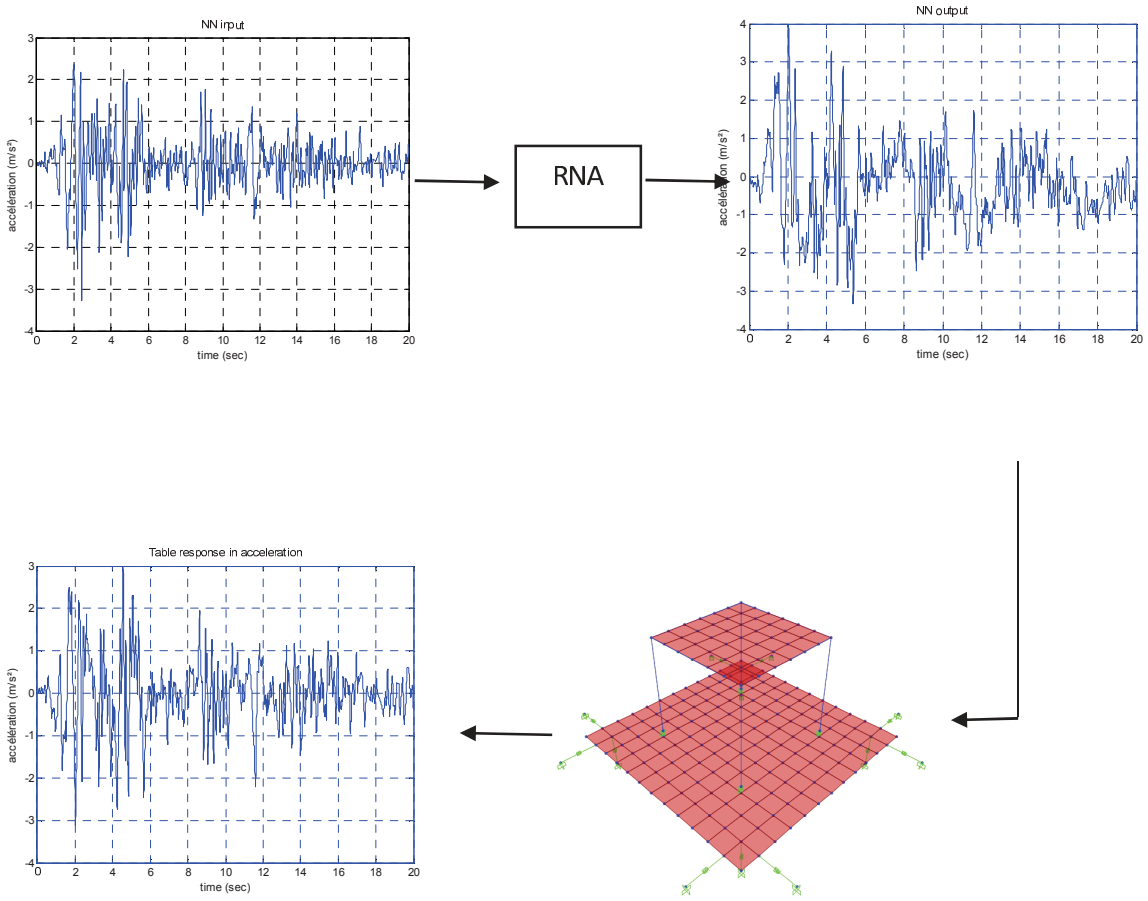


Figure 5.32. Schéma illustratif de la procédure de contrôle par RNA

La comparaison entre la réponse de la table et le signal désiré (accélérogramme d'El Centro) est montrée dans la figure suivante :

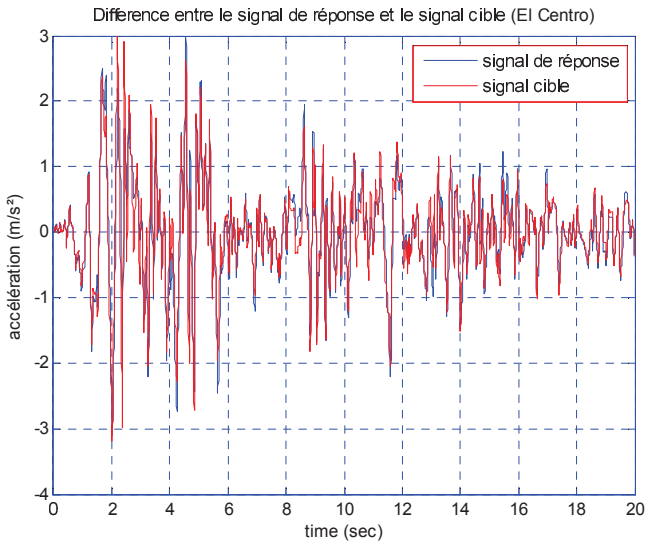


Figure 5.33. Signal de réponse de la table et signal cible (El Centro).

La performance du RNA pour un système sans contrôleur montre que le réseau proposé peut prédire avec une très grande précision le signal adéquat qui, une fois envoyé au système de commande, produit le signal désiré (l'accélérogramme réel par exemple) au niveau de la plateforme et de ce fait, le spécimen sera réellement excité par l'accélérogramme désiré et non l'accélérogramme distordu.

V.3. Performance du RNA pour un système avec contrôleur

V.3.1. Description des modèles numériques utilisés :

Afin de modéliser le comportement de la table vibrante et ses équipements de manière plus réaliste, un autre modèle réalisé sous Matlab a été construit avec essentiellement l'ajout de la servovalve et du contrôleur online. Comme le montre la figure 4.9 du chapitre précédent, le modèle inclue donc les composants suivants :

- mécaniques de la table (i.e : le bloc de réaction, la plateforme et le spécimen) ;
- hydrauliques (i.e : les servovalves et les vérins)
- et électroniques (i.e : le contrôleur).

Comme mentionné au paravent, étant donné que ce modèle ne servira qu'à la construction de la base de données de l'apprentissage, du test et de la validation du réseau de neurones, et pour des raisons simplificatrices et pratiques, le contrôleur utilisé est le PID.

Ces composants sont modélisés par leurs fonctions de transfert décrites au chapitre IV.

Deux modèles sont réalisés : un premier modélisant la table non chargée et un second modélisant la table chargé d'un spécimen à 1DDL.

Une série d'accélérogrammes en déplacement sera introduite comme input à la fonction de transfert totale de la table, ce sont les signaux de commande. Les signaux de réponse de la table sont alors recueillis.

Les signaux de réponse en déplacement sont convertis en vitesse puis en accélération. Ils représentent l'input du RNA (input à 3 vecteurs), les signaux de commande représenteront les signaux cibles auxquels sera comparé l'output du RNA.

V.3.2. Evaluation de la performance du système de contrôle

La base de données à constituer à partir du modèle sous Matlab de la table suppose que la table vibrante dispose d'un système de contrôle de type PID.

V.3.2.1. Modèle N° 1 : table sans spécimen

Dans cette partie, le contrôle par RNA vise à améliorer la performance du PID (le contrôleur online de la table) et à encore diminuer la distorsion de signal entre le signal de réponse et le signal désiré.

Similairement à la procédure utilisée pour le modèle de la table par éléments finis, l'input du RNA représente la réponse de la fonction de transfert en déplacement, vitesse et accélération, alors que le signal cible auquel serait comparé l'output du RNA est le signal de commande introduit à la fonction de transfert.

L'apprentissage du réseau par les accélérogrammes utilisés dans la première partie, contenant environ 4000points, se sont avérés insuffisant à un bon apprentissage. De ce fait, un accélérogramme artificiel a été généré par Seismoartif software (Seismosoft 2013) contenant 20000points.

La performance d'apprentissage du réseau montre qu'il prédit le signal désiré à une erreur quadratique de l'ordre de 10^{-3} comme le montre la figure suivante :

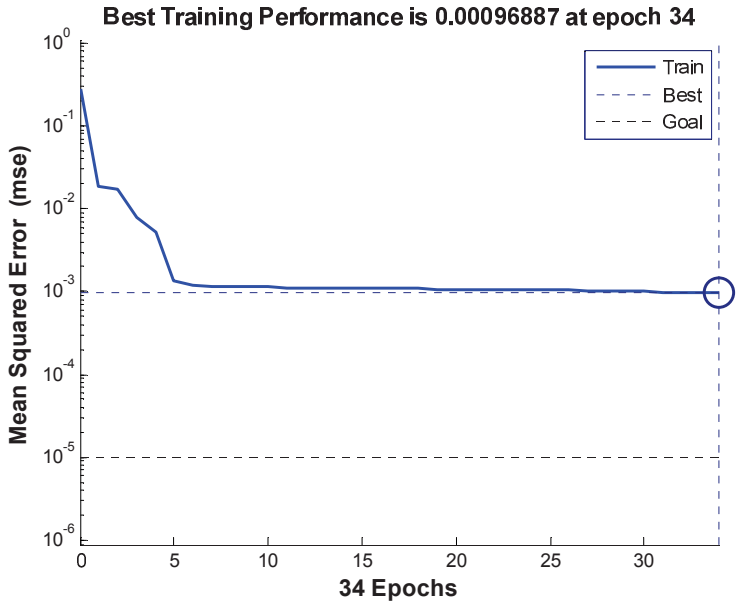


Figure 5.34. L'erreur quadratique moyenne entre le signal de réponse et le signal cible du modèle N°1.

Des régressions linéaires entre les signaux de réponse et les signaux cibles ont été réalisées afin d'évaluer la performance du réseau de neurones. Plusieurs accélérogrammes comme Friuli, Kocaeli et ElCentro ont été utilisés et le coefficient de corrélation obtenu est de R=0.96, 0.94 et 0.94 pour chacun des accélérogrammes cités respectivement.

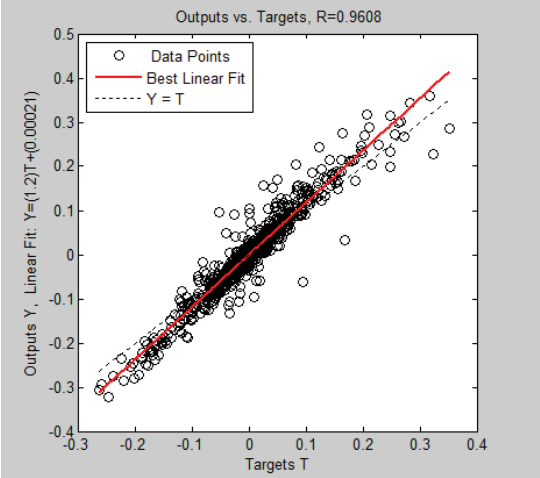


Figure 5.35. La régression linéaire entre la réponse du RNA et le signal cible (Friuli)

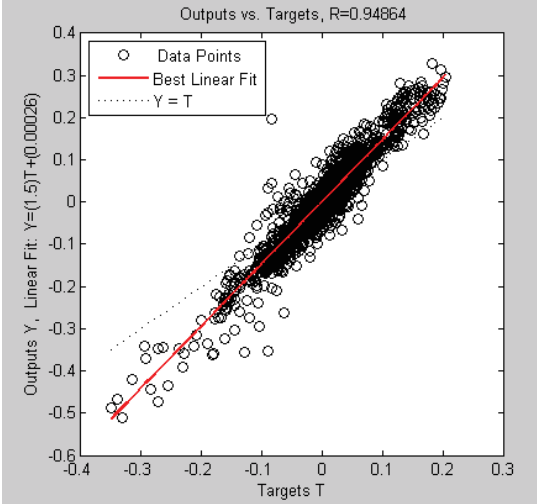


Figure 5.36. La régression linéaire entre la réponse du RNA et le signal cible (Kocaeli)

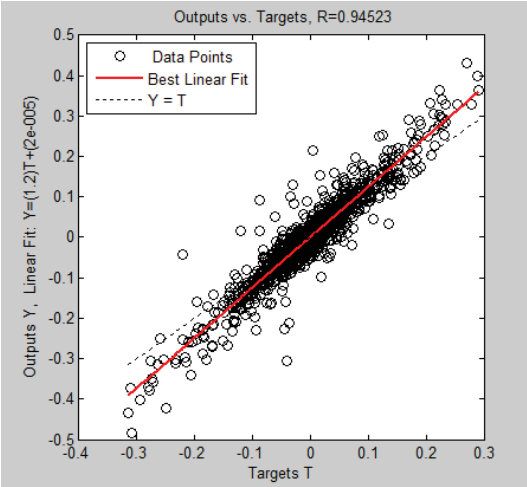


Figure 5.37. La régression linéaire entre la réponse du RNA et le signal cible (El Centro)

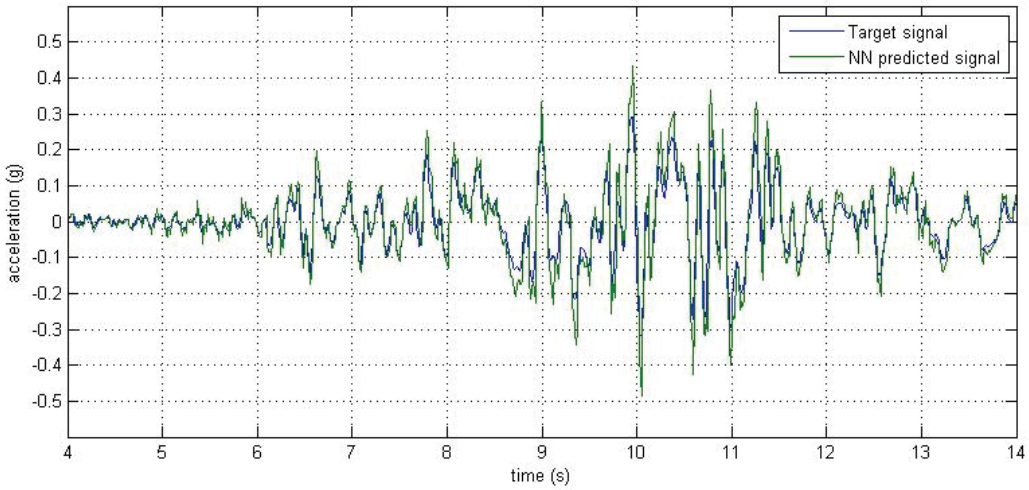


Figure 5.38. Signal de réponse et signal cible (El-Centro)

Le réseau de neurones s'avère moins efficace dans ce cas-là, c'est à dire il s'avère moins efficace en présence du PID que sans contrôleur. Les difficultés que le réseau a rencontrées lors de l'établissement d'une relation input/output sont essentiellement dues au principe de fonctionnement du PID car ce dernier compense l'erreur entre le signal de réponse et le signal cible de manière itérative. Du fait de ce fonctionnement très dépendant du type de chaque signal, le réseau n'a pas atteint une parfaite performance de généralisation. Néanmoins, l'utilisation du RNA avec le PID a donné de meilleurs résultats de correspondance de quelques signaux comme le montre la figure 5.39. Elle représente une comparaison entre le signal cible, le signal de réponse avec correction par PID seulement et le signal de réponse avec correction par PID et RNA.

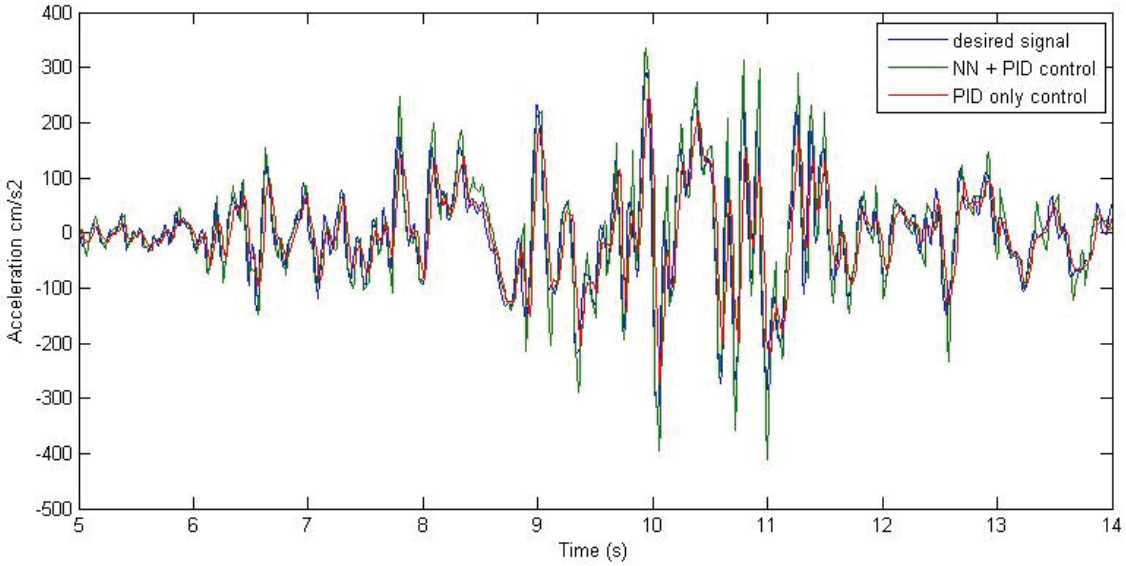


Figure 5.39. Signal désiré, signal de réponse corrigé par PID seul et signal de réponse corrigé par PID+RNA (El Centro) de la table sans spécimen.

V.3.2.2. Modèle N°2 : table avec spécimen à 1DDL.

La procédure suivie pour le modèle N°2 est la même que pour le modèle N°1 de la table non chargée.

L'apprentissage du réseau se fait avec l'accélérogramme artificiel de 200s généré par Seismoartif software. Le test et la validation du réseau se fait par plusieurs accélérogrammes réels tels que l'accélérogramme de Kobe, d'El Centro ou de Friuli.

La régression linéaire entre le signal de réponse et le signal cible a été réalisée afin d'évaluer la performance du réseau de neurones. Le coefficient de corrélation R obtenu pour l'accélérogramme d'El Centro est montré dans la figure suivante :

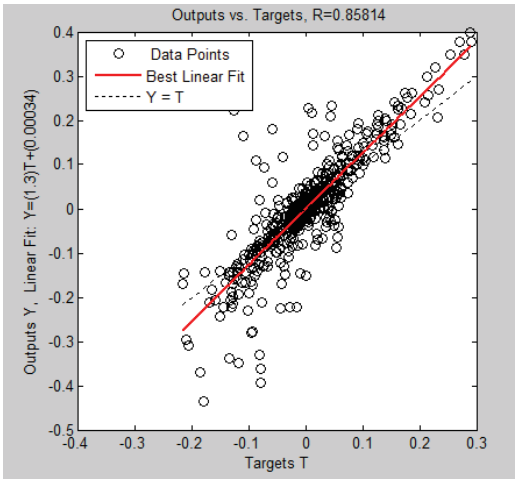


Figure 5.40. La régression linéaire entre la réponse du RNA et le signal cible (El Centro).

La comparaison entre la réponse de la table avec correction par PID, la réponse de la table avec correction par PID+RNA et le signal désiré (accélérogramme d'El Centro) est montrée dans la figure suivante :

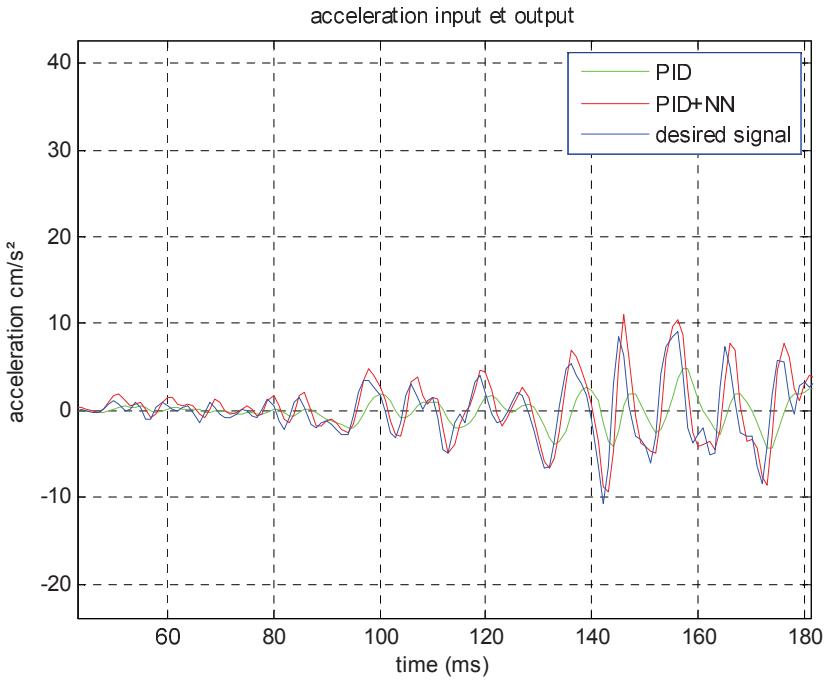


Figure 5.41. Signal désiré, signal de réponse par PID seul et signal de réponse corrigé par PID+RNA (El Centro) de la table avec spécimen à 1DDL.

Comme mentionné précédemment, plusieurs signaux ont été présentés au RNA afin de tester son efficacité quant à l'amélioration de la correction des signaux par PID, que ce soit dans le cas d'une table avec ou sans spécimen. Comme pour l'accélérogramme d'El Centro, de Kobe ou de Friuli, la correction additionnelle par RNA a donné de meilleurs résultats que ceux obtenus uniquement par la correction par PID. Cependant, pour les accélérogrammes présentant de grands déplacements, comme par exemple celui de Keddara ou de Northridge, la correction additionnelle par RNA ne donne pas les résultats attendus, comme le montre la figure 5.42. Le PID étant un contrôleur en déplacement, il corrige alors d'une manière plus efficace les distorsions même en présence de grands déplacements dans le signal d'excitation, ce qui résulte en une perte de précision dans les accélérations données par le RNA. A titre d'exemple, une correction par baseline a été réalisée dans le but de limiter les déplacements importants surtout à la fin du signal d'excitation, les résultats obtenus sont alors meilleurs. Les figures 5.43 et 5.44 montrent les résultats obtenus avec et sans correction baseline.

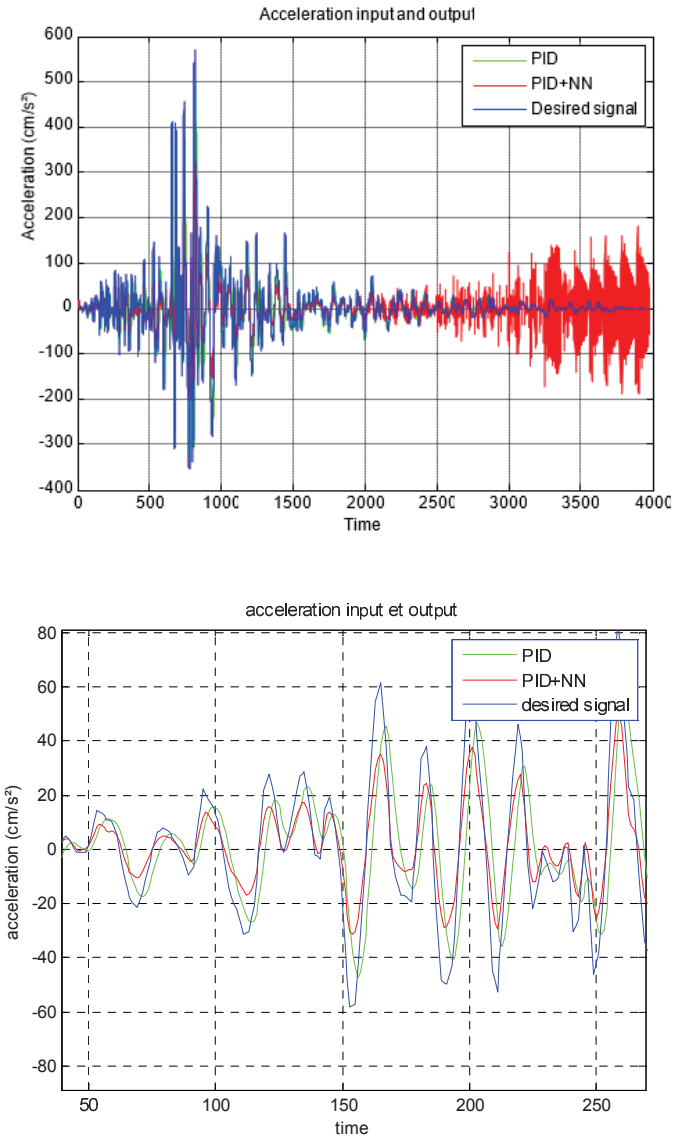
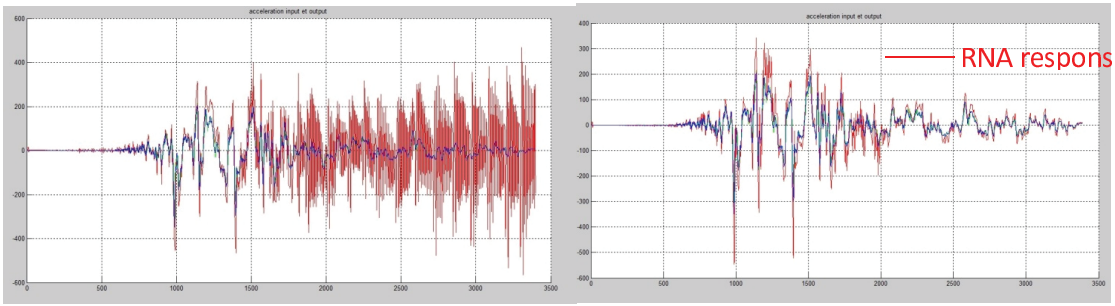
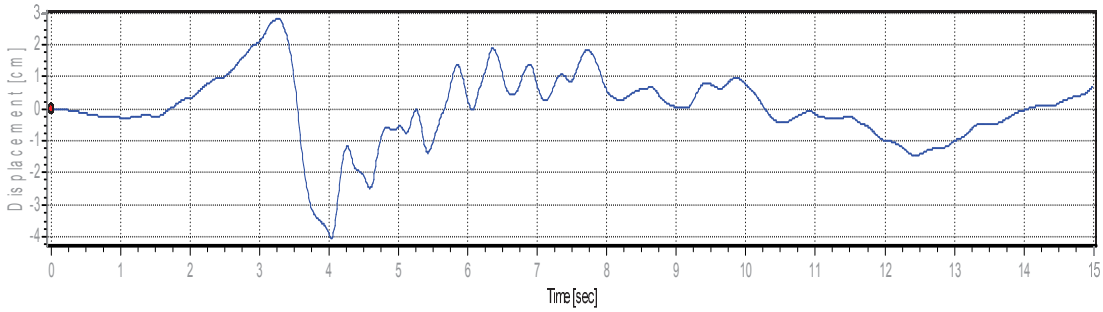


Figure 5.42. Signal de réponse et signal cible (Northridge) de la table avec spécimen à 1DDL sans correction baseline du signal d'excitation.

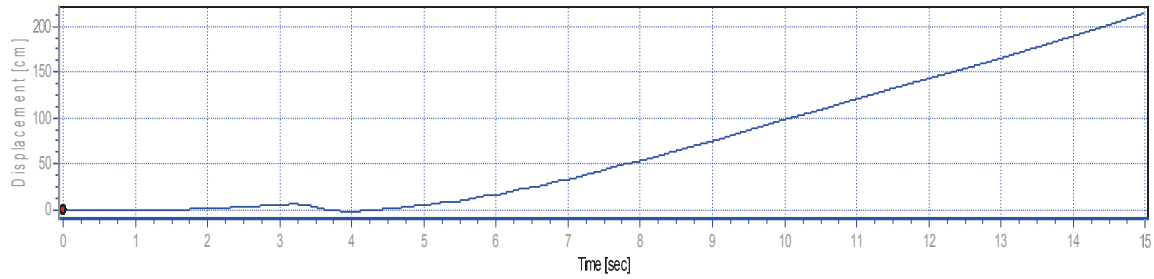


(a) (b)

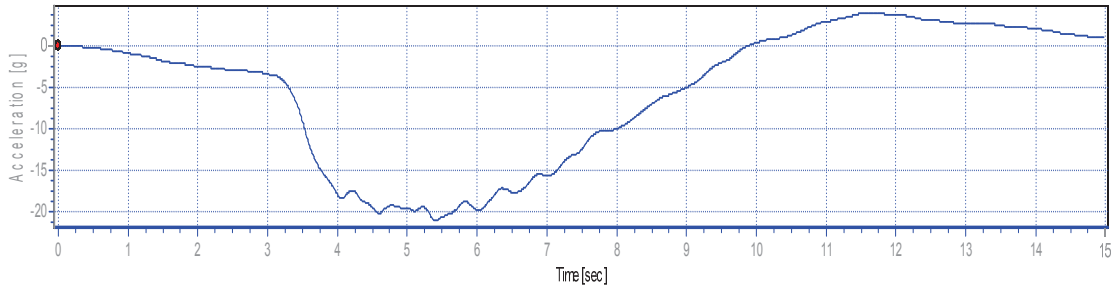
Figure 5.43 : signal de Friuli (a) sans correction baseline et (b) avec correction baseline.



(a)



(b)



(c)

Figure 5.44 : signal de Friuli en déplacement (a) input du RNA en déplacement sans correction baseline, (b) output du RNA avant correction baseline et (c) output du RNA après correction baseline

V.3.3. Essai d'application du système de contrôle par RNA:

Le principe de cette stratégie de contrôle est d'introduire le signal prédit par le RNA comme signal de commande de la table (input à la fonction de transfert) de telle sorte à ce que le signal de réponse de celle-ci soit égal au signal désiré. Le but étant également de montrer que la correction additionnelle par le RNA est plus efficace que celle par PID seulement (intégré dans la fonction de transfert).

Pour ce faire, un input sera présenté au réseau comportant l'accélération, la vitesse et le déplacement désiré. L'output du réseau sera le signal de commande introduit comme input à la fonction de transfert. Ensuite, le signal de réponse de la table en accélération sera comparé au signal désiré (accélérogramme réel).

Le test a été fait sur le modèle N°1. Le signal utilisé est l'accélérogramme d'El Centro.

La procédure de correction des signaux par RNA est illustrée dans la figure 5.32.

La comparaison entre la réponse de la table avec correction par PID, la réponse de la table avec correction par PID+RNA et le signal désiré (accélérogramme d'El Centro) est montrée dans la figure suivante :

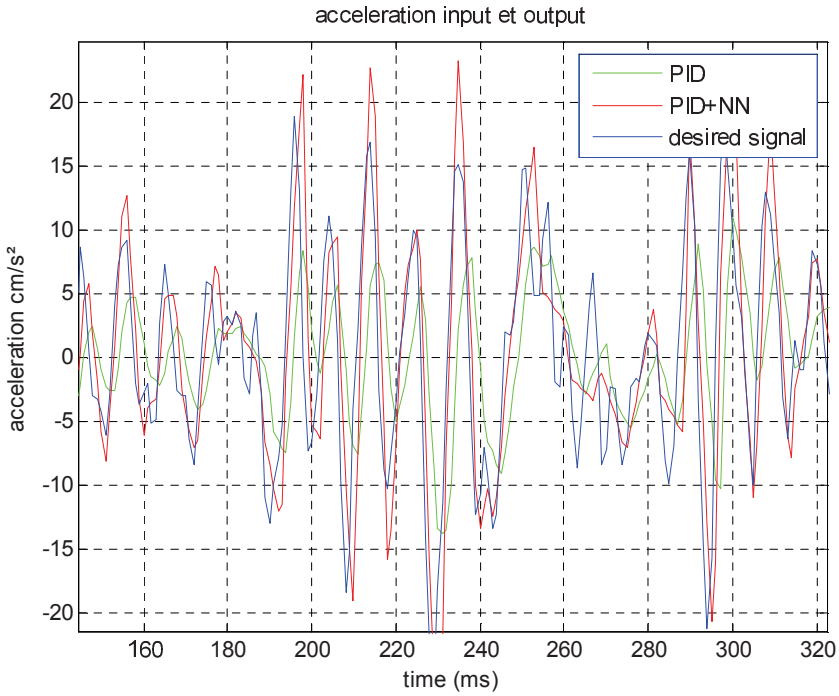


Figure 5.45. Signal de réponse et signal cible (El Centro) de la table+spécimen.

V.4. Conclusion

Le réseau de neurones artificiels choisi comme contrôleur offline de la table vibrante s'avère un moyen efficace dans la réduction de la distorsion de signal simulé sur la plateforme.

Dans le cas où la table est munie d'aucun contrôleur online, le contrôle par RNA prédit avec une très grande précision le signal de commande qui déplacera la plateforme de la table selon le signal désiré. Ceci étant valable pour une table non chargée, une table chargée avec un spécimen rigide ou flexible, linéaire ou non linéaire, avec des vérins hydrauliques non linéaires. Ce qui prouve que le RNA est un moyen de contrôle très efficace qui peut faire face même aux non linéarités du système et du spécimen.

Dans le cas où la table est munie d'un PID contrôleur, qui utilise un contrôle en déplacement, les résultats donnés par le RNA sont moins satisfaisants lors de grandes amplitudes de signal en déplacement. Ce comportement est attribué à une perte de précision en accélération pour de grands déplacements. A titre de démonstration, une correction additionnelle en baseline (élimination des grandes périodes du signal) améliore sensiblement la correction par RNA. Néanmoins, pour des signaux dont les déplacements sont modérés, le signal prédit par le RNA se rapproche encore plus du signal désiré que le signal corrigé par le PID seulement.

Chapitre VI : Conclusion générale

Les tests sur table vibrante suscitent de plus en plus d'intérêt du fait qu'ils sont seuls à pouvoir reproduire les forces d'inertie réelles développées lors d'un séisme, surtout quand la table vibrante arrive à reproduire les signaux désirés (accélérogrammes réels) avec une grande précision. Ceci reste alors le plus gros challenge pour les ingénieurs et chercheurs, car malgré le développement des stratégies et équipements de contrôle, les signaux sont toujours simulés avec un certain degré de distorsion.

Dans ce projet de recherche, une méthode de contrôle offline basée sur un réseau de neurones artificiels a été développée dans le but de prédire le signal de commande qui fera en sorte que le signal de réponse de la table (au niveau de la plateforme) qui excitera la base du spécimen soit égal au signal désiré.

En premier lieu, le système de table vibrante a été simulé par un modèle en éléments finis, représentant uniquement la plateforme, le spécimen et les vérins, sans aucun contrôleur. Ce modèle prend en compte la non-linéarité des vérins et du spécimen. Puis en second lieu, un modèle plus réaliste mais simplifié de la table et de ses équipements a été fait sous forme de fonctions de transfert sous Matlab. Il représente le bloc de réaction, la plateforme, le spécimen, le système servo-hydraulique et le contrôleur online de type PID. Ces deux modèles ont servi à la construction de la base de données d'apprentissage et de test du réseau de neurones.

Le réseau de neurones choisi est un réseau de type Elman constitué de deux couches cachées de 21 neurones chacune. L'algorithme d'apprentissage utilisé est l'algorithme de Lavenberg-Marquardt. Il reçoit comme input le signal recueilli au CDG de la plateforme ou la réponse de la fonction de transfert de la table, en terme d'accélération, de vitesse et de déplacement (input à 3 vecteurs) et l'output du réseau sera comparé au signal cible qui est le signal appliqué à la partie fixe du vérin ou le signal de commande envoyé à la fonction de transfert, en terme d'accélération (output à un vecteur).

Dans le cas d'une table munie d'aucun contrôleur online (modèle EF), le réseau de neurones s'est avéré particulièrement efficace en prédisant un signal de commande qui une fois envoyé à la table, celle-ci fournit un signal de réponse similaire au signal désiré avec une très grande précision. L'estimation de la performance du réseau a été évaluée en calculant le coefficient de corrélation R entre l'output du réseau et le signal cible. Dans le cas d'une table non chargée, chargée d'un spécimen non linéaire rigide ou flexible, et avec des vérins hydrauliques non linéaires, le coefficient de corrélation obtenu tend vers 1, pour plusieurs accélérogrammes présentés. Ceci indique la très grande fiabilité du RNA dans la prédiction des signaux de commande, même en présence de non-linéarités.

Dans le cas d'une table munie d'un PID contrôleur (modèle sous Matlab), le rôle du RNA est alors d'améliorer encore la correction des signaux obtenue par le PID. Le réseau de neurones s'est avéré satisfaisant, mais pas autant que dans le cas d'une table sans contrôleur. Pour plusieurs accélérogrammes, la plus grande valeur du coefficient de corrélation obtenue entre la réponse du RNA et le signal cible est de 0.96. Ceci est dû au fonctionnement du PID qui compense l'erreur entre le signal de réponse de la table et le signal cible de manière itérative, très dépendante du signal en question. Ce qui a été un obstacle pour le RNA pour établir une relation clair input/output puisque celle-ci changeait à chaque signal. Néanmoins, le réseau a été satisfaisant en réduisant les délais et les distorsions d'amplitudes. Par exemple pour plusieurs accélérogrammes comme celui d'ElCentro ou de

Kobe, le signal obtenu par la correction additionnelle du RNA s'est avéré plus proche du signal désiré que celui obtenu par la correction du PID uniquement. D'autre part, pour des signaux ayant des grands déplacements, le contrôle jumelé PID+RNA perd de précision en accélération, car le PID contrôle uniquement les déplacements. Quand on réduit les déplacements par une correction en baseline, le contrôle PID+RNA s'améliore nettement.

De manière générale, le contrôleur offline par RNA a montré un très grand potentiel dans la prédiction de signaux de commande aidant à réduire la distorsion entre le signal simulé sur la plateforme de la table et le signal désiré.

L'amélioration de la performance des tables vibrantes reste un but pour lequel nombreux chercheurs travaillent aujourd'hui. Ce projet de recherche ouvre alors nombreuses perspectives :

- améliorer la modélisation de la table et de ses équipements ;

- incorporer la modélisation du réel contrôleur online de la table (le TVC) ;

- étudier l'influence de d'autres types de spécimen ;

- tester l'efficacité de ce contrôleur offline par des tests expérimentaux.

REFERENCES BIBLIOGRAPHIQUES

[1] Abedihayati S. and Auslander D., (1977). Control of a single-actuator shaking table. Journal of American Society of Mechanical Engineers.

[2] Adeli H. and Yeh. C., (1989). Perceptron learning in engineering design. Journal of computer-aided civil and infrastructure engineering. **4**, 247-256.

[3] Airouche A.H., Bechtoula H., Aknouche H., Thoen B.K. and Benouar D., (2014). Experimental identification of the six DOF CGS, Algeria, shaking table system. Smart Structures and Systems. **13(1)**, 137-154.

[4] Airouche A.H., Casarotti C., Thoen B.K., Dacarro F. and Pavese A. (2008). Numerical modeling and experimental identification of the EUCENTRE TREES LAB shake table. The 14th World Conference on Earthquake Engineering, 2008, China.

[5] Ammanagi S., Poornima V., Sera A. and Sunder R., (2008). Development of a digitally-controlled three-axis earthquake shake table. Journal of current science. **91(2)**.

[6] Aristizabal-Ochoa J.D et Clark A.J (1980). Large-scale earthquake simulation tables. The 7th world conference on earthquake engineering, Istanbul, Turkey. **7**, 157-164.

[7] Badaoui M., Chateauneuf A., Fournely E., Bourahla N. and Bensaïbi M.(2012). Evaluation of accidental eccentricity for buildings by artificial neural networks. Structural Engineering and Mechanics, **41(4)**.

[8] Bauchau O.A. and Liu H., (2006). On the Modeling of Hydraulic Components in Rotorcraft Systems. Journal of the American Helicopter Society. **51(2)**, 175-184.

[9] Bekkouche, A., Derras, B., & Zendagui, D. (2005) Quantification et estimation du risque lié à l'effet de site par la méthode neuronale. 23ème rencontres universitaires de génie civil AUGC, Grenoble, France.

[10] Bernard Widrow B. and Walach E., 1995. Adaptive Inverse Control. Prentice-Hall, Englewood Cliffs, NJ, 1995.

[11] Blondet M. and Esparza C. (1988). Analysis of shaking table-structure interaction effects during seismic simulation tests. Journal of Earthquake engineering and structural dynamics. **16(4)**, 473-490.

[12] Blum A., (1992). Neural Networks in C++, Wiley.

[13] Bouwkamp J.G., Clough R.W., Penzien J. and Rea D. (1969). Design and research potential of two earthquake simulator facilities. The 4th World Conference on Earthquake Engineering, Santiago, Chile.

[14] Carydis P.G., Mouzakis H.P., Vougioukas E.A., Crewe A.J., Taylor C.A., Franchioni G. and Bairrao R. (1996). Comparative assessment of shaking tables. The 11th world conference on earthquake engineering, Acapulco, Mexico. Paper No. **2044**.

[15] Chase J.G., Hudson N.H., Lin J., Elliot R. and Sim A., (2005). Nonlinear shake table identification and control for near-field earthquake testing. Journal of Earthquake Engineering. **9(4)**, 461–482

[16] Chen S., Billings S.A. and Grant P.M. (1990). Non-linear system identification using neural networks. International Journal of Control. **51**, 1191-1214.

[17] Clark A.J. et Burton G.W (1978). Design considerations for large scale shaking table systems. The 6th European Conference on earthquake engineering, Dubrovnik, Yugoslavia.

[18] Clark A.J., (1992). Dynamic characteristics of large multiple degrees of freedom shaking tables. Tenth world conference earthquake engineering, Madrid, Spain. 2823-2828.

[19] Conte J.P. and Trombetti T.L, (2000). Linear dynamic modeling of a uniaxial servo-hydraulic shaking table

system. Earthquake engineering and structural dynamics, **29**, 1375-1404.

[20] Conte J.P., Luco J.E. and Ozcelik O. (2010). Shake table control fidelity in signal reproduction and sources of signal distortion. Shake table training workshop, San Diego, California, USA.

[21] Crewe A.J. (1998). The characterisation and optimisation of earthquake shaking table performance. Depart. Of Civil Engineering, Bristol University, UK.

[22] Crewe A.J. and Severn R.T., (2001). The European collaborative program on evaluating the performance of shaking tables. Philosophical Transactions of the Royal Society London A., **359**, 1671-1696.

[23] Derbal I. et Bourahla N., (2013). Evaluation de l'excentricité accidentelle par les RNAs à partir d'une réponse dynamique. Conférence Nationale sur les Matériaux et Structures, L'Ecole Nationale Polytechnique. Algérie, Octobre 2013.

[24] Derras B. and Bekkouche A.M., (2010). Estimation de la vitesse maximale du sol par la méthode des réseaux de neurones artificiels. Journées Nationales de Géotechnique et de Géologie de l'Ingénieur -Grenoble 7-9 juillet 2010.

[25] Dyke S.J., Spencer Jr B.F., and al., (1995). Experimental verification of acceleration feedback control strategies for an active tendon system. Technical report of the National Center for Earthquake Engineering Research, Buffalo, New York. 94-0024.

[26] Elman J.L., (1990). Finding Structure in Time. Cognitive Science, **14**, 179-211.

[27] Fabri, S. and V. Kadirkamanathan., (1996). Dynamic structure neural networks for stable adaptive control of nonlinear systems. IEEE Transactions on Neural Networks. **7**, 1151-1167.

[28] Filiatrault A., Kremmidas S., Seible F., Clark A.J., Nowak R. AND Thoen B.K. (2000). Upgrade of first generation uniaxial seismic simulation system with second generation real-time three-variable digital control system. The 12th World Conference on Earthquake Engineering, Auckland, New Zealand, 2000.

[29] Filiatrault, Tremblay R., Thoen B.K. and Rood R. (1996). A second generation earthquake simulation system in Canada: description and performance. The 11th World Conference on Earthquake Engineering, Acapulco, Mexico. Paper No. **1204**.

[30] Ghaboussi J., and Lin C J., (1998). New method of generating spectrum compatible accelerograms using Neural networks. Earthquake Engineering & Structural Dynamics. **27**, 377-396.

[31] Ghaboussi J., J. H. Garrett and X., (1991). Knowledge-based Modeling of Material Behavior with Neural Networks", Journal of Engineering. Mechanical. **117(1)**, 132 -153.

[32] Gomez E.G. and Stoten D.P., (2000). A comparative study of the adaptive MCS control algorithm on European shaking tables. The 12th World Conference on Earthquake Engineering. Paper No. **2626**.

[33] Haykin S., (1998). Neural networks, a comprehensive foundation. University of McMaster, Canada.

[34] Hola J., Schabowicz K., (2005). New technique of non-destructive assessment of concrete strength using artificial intelligence. NDT&E International. 2005, **38(4)**, 251-259.

[35] Hopfield J.J., (1982). Neural networks and physical Sytems with emergent collective computational abilities. Proceeding Nat. Acad. Sci. USA, Biophysics. **79**, 2554-2558.

[36] Hornik K., Stinchcombe M. and White H., (1989). Multilayer feedforward networks are universal approximators. Journal of Neural Networks. **2**, 359-366.

[37] Hornik, K. (1991). Approximation capabilities of multilayer feedforward networks. Neural Networks. **4**, 251-257.

[38] Igarashi A., Iemura H. and Suwa T., (2000). Development of substructured shaking table test method. The

12th World Conference on Earthquake Engineering. Paper No. **1775**.

[39] Inel M., (2007). Modeling ultimate deformation capacity of RC columns using artificial neural networks. Engineering Structure. 29, 329-335.

[40] Kewalramani M.A., Gupta R., (2006). Concrete compressive strength prediction using ultrasonic pulse velocity through artificial neural networks. Automation in Construction. **15**, 374-379.

[41] Kouassi A.M., Koffi Y.B., Kouamé K.F, Lasm T. and Biemi J., (2013). Application d'un modèle conceptuel et d'un modèle de réseaux de neurones artificiels à la simulation des débits annuels dans le bassin versant du N'zi-Bandama (Côte d'Ivoire). Journal of Afrique Science. **09(1)**, 64-76.

[42] Kouki S., (2010). Apprentissage multi-objectifs de réseaux de neurones pour la prévision des crues. Thèse Master2, Université de Pierre et Marie Curie, France.

[43] Krawinkler H. (1979). Possibilities and limitations of scale-model testing in earthquake engineering. The 2nd US National Conference on Earthquake Engineering, Stanford University, USA. 283-292.

[44] Kuehn J., Epp D. and Patten W.N., (1999) high-fidelity control of a seismic shake table. Journal of Earthquake Engineering and Structural Dynamics. **28**, 1235-1254.

[45] Kusner D.A., Rood J.D. and Burton G.W., (1992). Signal reproduction fidelity of servo-hydraulic testing equipment. The 10th World Conference on Earthquake Engineering, Rotterdam, the Netherlands. 2683-2688.

[46] Kuzniar K., Maciag E., Obiala R. and Waszczyszyn C., (2000). Application of Neural Networks in natural periods identification and simulation of prefabricated buildings response. Soil Dynamics and Earthquake Engineering, **20**, 217-222.

[47] Lazar M. and Pastravanu O. (2002). A neural predictive controller for nonlinear systems.

[48] LeCun Y., Boser B., Denker J.S., Henderson D. and al., (1989). Back propagation applied to handwritten zip code recognization. Neural Computations. **1**, 541-551.

[49] Line Kong A Siou, (2011). Modélisation des crues de bassins karstiques par réseaux de neurones. Cas du bassin du Lez (France). Hydrology. Université de Montpellier II - Sciences et Techniques du Languedoc. French.

[50] Liu S.W., Huang J.H., Sung J.C., Lee C.C., (2002). Detection of cracks using neural networks and computational mechanics. Comput. Methods Appl. Mech. Engrg. **191**, 2831- 2845.

[51] Liu, G.P., V. Kadirkamanathan and S.A. Billings., (1998). Predictive control for non-linear systems using neural networks. International journal of Control. **71**, 1119-1132.

[52] Mahajerin Nima, (2012). Identification and predictive control using Recurrent Neural Networks.international master's thesis, university of Orebro, Sweden.

[53] Mansour M.Y., Dicleli M., Lee J.Y., Zhang J., (2004). Predicting the shear strength of reinforced concrete beams using artificial neural networks. Engineering Structure. **26**, 781-799.

[54] Minowa C., Ogawa N., Mikoshiba T., Ohtani K. and Oyagi N., (1991). Vibration of large scale shaking table foundation. The 23rd Joint Meeting of the US-Japan Cooperative Program in Natural Resources Panel on Wind and Seismic Effects, Tsukuba, Japan.

[55] Nakata N., (2010). Acceleration Trajectory Tracking Control for Earthquake Simulators. Engineering Structures. **32(8)**, 2229‑2236.

[56] Narenda K.S., (1990). Identification and Control of Dynamical Systems Using Neural Networks. IEEE transactions on Neural Networks. **1(1)**.

[57] Newell D.P., Hongliang D., Sain M.K., Quast P. and Spencer Jr B.F., (1995). Nonlinear modeling and control of a hydraulic seismic simulator. Proceeding of the American Control Conference, **1**, 801-805.

[58] Noura H., Abed M. and Bouarfa H., (2011). Prédiction de la vulnérabilité des constructions par un système de réseaux de neurones. 8ème Colloque National AFPS, Ecole des Ponts ParisTech 9, 2011.

[59] Nowak R.F., Kusner D.A, Larson L.R. and Thoen B.K., (2000). Utilizing modern digital signal processing for improvement of large scale shaking table performance. The 12th World Conference on Earthquake Engineering. Paper No. **2035**.

[60] O'Hagan J.T and Ma, (2012). Experimental assessment of PID control for a uniaxial shake table. The 15th World Conference on Earthquake Engineering, 2012, Lisboa.

[61] Ohtani K., Takahashi H, Fujiwara K. (1984). Research and development of three-dimensional shaking table. The 8th World Conference on Earthquake Engineering, San Francisco, USA. **6**, 39-46.

[62] Ozcelik O, (2008). A Mechanics-based Virtual Model of NEES-UCSD Shake Table: Theoretical Development and Experimental Validation. University of San Diego, California.

[63] Ozcelik O., Conte J.P. and Luco J.E. (2008). Virtual model of the UCSD-NEES High Performance Outdoor shake table. The 4th world conference on structural control and monitoring, San Diego, USA.

[64] Ozcelik O., Luco J.E. and Conte J.P., (2008). Identification of the mechanical subsystem of the NEES-UCSD shake table by a least-squares approach. *Journal of Engineering Mechanics.* **134(1)**.

[65] ÖZTAŞ A., Pala M., Özbay E., Kanca E., CAĞLAR N., Bhatti M.A., (2006). Predicting the compressive strength and slump of high strength concrete using neural network. Construction and Building Materials. **20**, 769-775.

[66] Pagano S., Russo R., Strano S. and Terzo M., (2013). Non-linear modelling and optimal control of a hydraulically actuated seismic isolator test rig. Mechanical Systems and Signal Processing. **35**, 255-278.

[67] Paolucci, R., Colli, P., & Giacinto G. (2000). Assessment of seismic site effect in 2-D alluvial Valleys Using Neural Networks. Earthquake Spectra, **16(3)**.

[68] Pham D.T. and Liu X., (1993). Identification of linear and nonlinear dynamic systems using recurrent neural networks. Artificial Intelligence in Engineering. **8(1)**, 67-75.

[69] Pham D.T. and Liu X., (1996). Training of Elman networks and dynamic system modelling. International Journal of of Systems Science. **27(2)**, 221-226.

[70] Phillips B.M. and Spencer B.F Jr., (2011). Feedforward-feedback tracking control for real-time hybrid simulation. The 6th International Workshop on Advanced Smart Materials and Smart Structures Technology 2011, Dalian, China.

[71] Rafiq M.Y., Bugmann G. and Easterbrook D.J., (2001). Neural Network design for engineering applications. Computers and Structures. **79(17)**, 1541-1552.

[72] Rea D. and Penzien J. (1974). Dynamic response of a 20ft x 20ft shaking table. The 5th World Conference on Earthquake Engineering, Rome. 1447-1456.

[73] Reinhorn A.M. and Prawel S.P (1986). Ferrocement in a large shaking table. Journal of Structural Engineering. **112(2)**, 401-416.

[74] Reinhorn A.M. et Prawel S.P (1983). Design of a composite shaking table for earthquake simulation. The 8th world conference on Electronic computation, Houston, Texas, USA. 725-736.

[75] Rohwer R. and Forrest B., (1987). Training Time Dependence in Neural Networks. Proc. IEEE ICNN, San Diego, II-701.

[76] Roodsari T., Mehrzad and Habibi, Reza M., (2008). Using neural network for prediction of the dynamic period and amplification factor of soil for microzonation in urban area. The 14th World Conference of Earthquake Engineering, Beijing, China.

[77] Rumelhart, D.E., Hinton, G. E., Williams, R. J., (1986). Learning Internal Representations by error backpropagation. Parallel Distributed Processing : Explorations in the Microstructure of Cognition. 318-362.

[78] Sakla S.S.S., Ashour A.F., (2005). Prediction of tensile capacity of single adhesive anchors using neural networks. Computers and Structure. **83**, 1792-1803.

[79] Seung C.L., & Sang W.H., (2002) Neural-network-based models for generating artificial earthquakes and response spectra. Computers and Structures. **80**, 1627–1638.

[80] Shaw M.R., Millard S.G., Molyneaux T.C.K., Taylor M.J., Bungrey J.H., (2005). Location of steel reinforcement in concrete using ground penetrating radar and neural networks. NDT&E International. **38(3)**, 203-212.

[81] Shen G., Zhu Z.C. and Han J.W., (2013) Combined control strategy using internal model control and adaptive inverse control for electro-hydraulic shaking table. The Institution of Mechanical Engineers, PartC-Journal of Mechanical Engineering Science.

[82] Simova M. and Mamuçevski D. (1980). Online control of single component of shaking table. The 7th world conference on earthquake engineering, Istanbul, Turkey. **2**, 63-69.

[83] Sivaselvan M.V., Reinhorn A.M., Shao X. and Weinreber S., (2008). Dynamic force control with hydraulic actuators using added compliance and displacement compensation. Journal of Earthquake Engineering and structural dynamics, **00**, 1-10.

[84] Siyahi B., Akbas B. and Onder N.D., (2008). Evaluation of liquefaction-induced lateral spreading by a neural network (nn) model. The 14th World Conference on Earthquake Engineering, 2008.

[85] Spencer Jr., B. F., Yang, G. (1998). Earthquake simulator control by transfer function iteration. Proceedings of the 12th ASCE Engineering Mechanics Conference.

[86] Stehman, M. and Nakata, N., (2012). Direct acceleration feedback control of shake tables with force stabilization. Journal of Earthquake Engineering 2012, **17(5)**, 736-749.

[87] Stoten D.P. and Gomez E.G., (2001). Adaptive control of shaking tables using the minimal controller synthesis algorithm. Philosophical transactions of the Royal Society.A, **359(1786)**, 1697-1723.

[88] Svozil D., Kvasnicka V. and Pospichal J., (1997). Introduction to multi-layer feed-forward neural networks. Chemometrics and Intelligent Laboratory System. **39**, 43-62.

[89] Takahashi Y., Rea D. and Abedi-Hayati S. (1974). Effects of test specimen reaction loads on shaking tables. The 5th world conference on earthquake engineering, Rome. **2**, 1426-1429.

[90] Thoen B.K. and Laplace P.N., (2004). Offline tuning of shaking tables. The 13th World Conference on Earthquake Engineering, Vancouver B.C., Canada. Paper No.960.

[91] Tienfuan K., & Ting S.B., (2005). Neural network estimation of ground peak acceleration at stations along Taiwan high-speed rail system. Engineering Applications of Artificial Intelligence. **18**, 857–866.

[92] Tricaud C. and Chen Y.Q. (2008). Linear and Nonlinear Model Predictive Control Using A General Purpose Optimal Control Problem Solver RIOTS 95. Control and Decision Conference, 2008, Chinese, 1552-1557.

[93] Trombetti T.L. and Conte J.P., (2002). Shaking table dynamics: results from a test-analysis and comparison study. Journal of earthquake engineering, **6(4)**, 513-551.

[94] Tsuchimoto K., Wada N and Kitagawa Y., (2004). Structural Assessment System for Damage and Degradation: Two-Stage Damage Identification Based on Neural Networks and Improved MDLAC Method.

Conference of Pressure Vessels and Piping, Seismic Engineering, San Diego, California. **2**, 115-120.

[95] Vazquez J., Zug M., Bellefleur D., Grandjean B. and Scrivener O., (1999). Utilisation d'un réseau de neurones pour appliquer le modèle de Muskingum aux réseaux d'assainissement. Revue des sciences de l'eau, **12(3)**, 577-595.

[96] Ventura C.E., Latendresse V., Nichol H.R., Tauscher R.C. and Stroud R.C. (1996). Multi-axis control system for shake-table. The 11th world conference on earthquake engineering, Acapulco, Mexico. Paper No. **1207**.

[97] Viriyametanont K., (2008). Reconnaissance physique et géométrique d'éléments en béton armé par Radar et réseaux de neurones artificiels. INSA, Toulouse, France.

[98] Widrow B. and Plett G.L, (1996). Adaptive Inverse Control based on Linear and Nonlinear Adaptive Filtering. International Workshop on Neural Networks for Identification, Control, Robotics and Signal/Image Processing.

[99] Williams R. and Zipser D., (1989-a). A Learning Algorithm for Continually Running Fully Recurrent Neural Networks. Neural Computation **1**, 270.

[100] Williams R. and Zipser D., (1989-b). Experimental Analysis of the Real-time Recurrent Learning Algorithm. Connection Science **1**, 87.

[101] Williams, D. M., Williams, M. S., and Blakeborough, A. (2001). Numerical modeling of a servohydraulic testing system for structures. Journal of Engineering Mechanics. **127(8)**, 816–827.

[102] Wu, X.., Ghaboussi, J. and Garrrett, J. H., (1992). Use of Neural Networks in Detection of Structural Damage. Computers and Structures. **42(4)**, 649-659.

[103] Xie X.Q., Zhou D.H. and Jin Y.H., (1998). Strong tracking filter based adaptive generic model control. Journal of Process Control. **9**, 337-350.

[104] Yang T.Y., Kang Li, Jian Yuan Lin and Yuanjie Li, (2013). Development of nonlinear control algorithms for shaking table tests. The ASME 2013 Dynamic Systems and Control Conference, Palo Alto, California, USA.

[105] Yeh I.C., (1998). Modeling of strength of high-performance concrete using artificial neural networks'. Cement and Concrete Research, 1998, **28(12)**, 1797-1808.

[106] Yong-Zai Lu, 1996. Industrial Intelligent Control: Fundamentals and Applications. British Library.

[107] Zaid M., Gaydecki P., Quek S., Miller G., Fernandes B., (2004). Extracting dimensional information from steel reinforcing bars in concrete using neural networks trained on data from an inductive sensor. NDT&E International. **37**, 551-558.

[108] Zapico J.L. and Gonzalez M.P., (2003). Damage assessment using neural networks. Mechanical Systems and Signal Processing. **17(1)**, 119-125.

[109] Zhao J., French C., Shield C. and Posbergh T., (2004). Effective force testing with nonlinear velocity feedback compensation. The 13th world conference on earthquake engineering, Vancouver, Canada. Paper No. **268**.

[110] Zhao J., Shield C., French C. and Posbergh T., (2005). Nonlinear System Modeling and Velocity Feedback Compensation for Effective Force Testing, Journal of Engineering Mechanics. **131(3)**, 244-253.

ANNEXE 1 : Les résultats de test de validation du RNA pour la prédiction du signal de commande.

1- Modèle en Eléments Finis de la table :

1.1-Modèle N°1 :

1-1-1. Accélérogramme de Dar El Beida : R=0.99

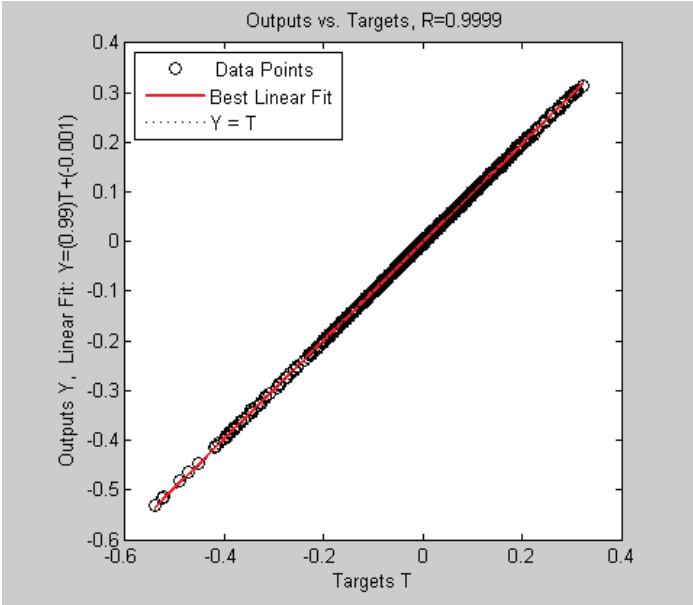


Figure 1.1.1. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°1.

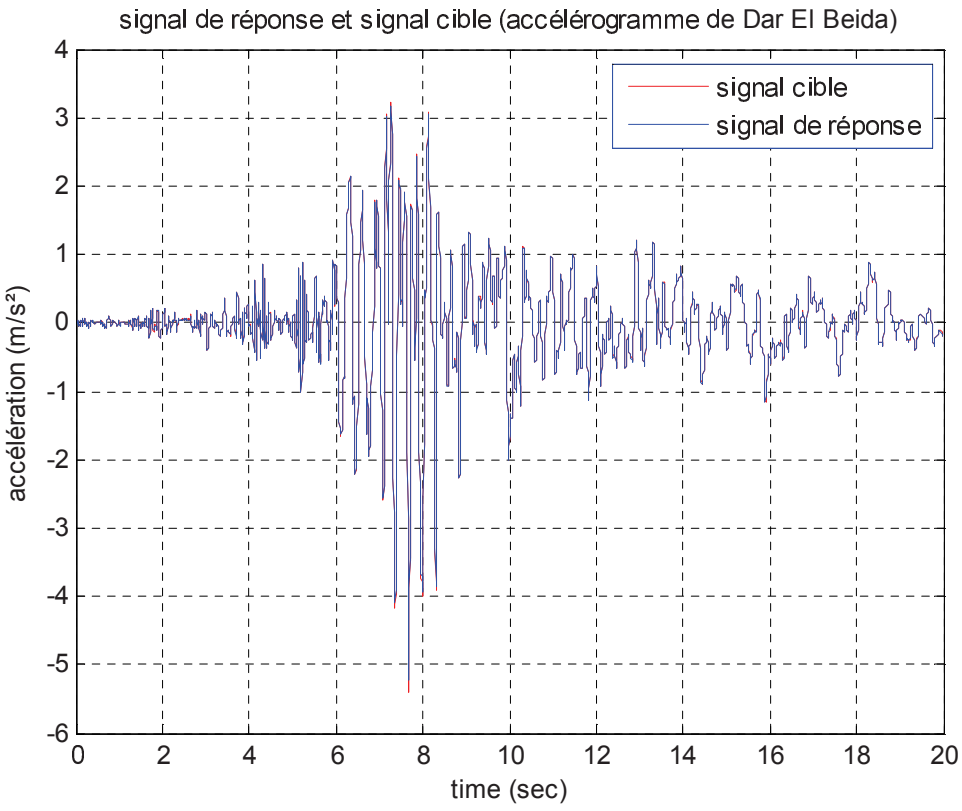


Figure 1.1.2. Différence entre le signal de réponse et le signal cible du modèle N°1.

1-1-2. Accélérogramme de Blida : R=0.99

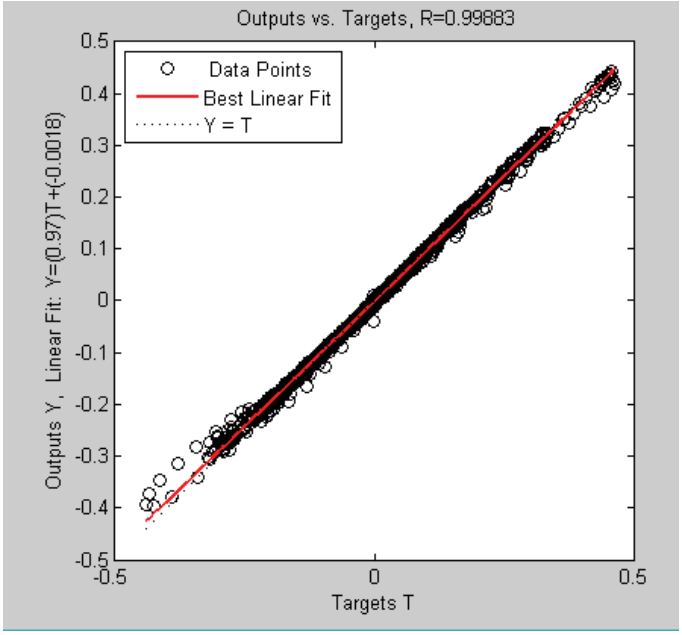


Figure 1.1.3. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°1.

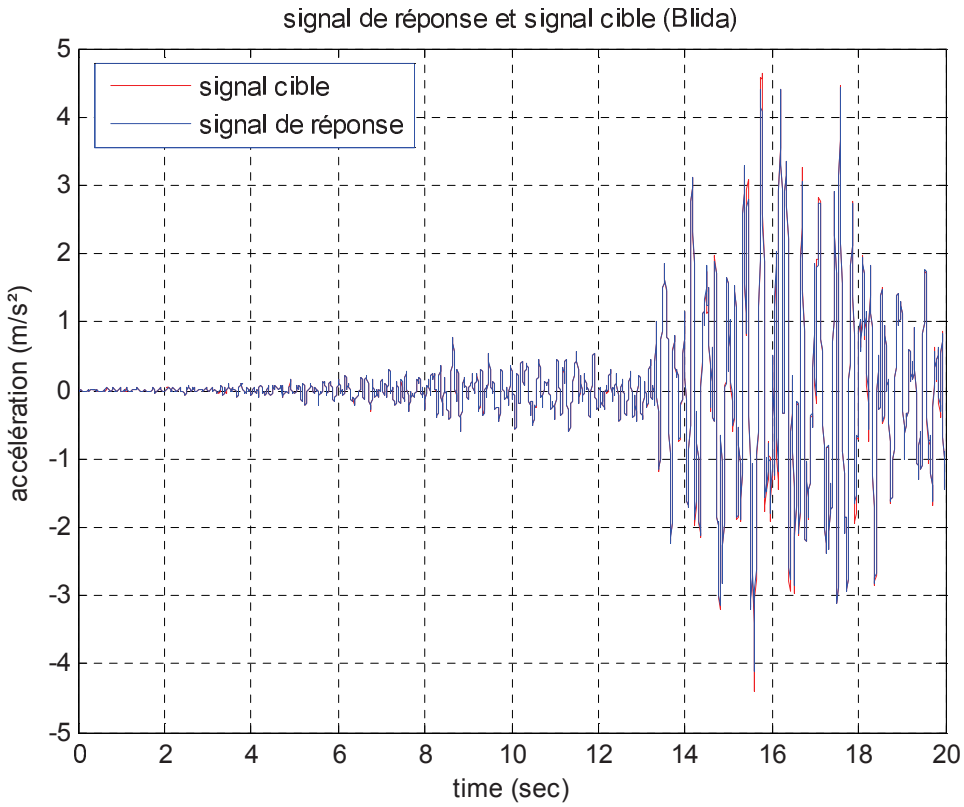


Figure 1.1.4. Différence entre le signal de réponse et le signal cible du modèle N°1.

1-1-3. Accélérogramme d'Hussein Dey : R=0.99

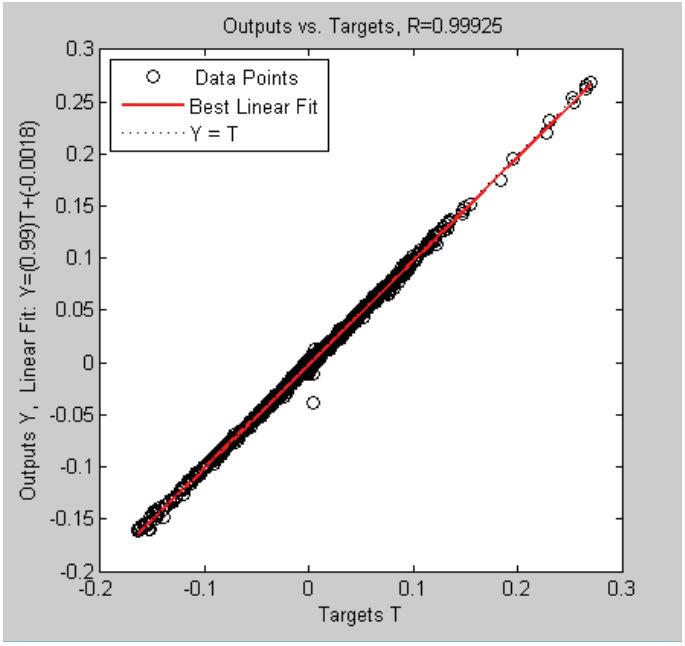


Figure 1.1.5. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°1.

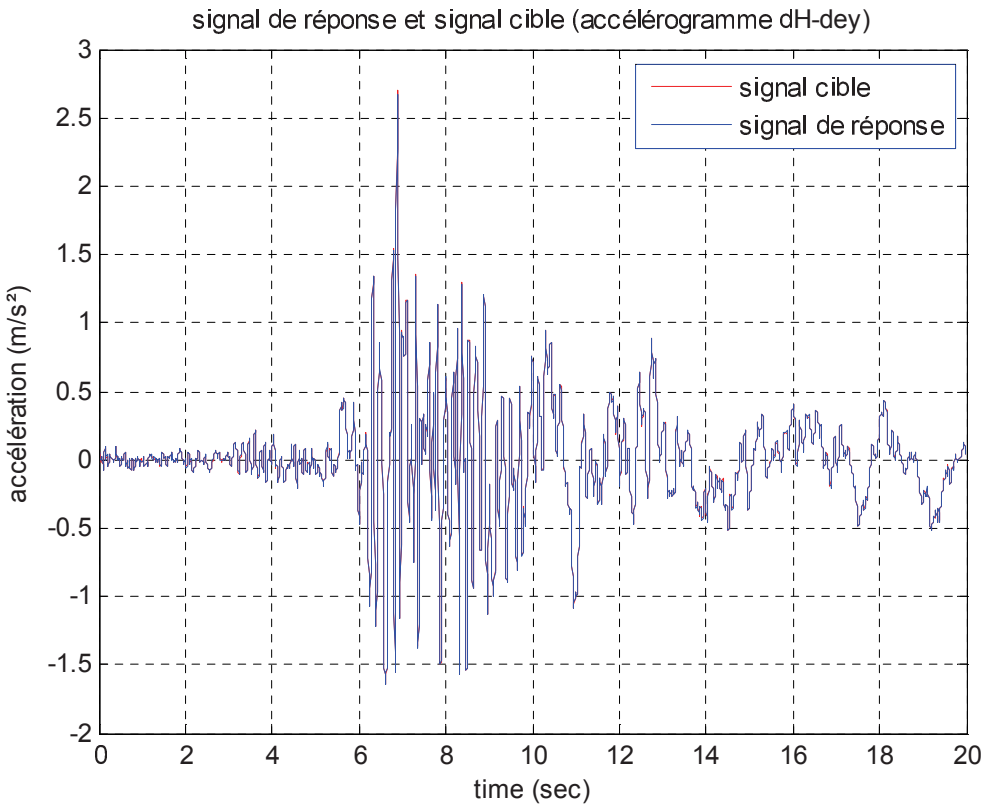


Figure 1.1.6. Différence entre le signal de réponse et le signal cible du modèle N°1.

1-1-4. Accélérogramme de Meliana : R=0.99

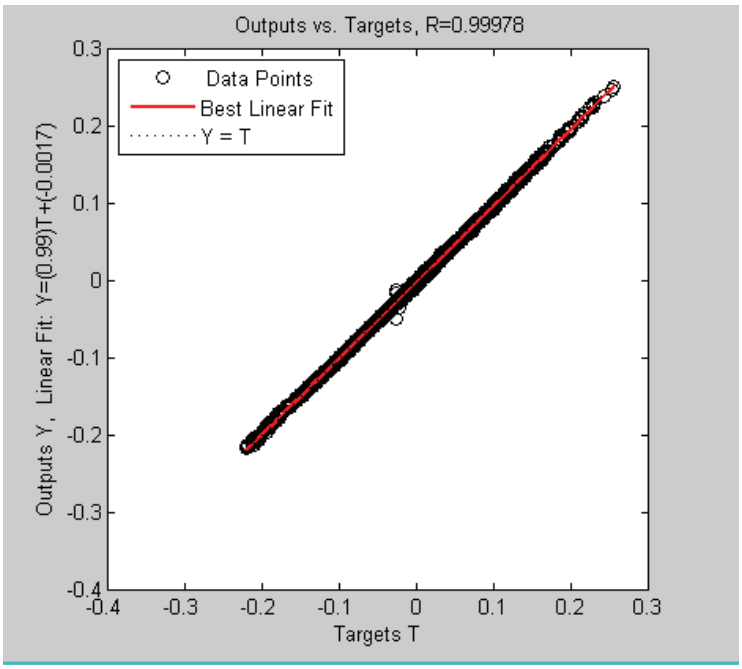


Figure 1.1.7. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°1.

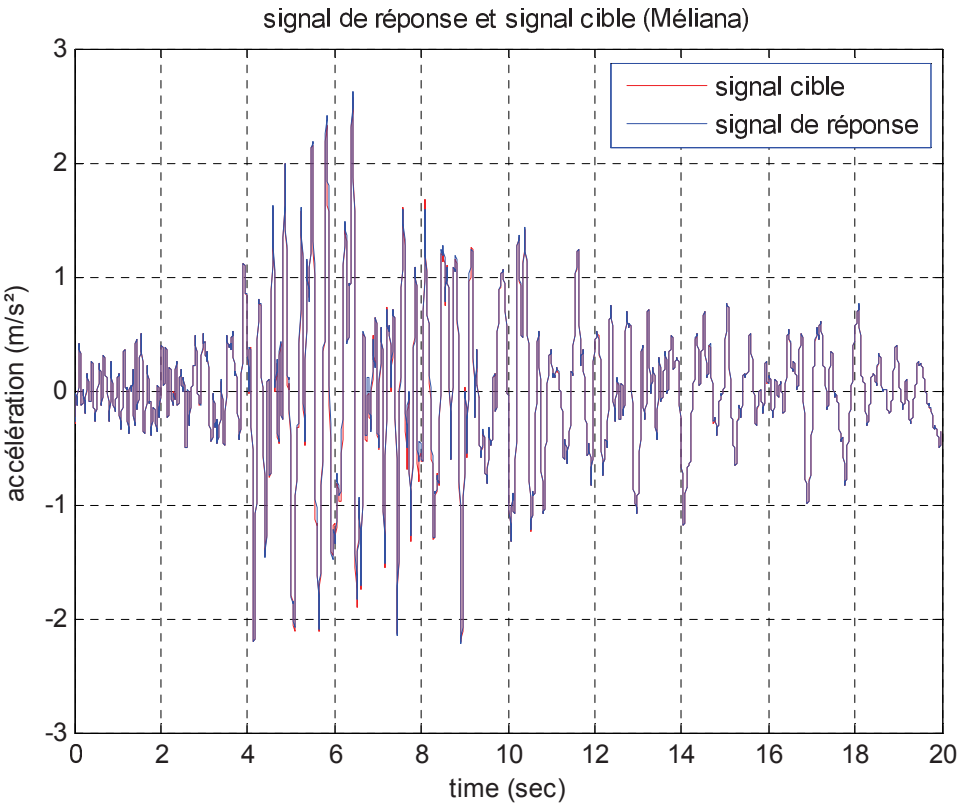


Figure 1.1.8. Différence entre le signal de réponse et le signal cible du modèle N°1.

1-1-5. Accélérogramme d'El Centro : R=0.99

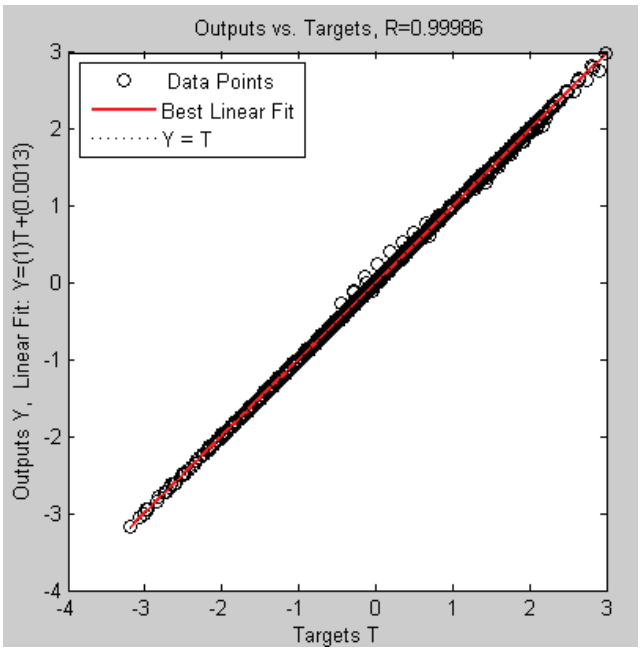


Figure 1.1.9. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°1.

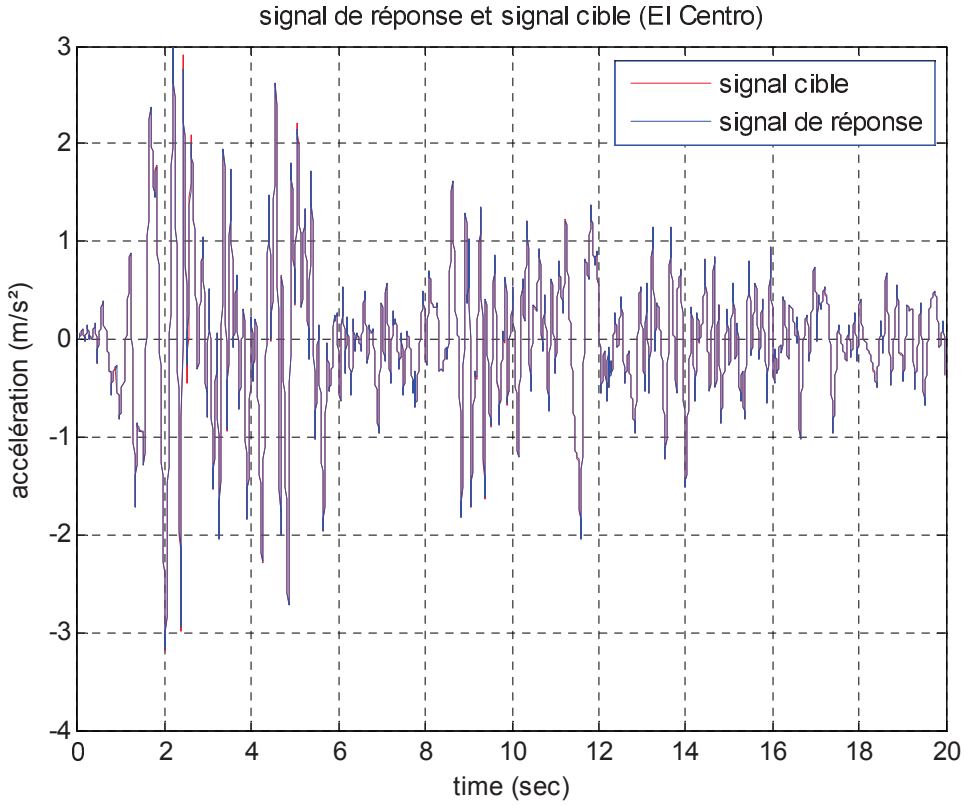


Figure 1.1.10. Différence entre le signal de réponse et le signal cible du modèle N°1.

1-1-6. Accélérogramme de Kobe : R=0.99

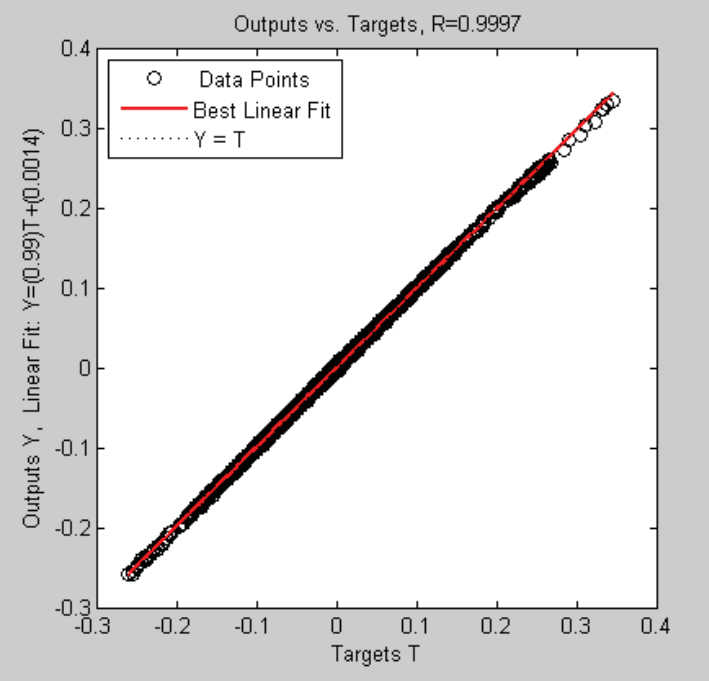


Figure 1.1.11. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°1.

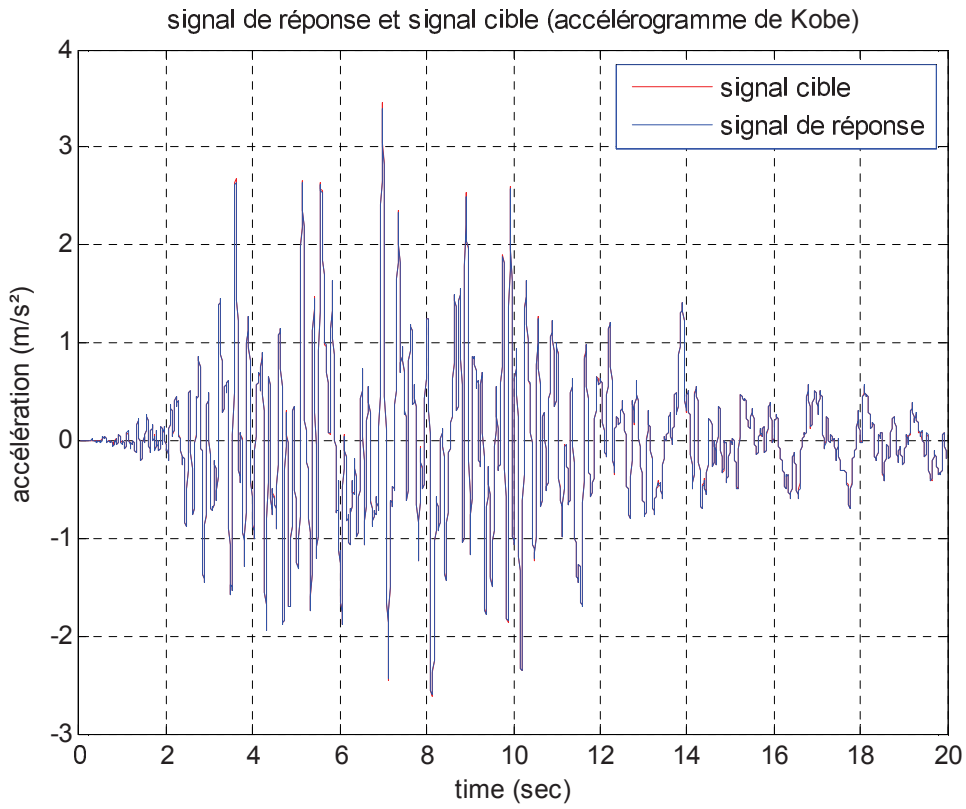


Figure 1.1.12. Différence entre le signal de réponse et le signal cible du modèle N°1.

1-1-7. Signal sinusoïdal : R=0.99

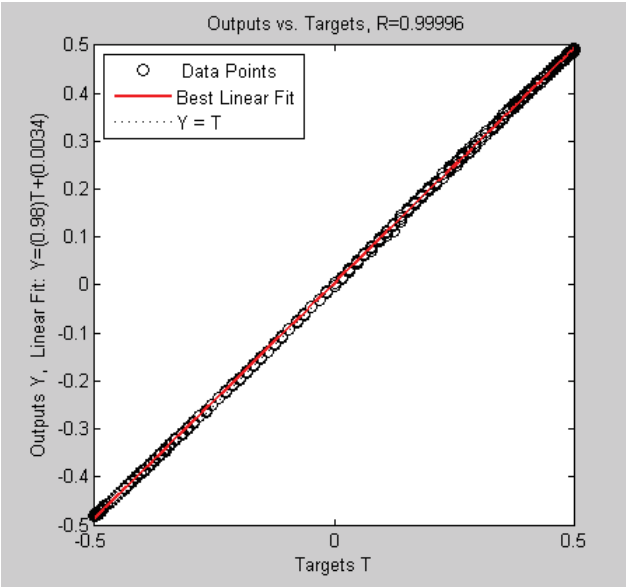


Figure 1.1.13. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°1.

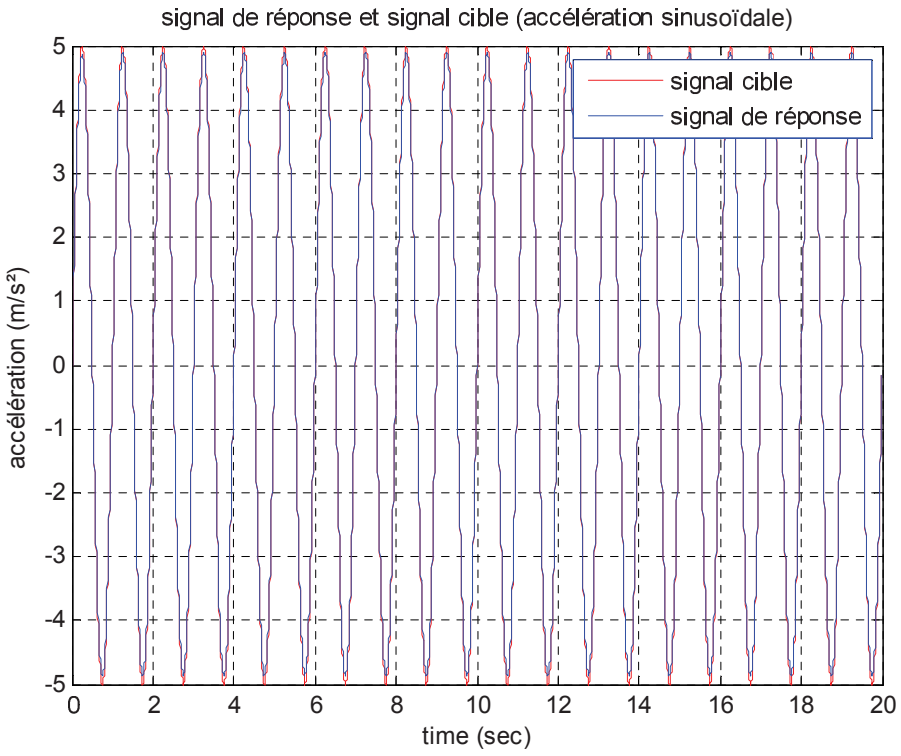


Figure 1.1.14. Différence entre le signal de réponse et le signal cible du modèle N°1.

1.2-Modèle N°2 :

1.2-1. Accélérogramme de Dar El Beida : R=0.99

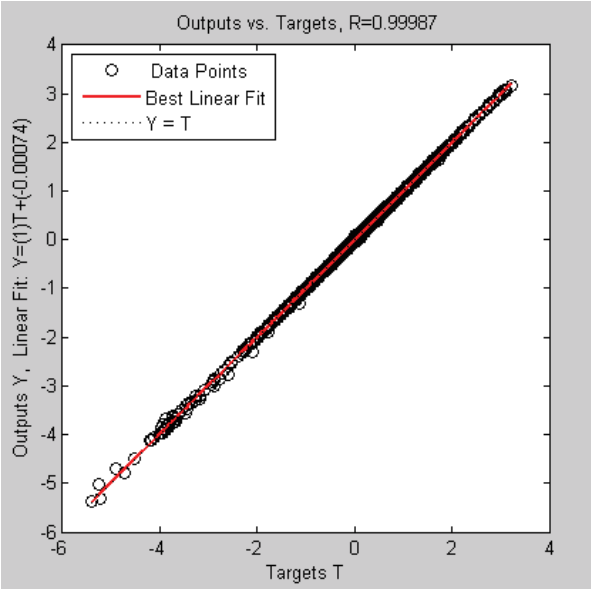


Figure 1.2.1. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°2.

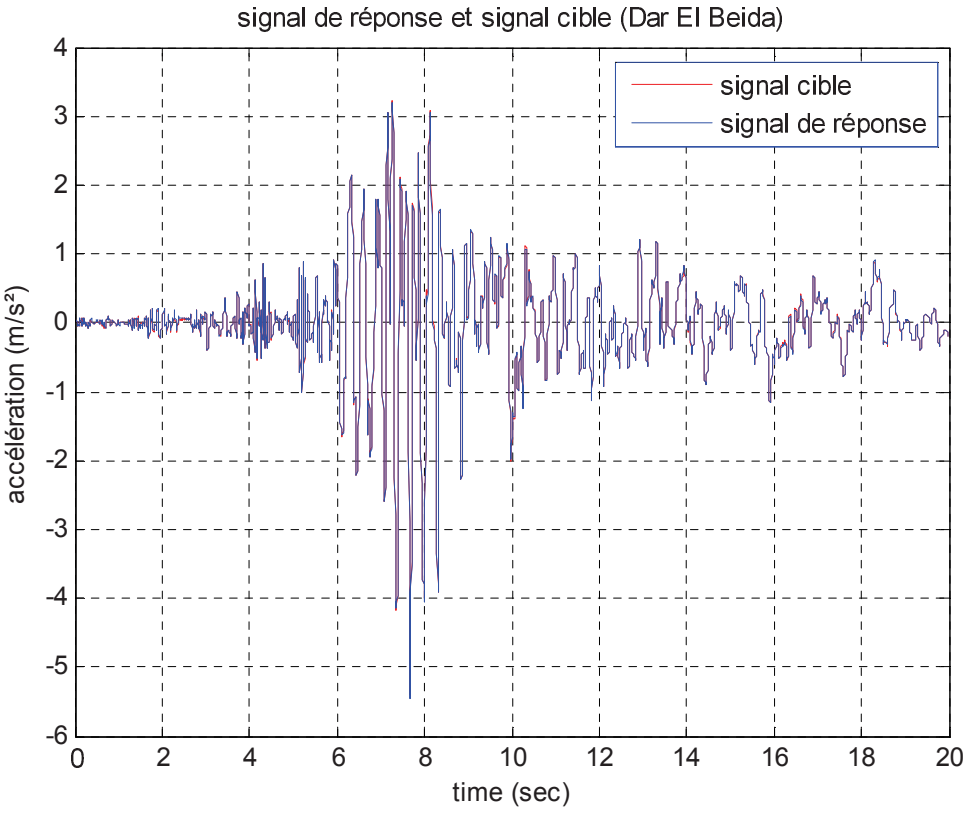


Figure 1.2.2. Différence entre le signal de réponse et le signal cible du modèle N°2.

1.2-2. Accélérogramme de Blida : R=0.99

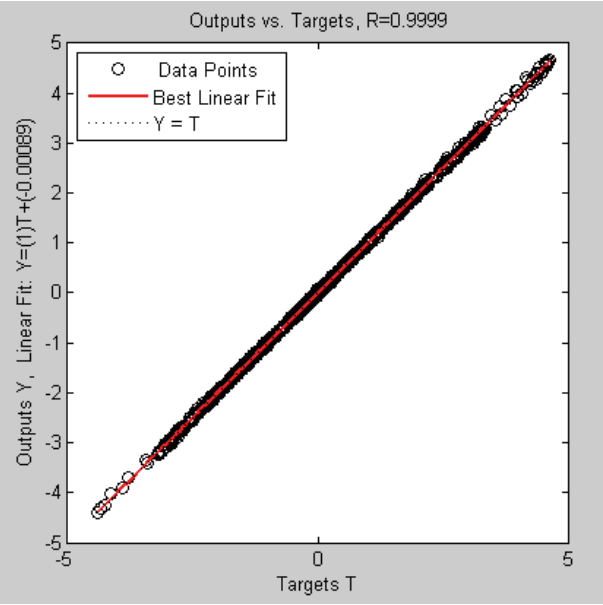


Figure 1.2.3. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°2.

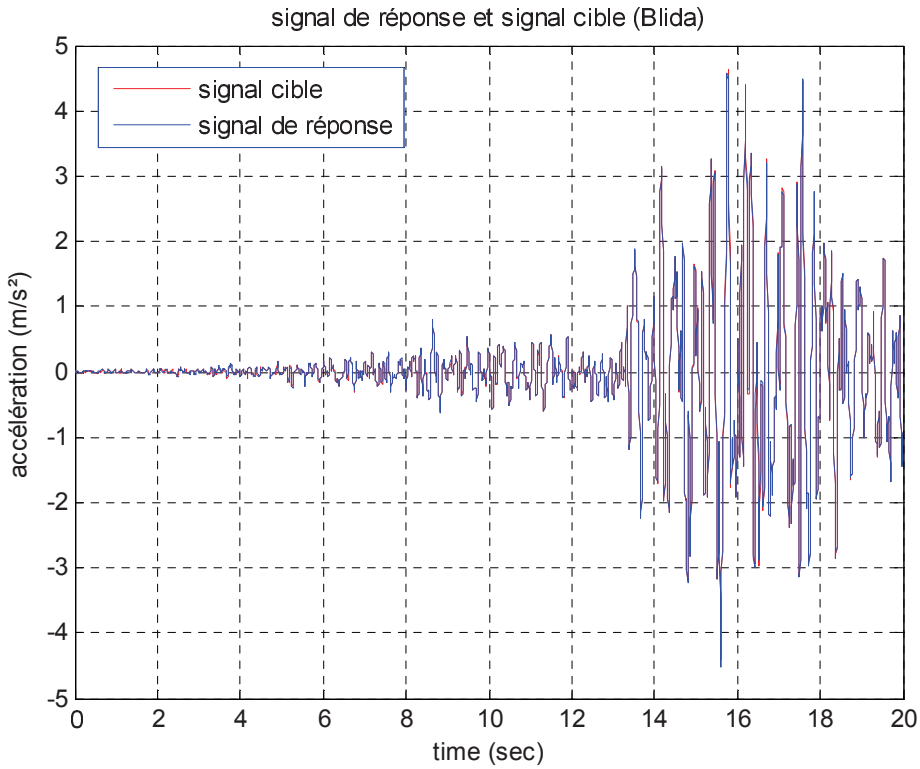


Figure 1.2.4. Différence entre le signal de réponse et le signal cible du modèle N°2.

1.2-3. Accélérogramme d'Hussein Dey : R=0.99

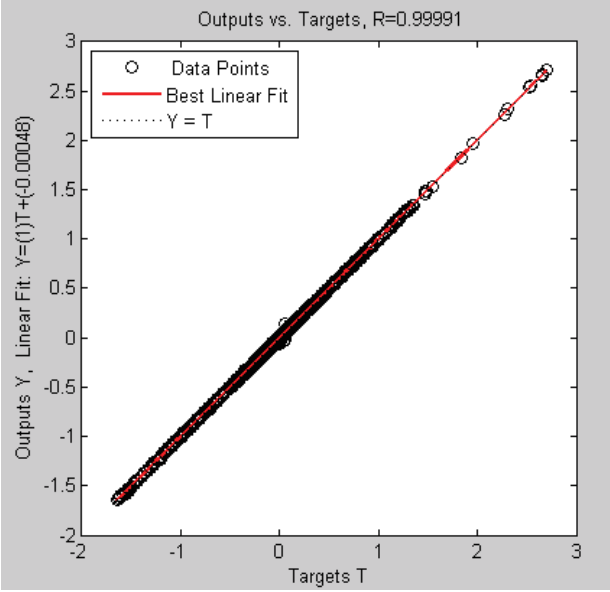


Figure 1.2.5. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°2.

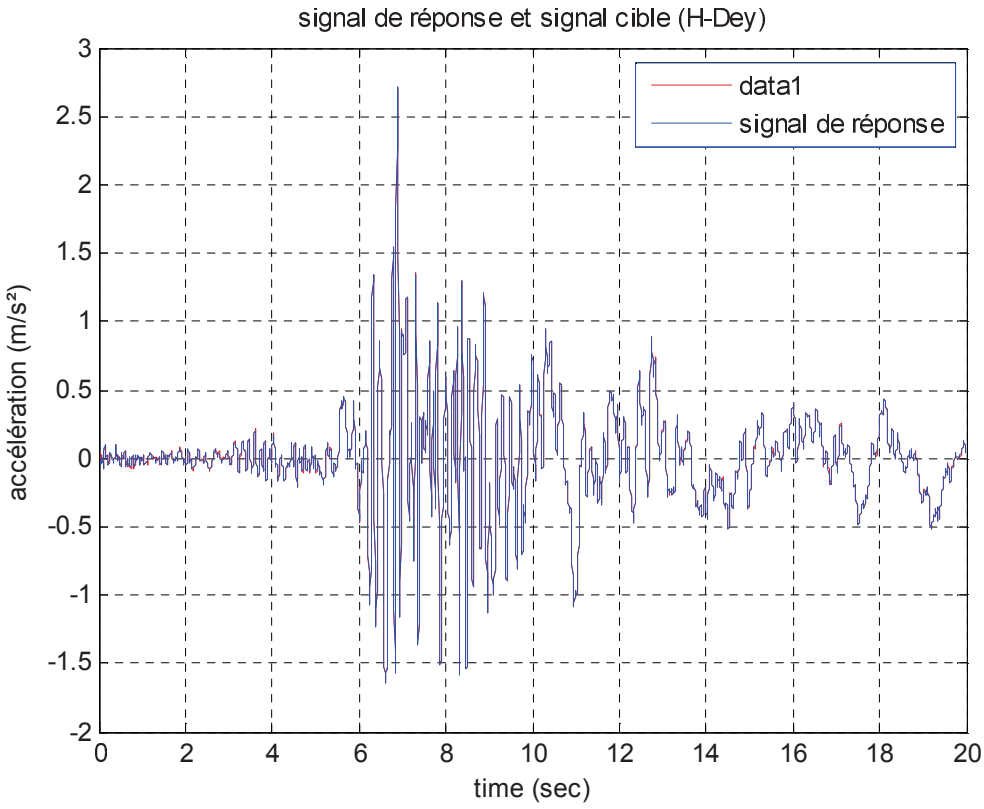


Figure 1.2.6. Différence entre le signal de réponse et le signal cible du modèle N°2.

1.2-4. Accélérogramme d'Azazga : R=0.99

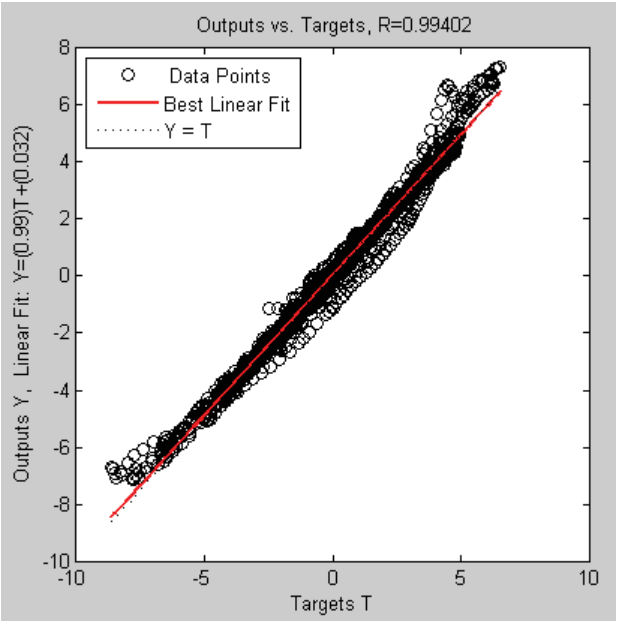


Figure 1.2.7. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°2.

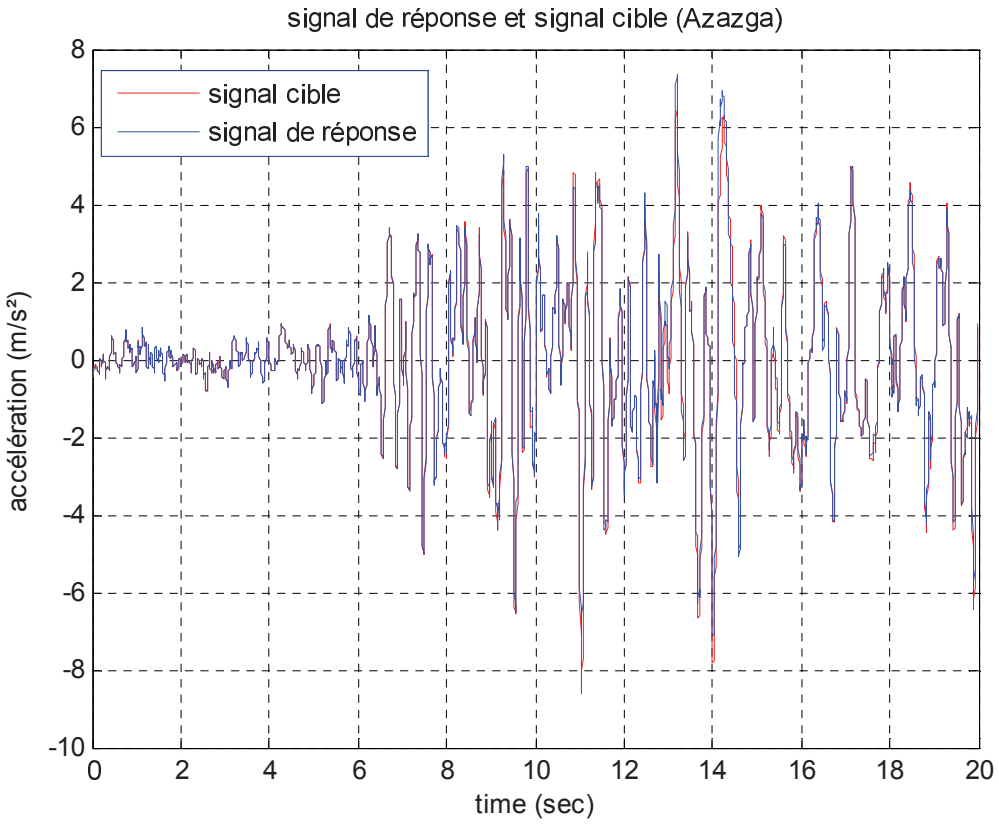


Figure 1.2.8. Différence entre le signal de réponse et le signal cible du modèle N°2.

1.2-5. Accélérogramme de Kedara : R=0.99

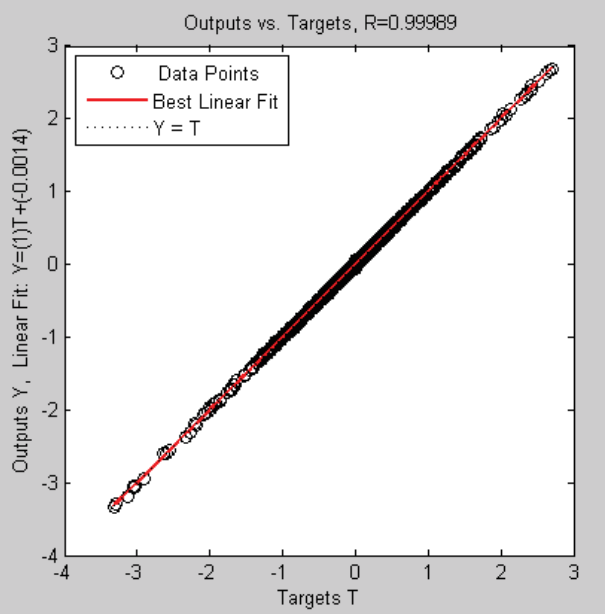


Figure 1.2.9. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°2.

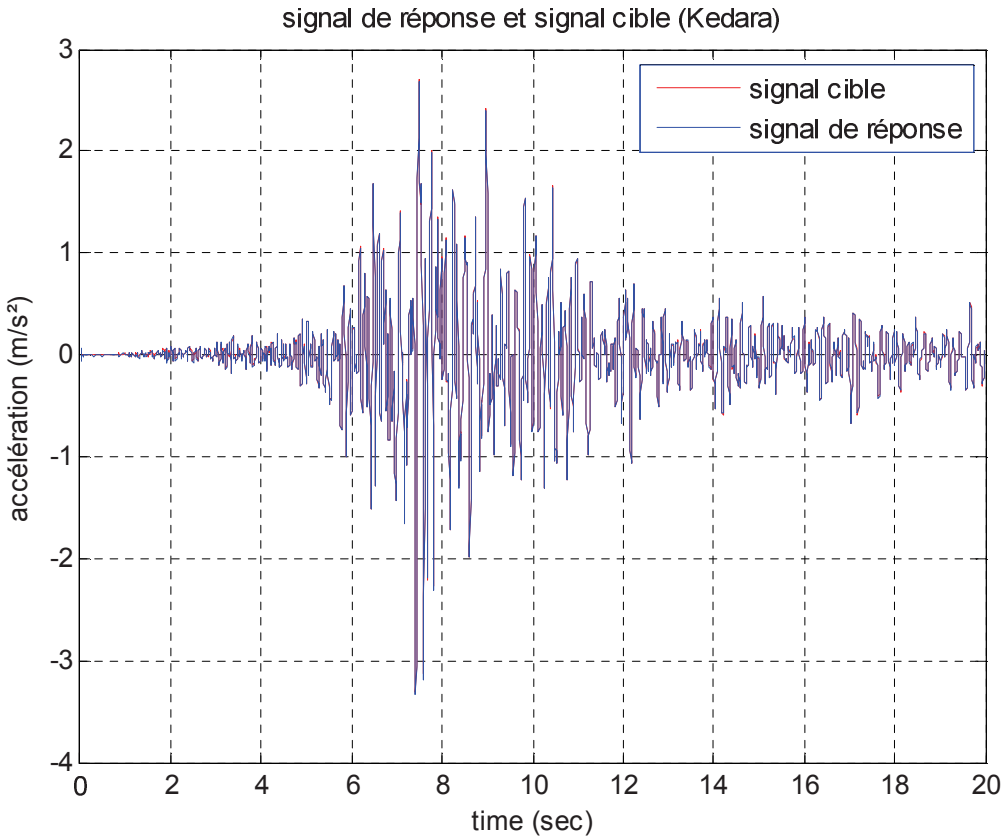


Figure 1.2.10. Différence entre le signal de réponse et le signal cible du modèle N°2.

1.2-6. Accélérogramme d'El centro : R=0.99

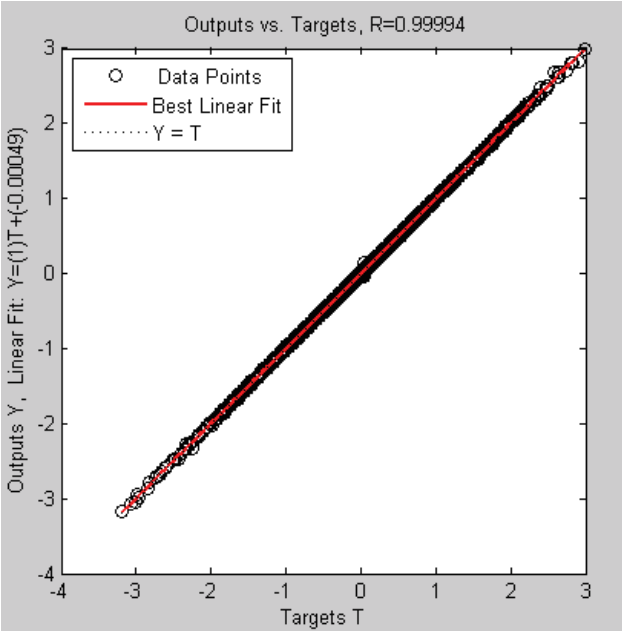


Figure 1.2.11. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°2.

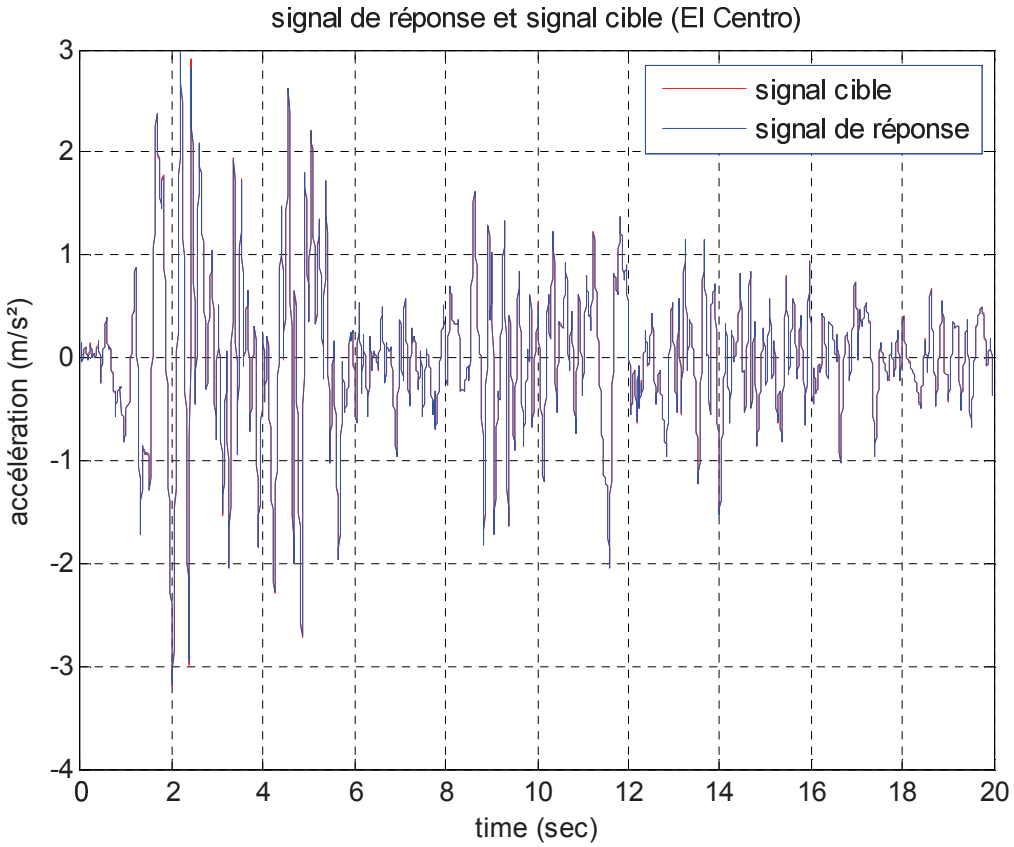


Figure 1.2.12. Différence entre le signal de réponse et le signal cible du modèle N°2.

1.2-7. Accélérogramme de Kobe : R=0.99

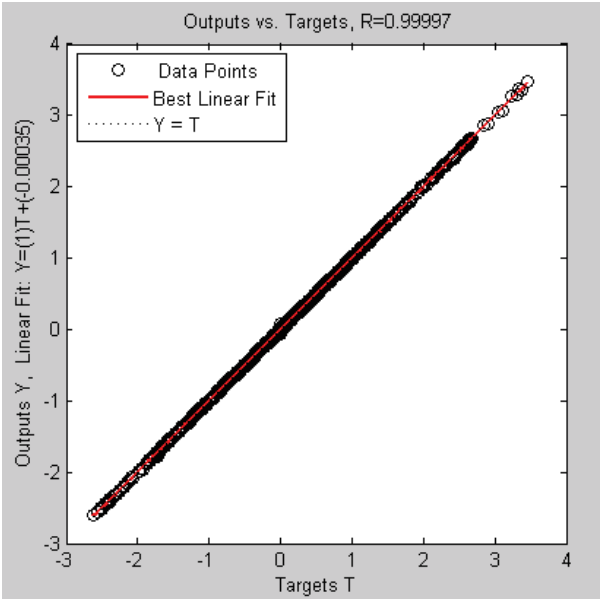


Figure 1.2.13. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°2.

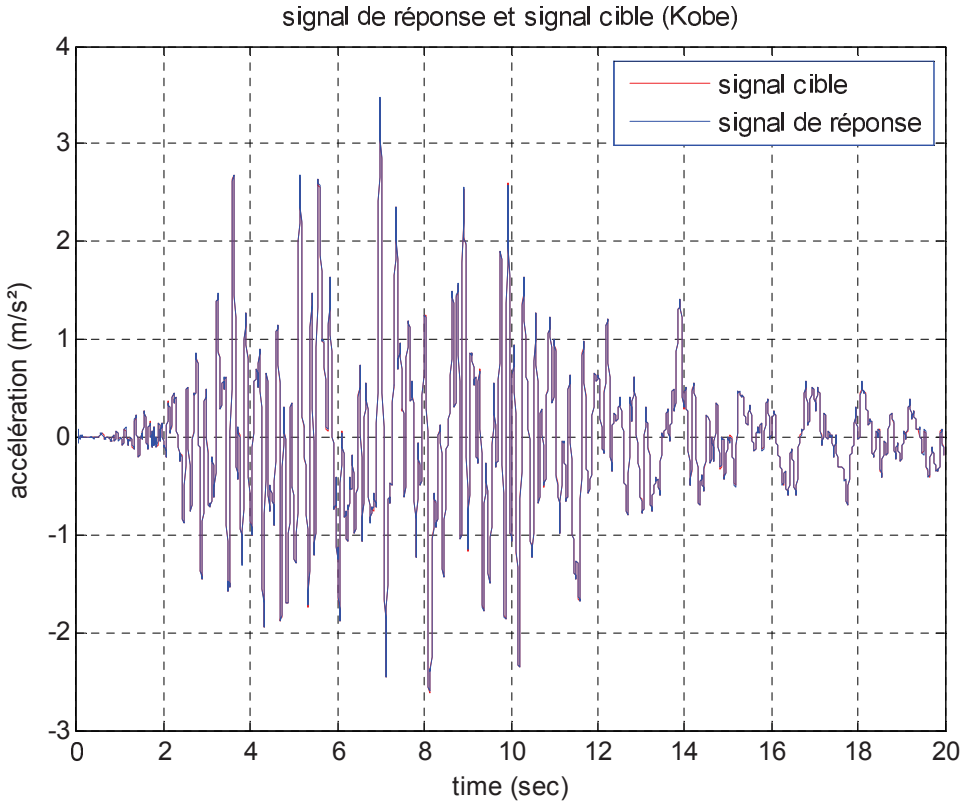


Figure 1.2.14. Différence entre le signal de réponse et le signal cible du modèle N°2.

1.2-8. Signal sinusoïdal : R=0.99

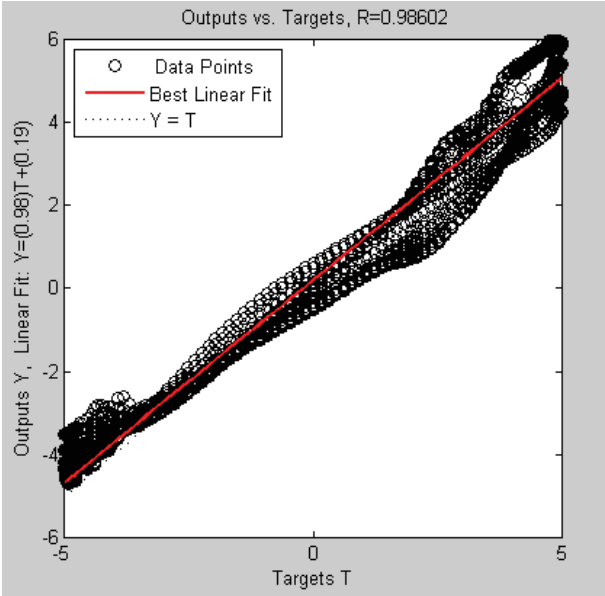


Figure 1.2.15. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°2.

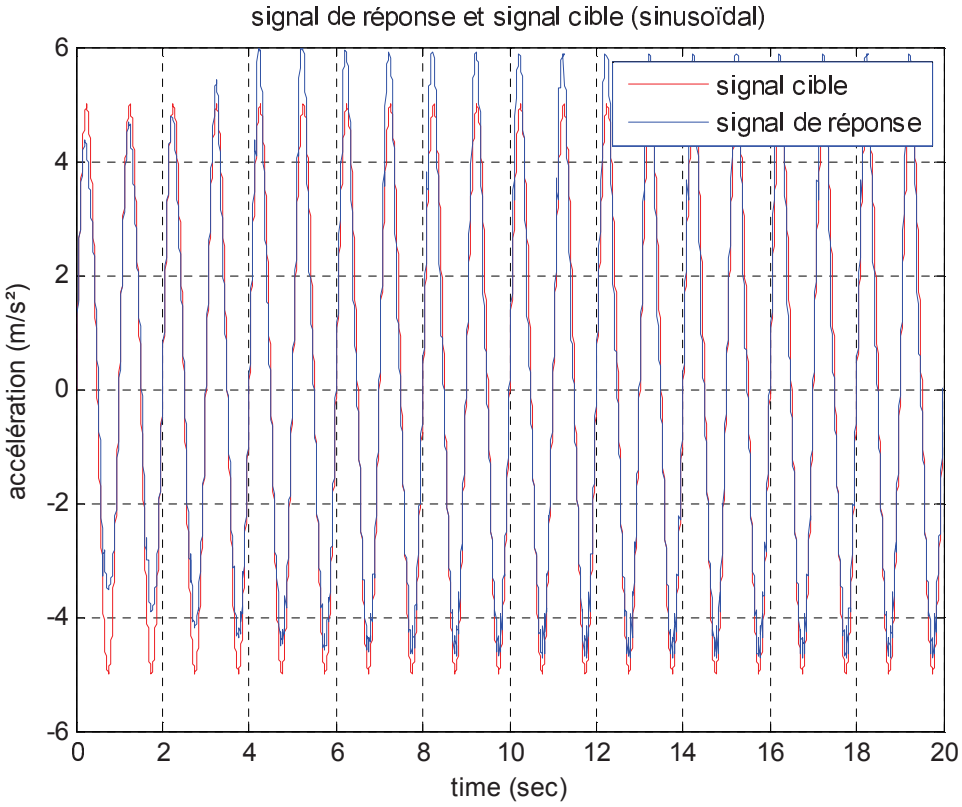


Figure 1.2.16. Différence entre le signal de réponse et le signal cible du modèle N°2.

1.3-Modèle N°3 :

1.3-1. Accélérogramme de Dar El Beida :

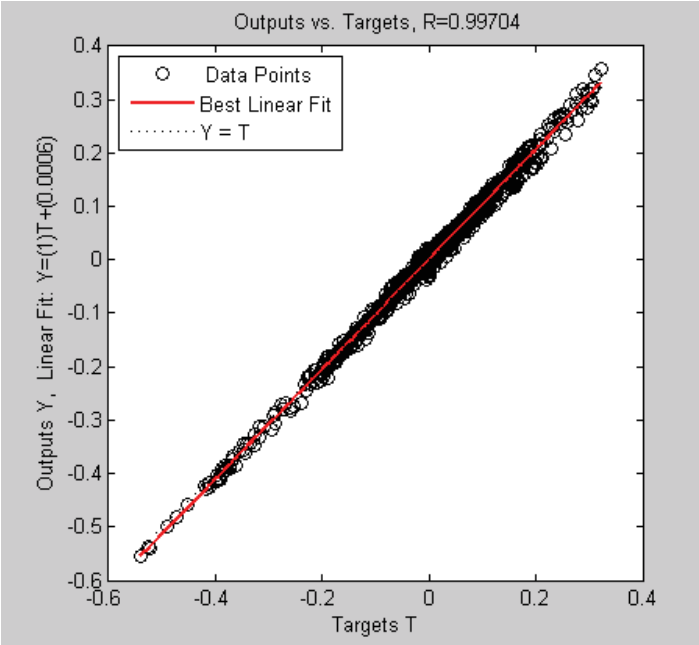


Figure 1.3.1. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°3.

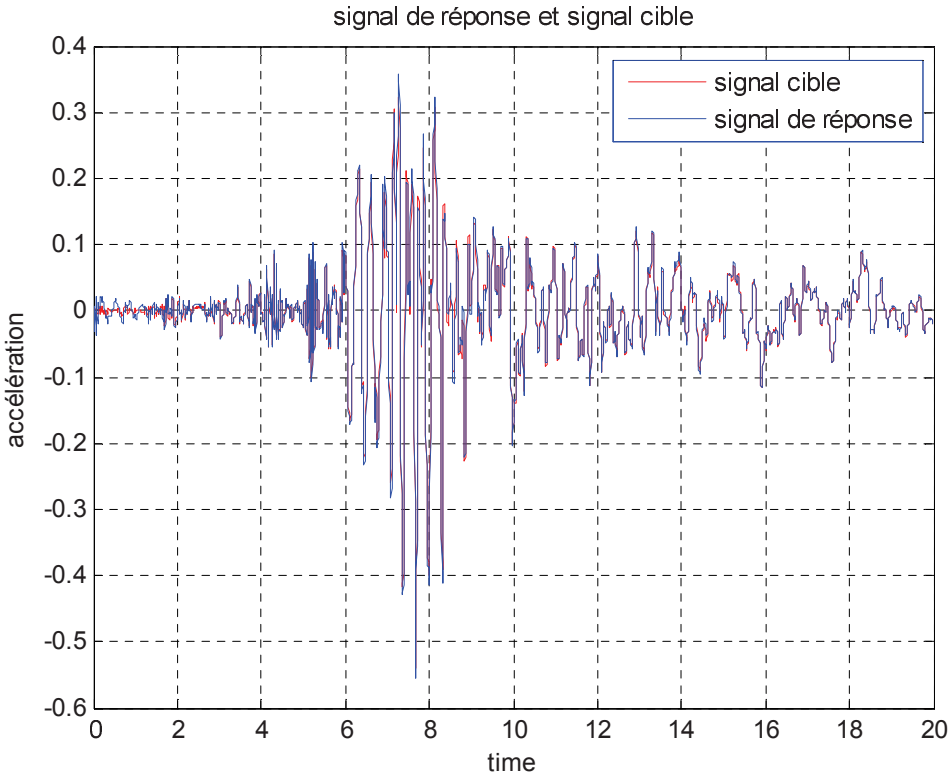


Figure 1.3.2. Différence entre le signal de réponse et le signal cible du modèle N°3.

1.3-2. Accélérogramme de Kedara : R=0.99

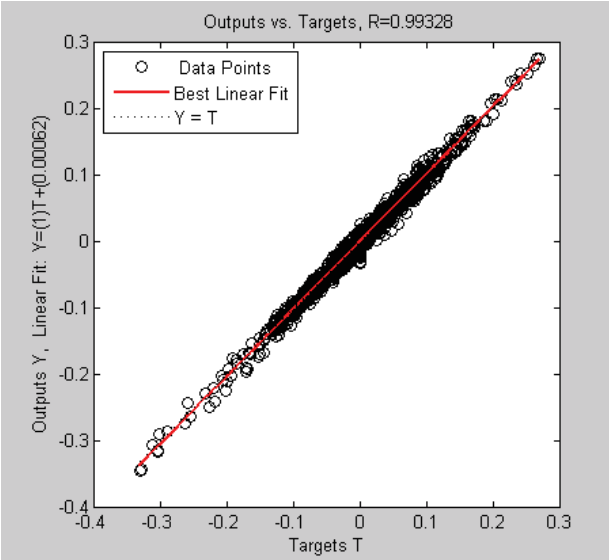


Figure 1.3.3. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°3.

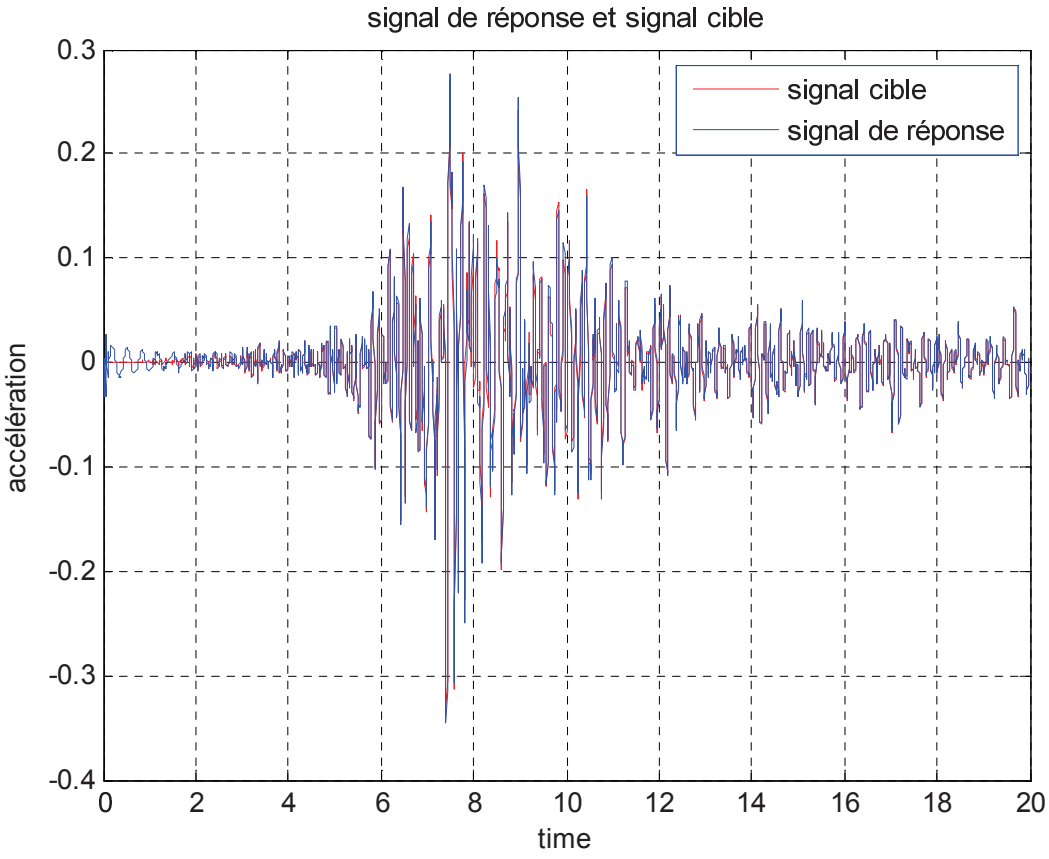


Figure 1.3.4. Différence entre le signal de réponse et le signal cible du modèle N°3.

1.3-3. Accélérogramme de Blida : R=0.99

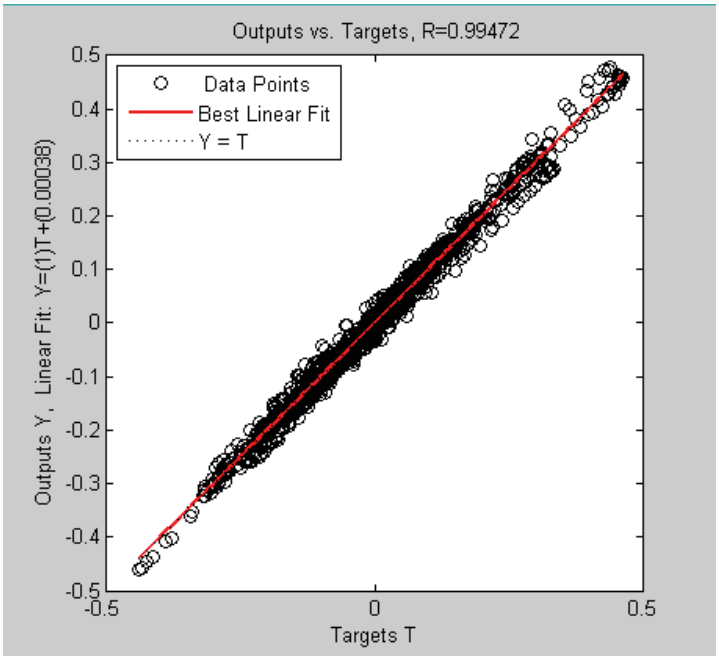


Figure 1.3.5. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°3.

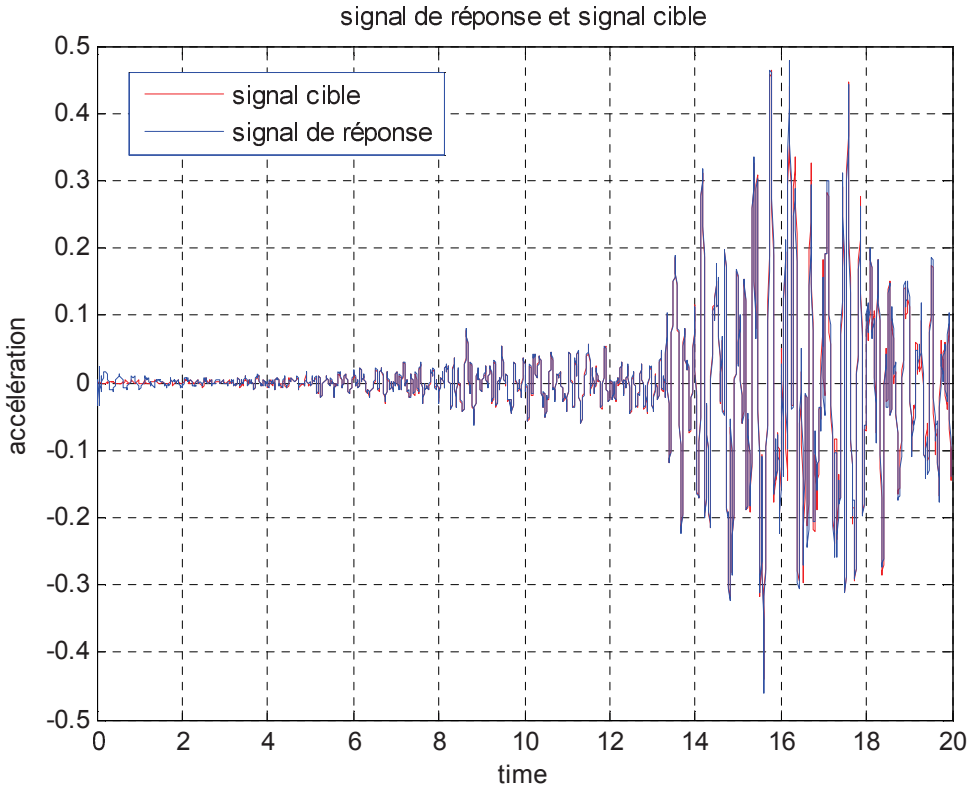


Figure 1.3.6. Différence entre le signal de réponse et le signal cible du modèle N°3.

1.3-4. Accélérogramme d'Hussein Dey : R=0.99

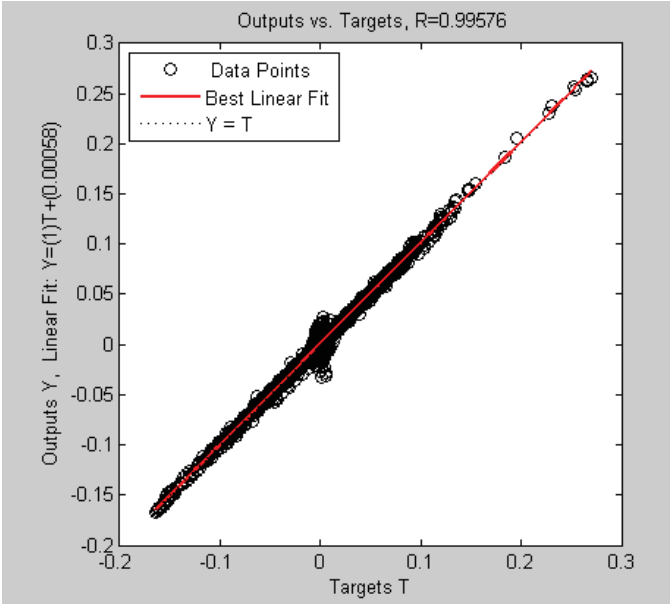


Figure 1.3.7. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°3.

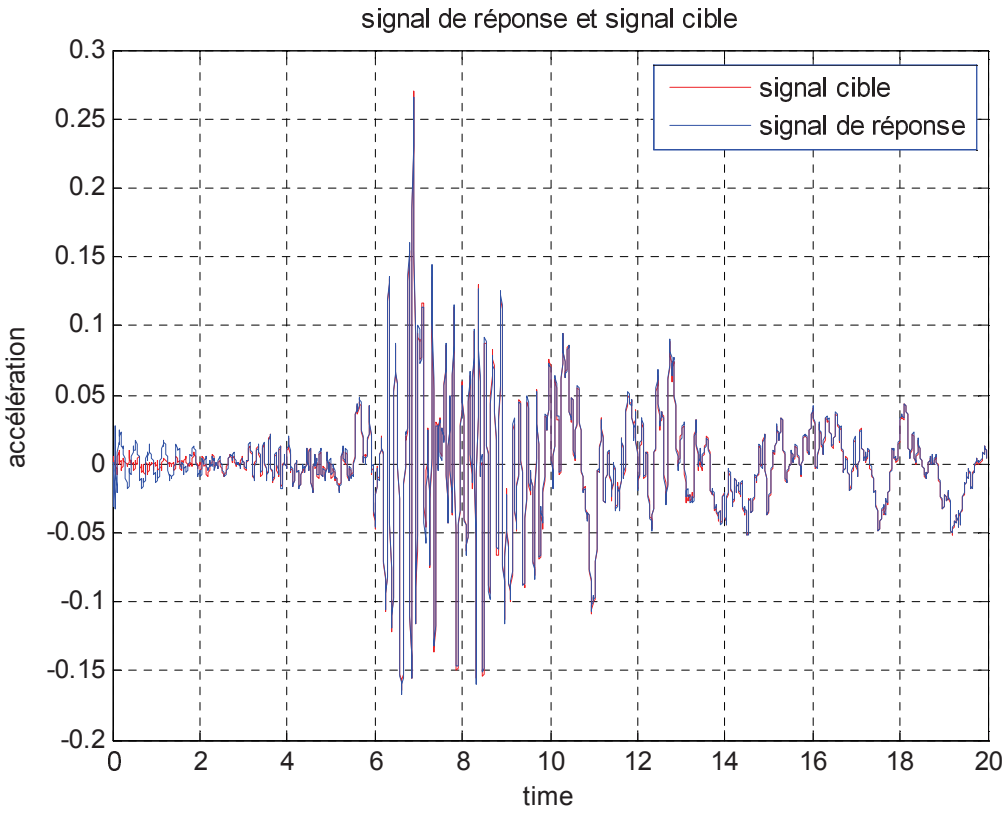


Figure 1.3.8. Différence entre le signal de réponse et le signal cible du modèle N°3.

1.3-5. Accélérogramme de Méliana : R=0.99

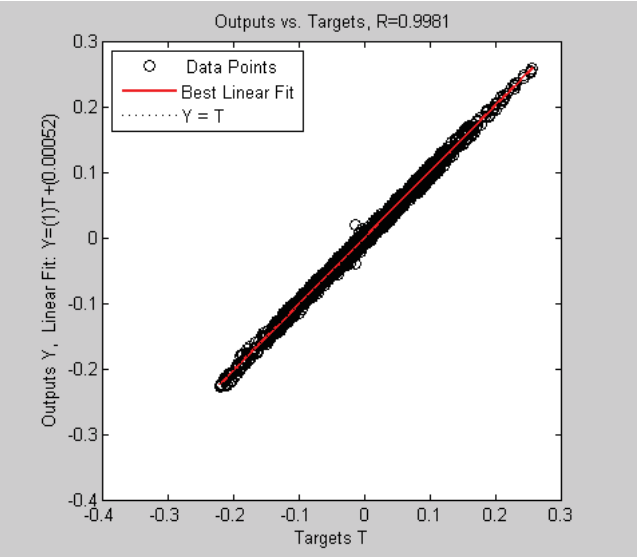


Figure 1.3.9. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°3.

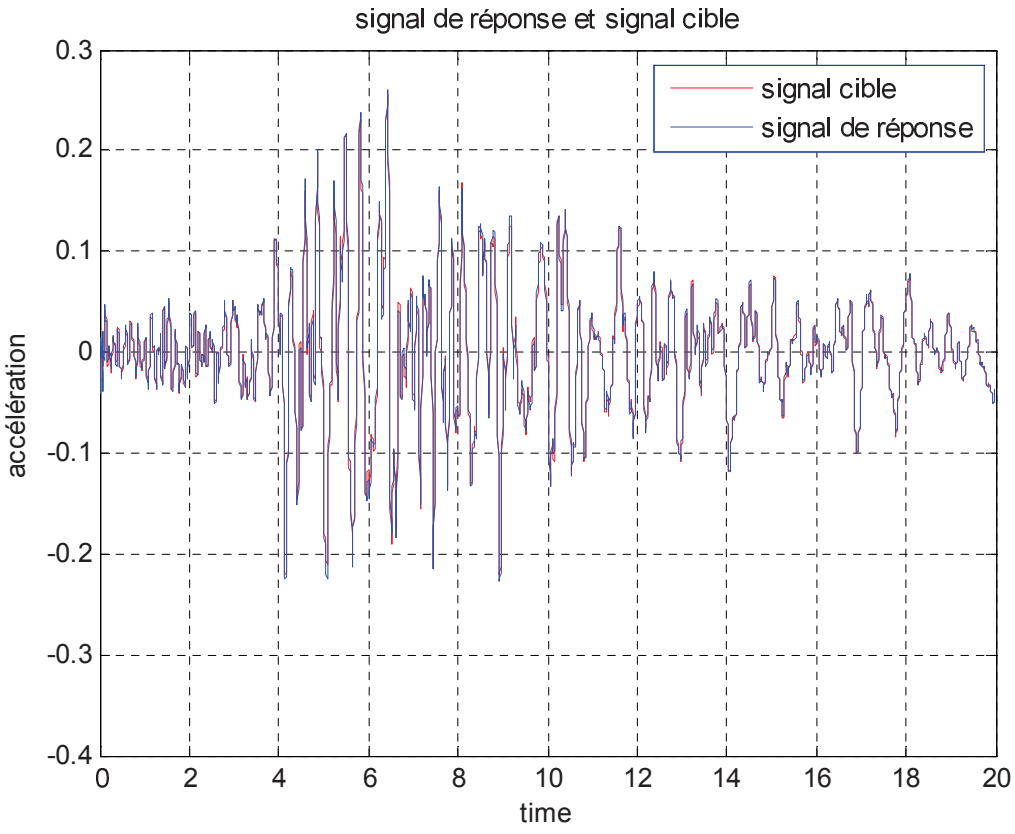


Figure 1.3.10. Différence entre le signal de réponse et le signal cible du modèle N°3.

1.3-6. Accélérogramme d'El Centro : R=0.99

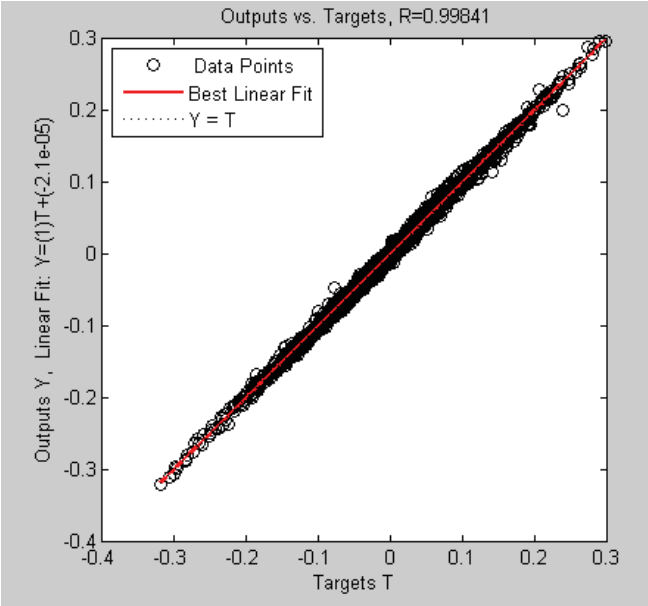


Figure 1.3.11. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°3.

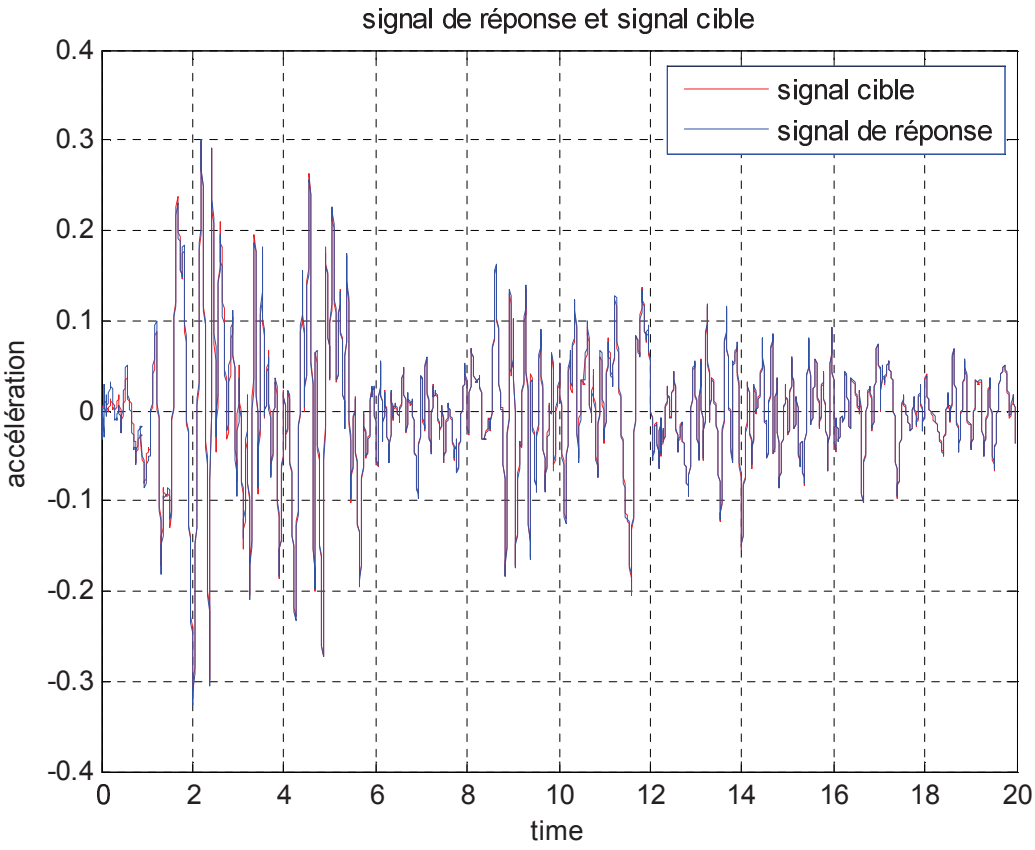


Figure 1.3.12. Différence entre le signal de réponse et le signal cible du modèle N°3.

1.3-7. Accélérogramme de Kobe : R=0.99

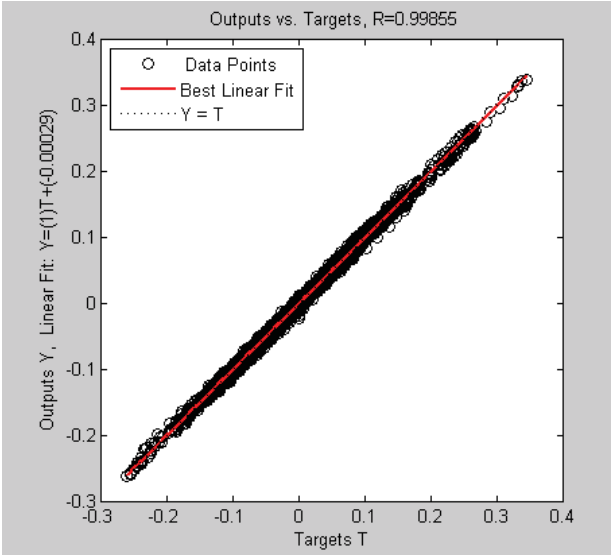


Figure 1.3.13. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°3.

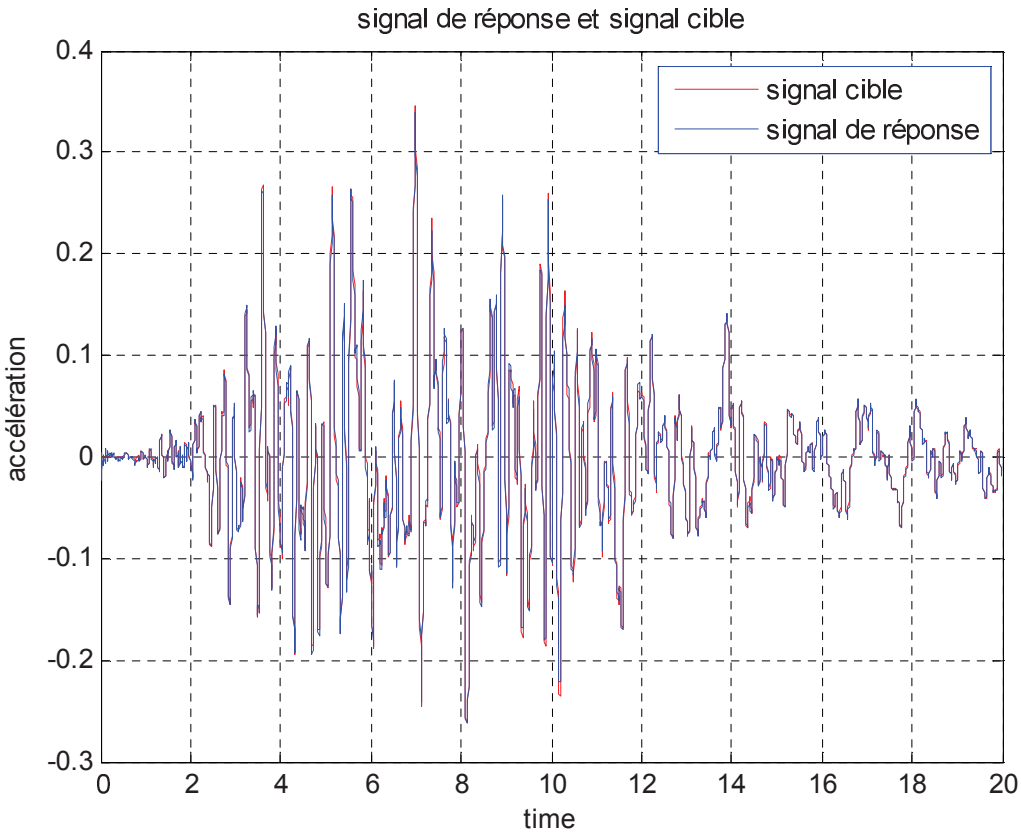


Figure 1.3.14. Différence entre le signal de réponse et le signal cible du modèle N°3.

1.3-8. Signal sinusoïdal (amplitude 0.5m/s²) : R=0.99

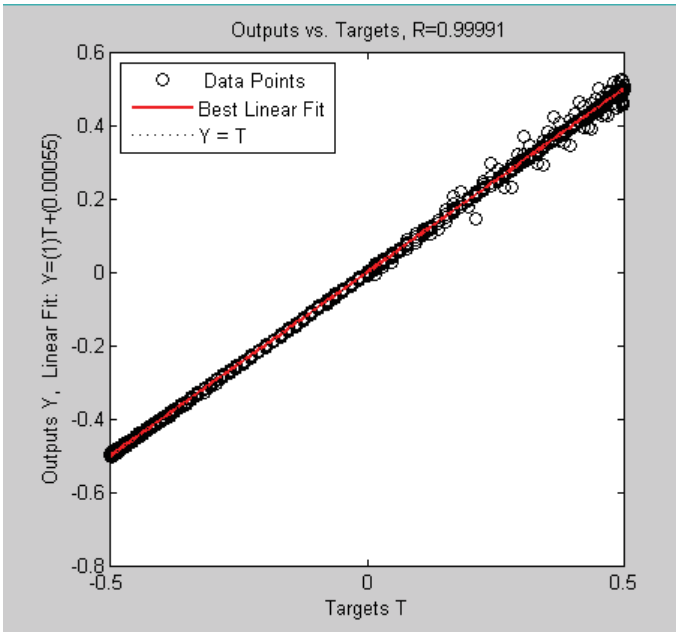


Figure 1.3.15. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°3.

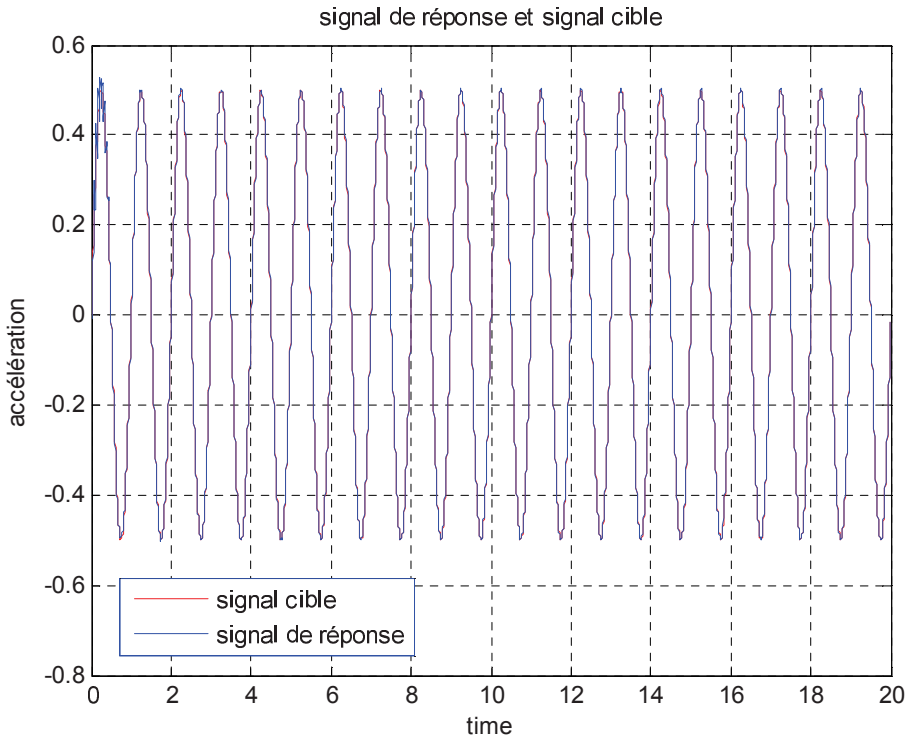


Figure 1.3.16. Différence entre le signal de réponse et le signal cible du modèle N°3.

1.4-Modèle N°4 :

1.4-1. Accélérogramme de Dar El Beida : R=0.99

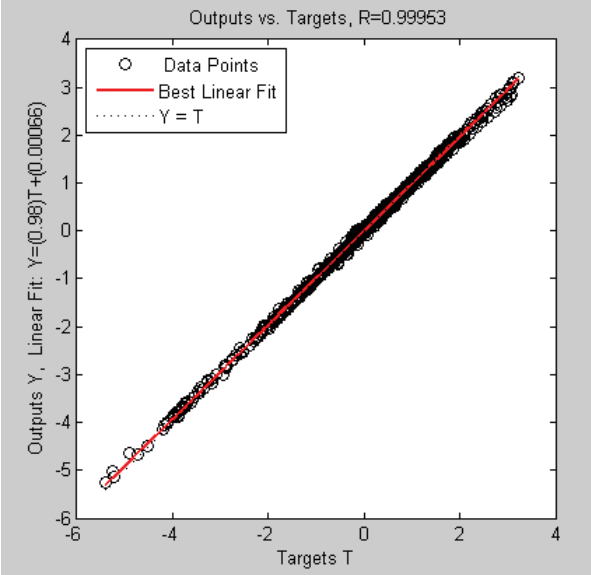


Figure 1.4.1. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°4.

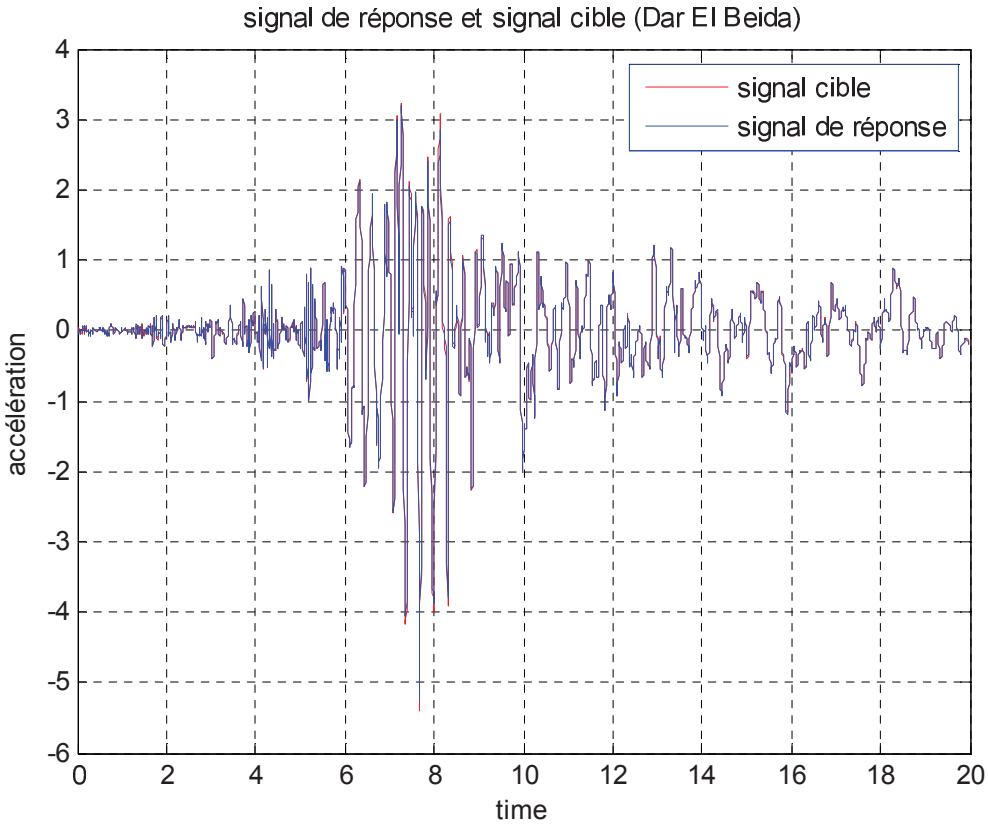


Figure 1.4.2. Différence entre le signal de réponse et le signal cible du modèle N°4.

1.4-2. Accélérogramme de Kedara : R=0.99

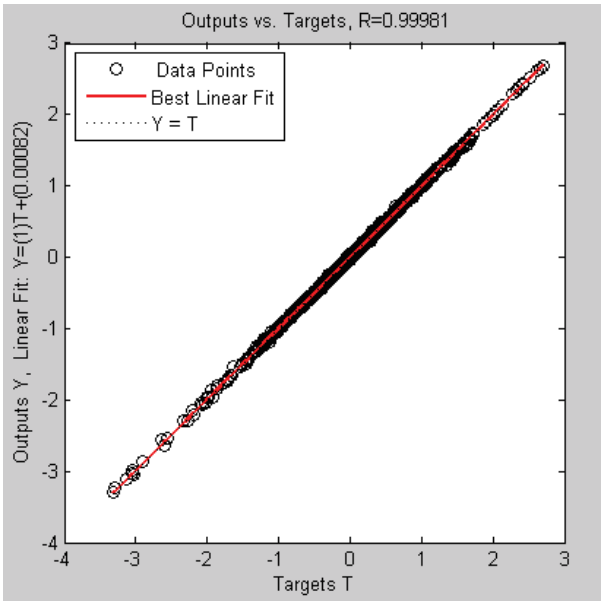


Figure 1.4.3. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°4.

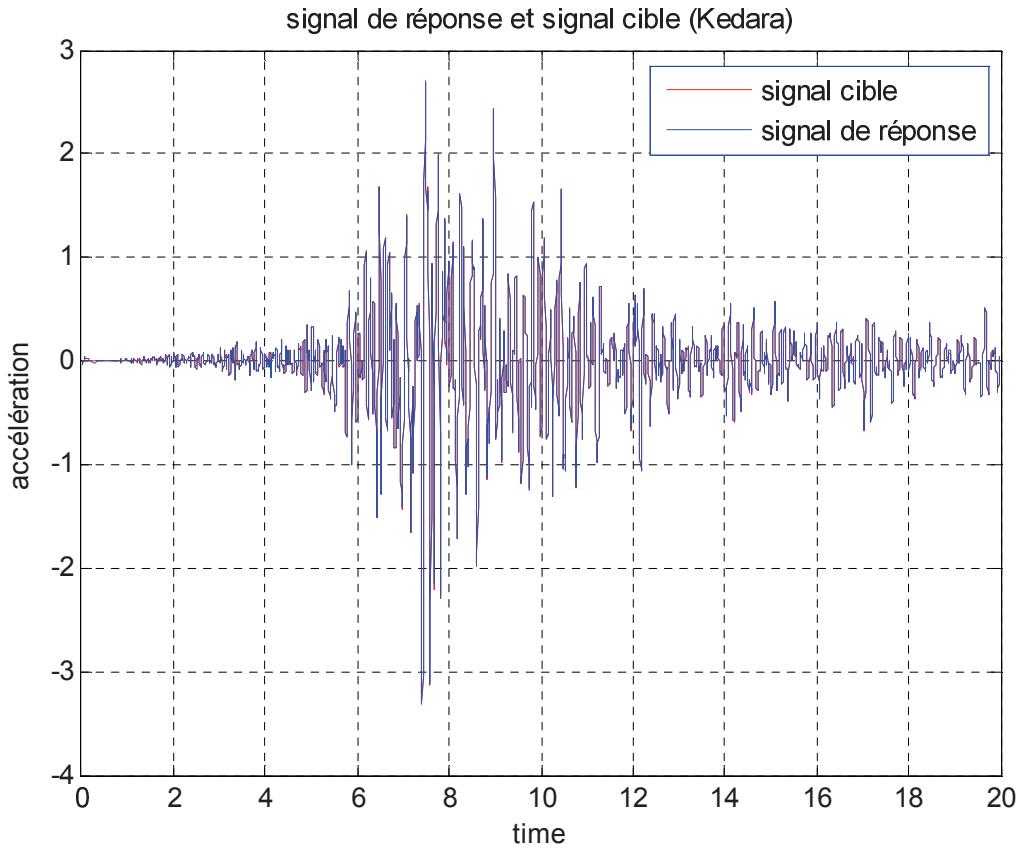


Figure 1.4.4. Différence entre le signal de réponse et le signal cible du modèle N°4.

1.4-3. Accélérogramme de Blida : R=0.99

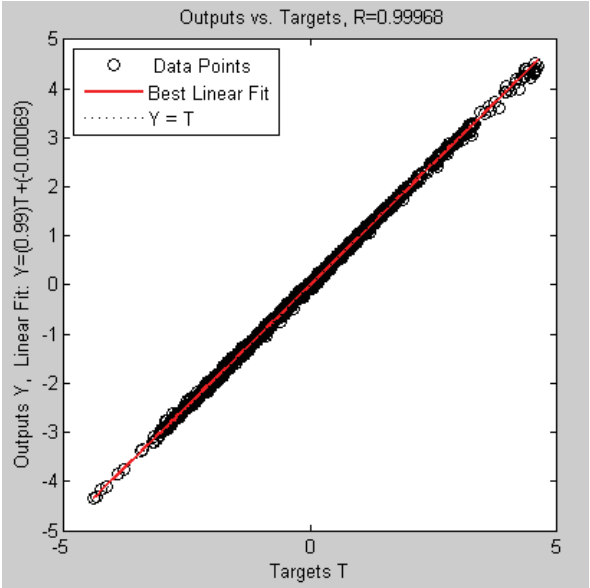


Figure 1.4.5. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°4.

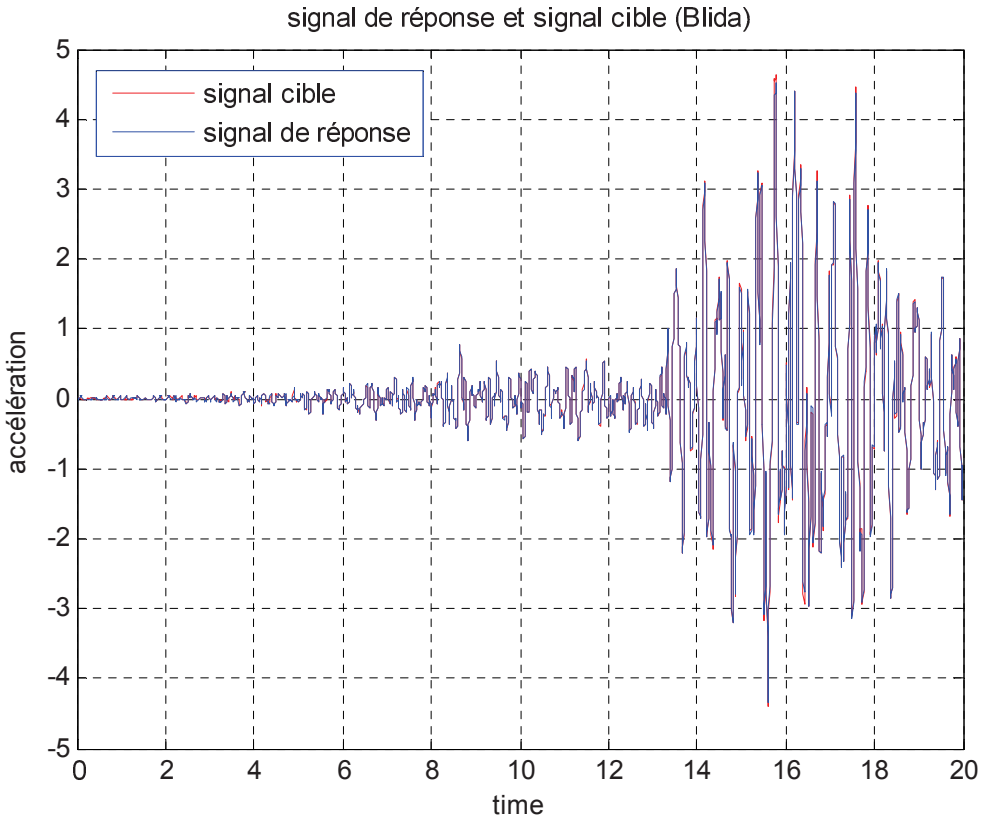


Figure 1.4.6. Différence entre le signal de réponse et le signal cible du modèle N°4.

1.4-4. Accélérogramme d'Hussen Dey : R=0.99

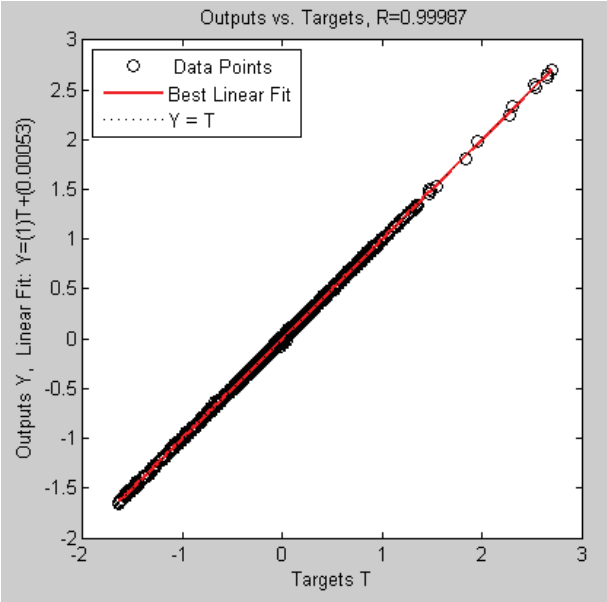


Figure 1.4.7. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°4.

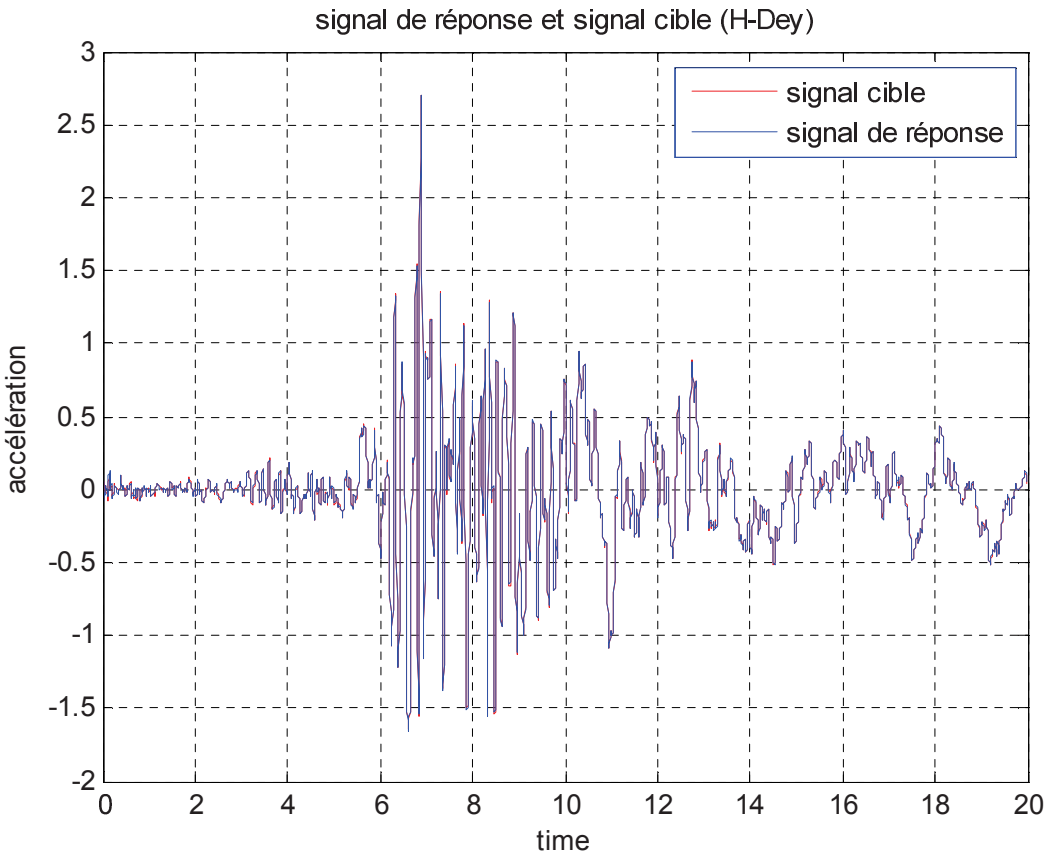


Figure 1.4.8. Différence entre le signal de réponse et le signal cible du modèle N°4.

1.4-5. Accélérogramme d'El Centro : R=0.99

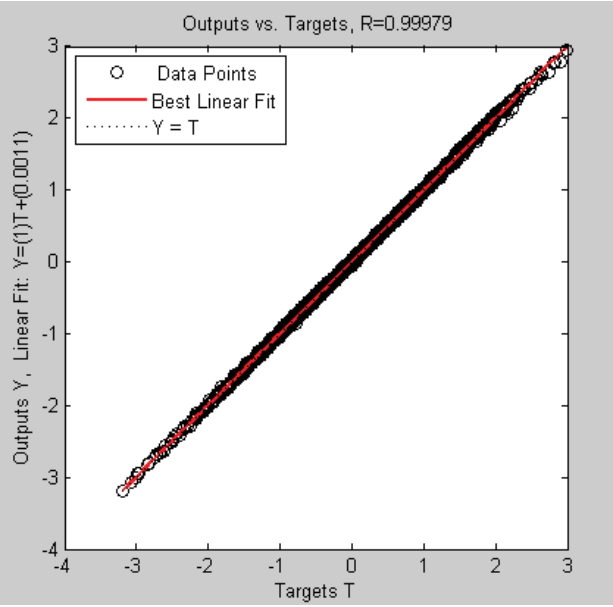


Figure 1.4.9. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°4.

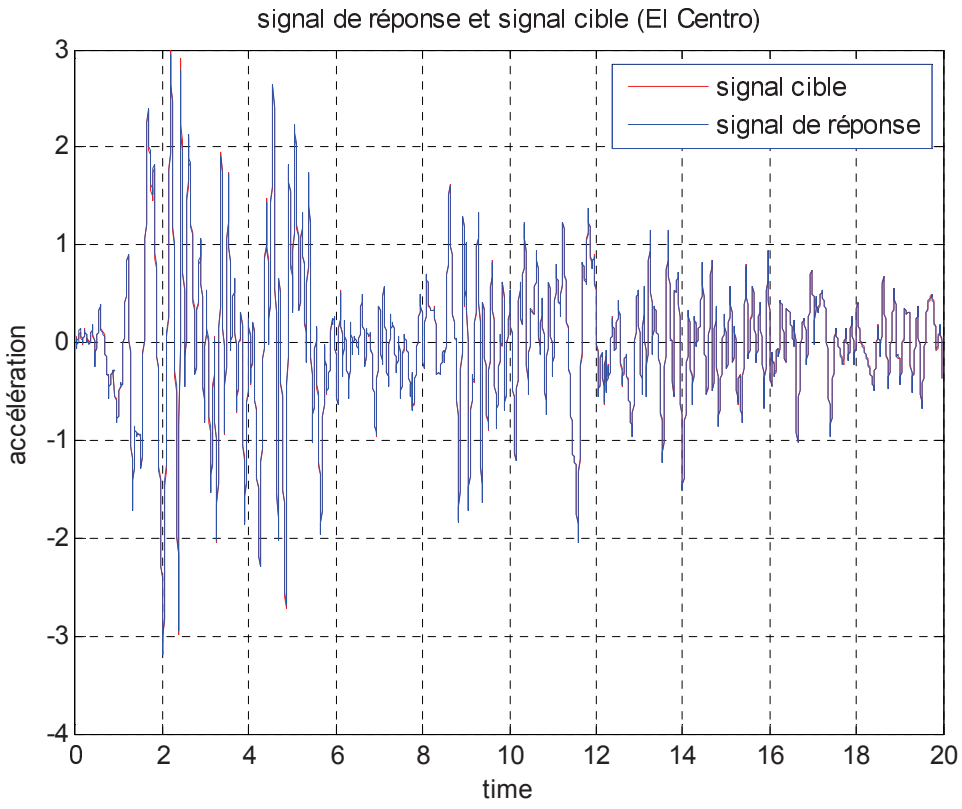


Figure 1.4.10. Différence entre le signal de réponse et le signal cible du modèle N°4.

1.4-6. Accélérogramme de Kobe : R=0.99

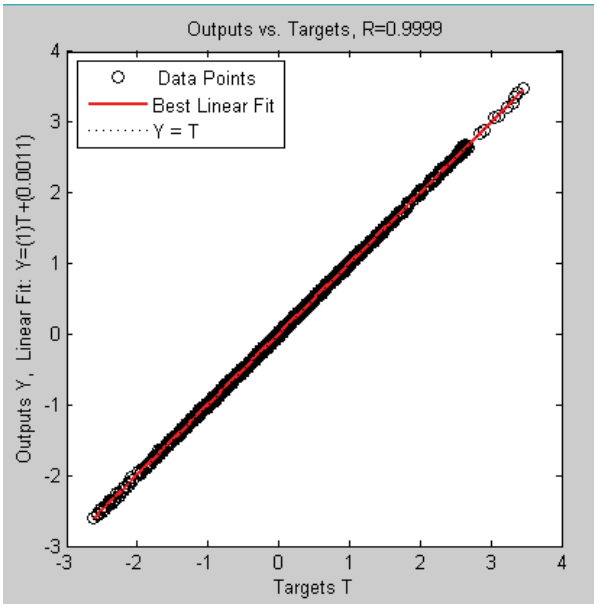


Figure 1.4.11. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°4.

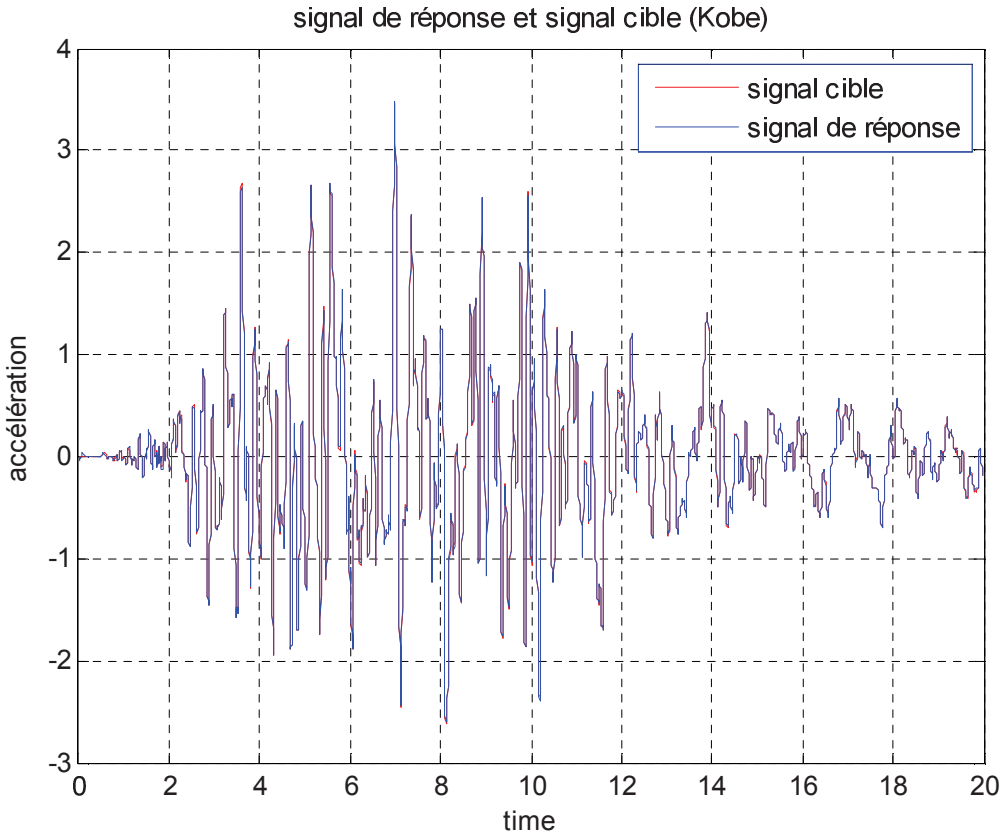


Figure 1.4.12. Différence entre le signal de réponse et le signal cible du modèle N°4.

1.4-7. Signal sinusoïdal : R=0.99

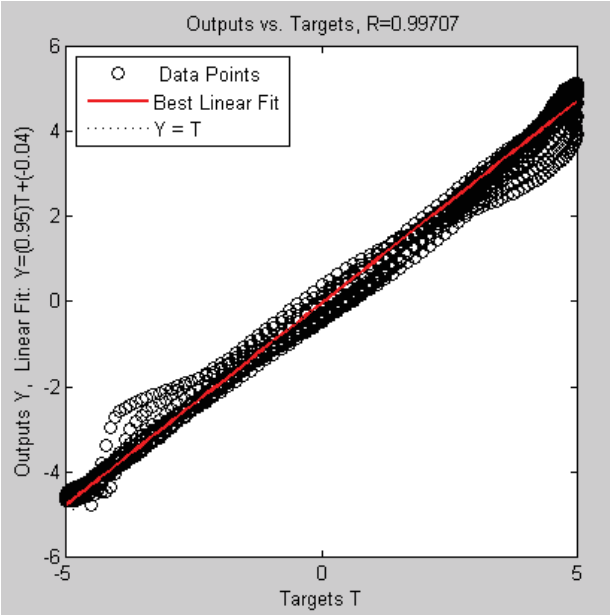


Figure 1.4.13. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°4.

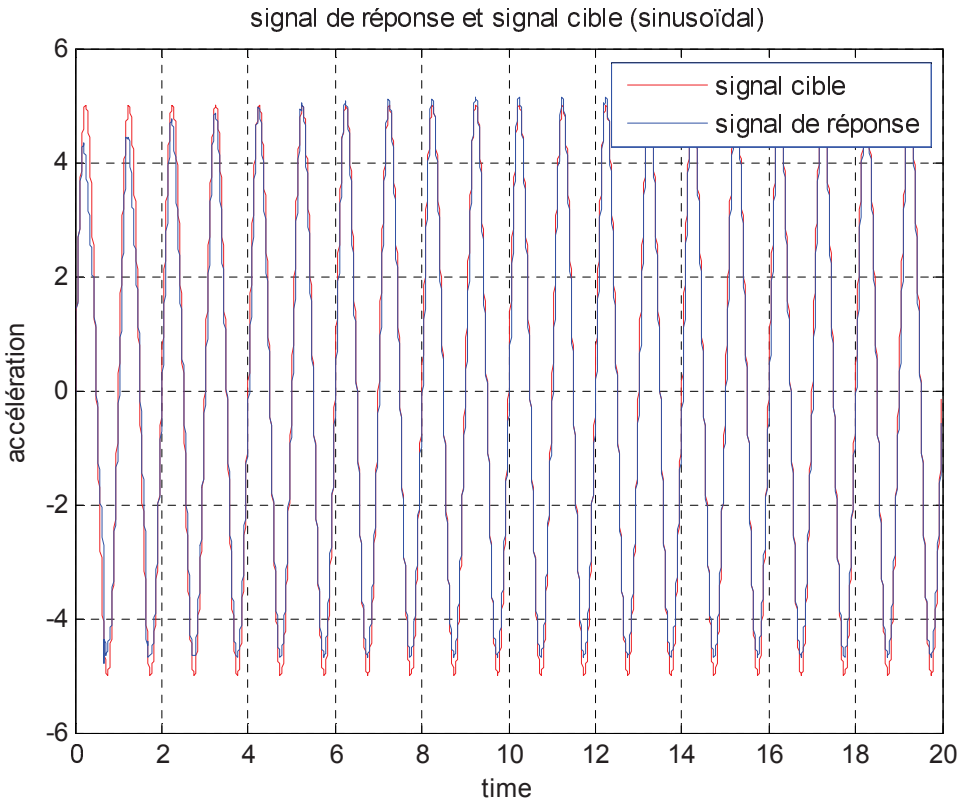


Figure 1.4.14. Différence entre le signal de réponse et le signal cible du modèle N°4.

1.5-Modèle N°5 :

1.5-1. Accélérogramme de Dar El Beida : R=0.99

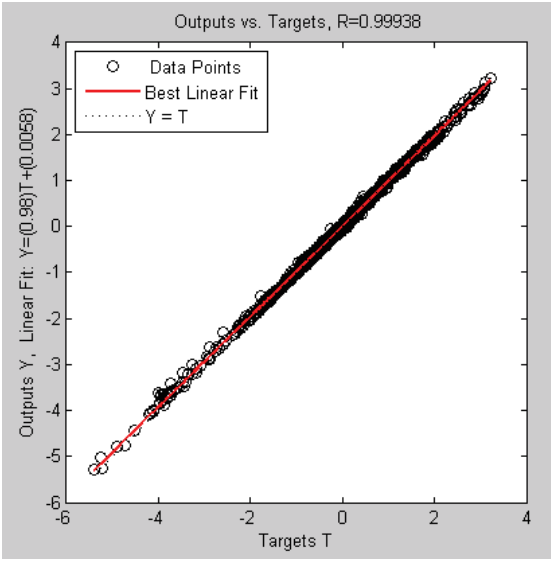


Figure 1.5.1. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°5.

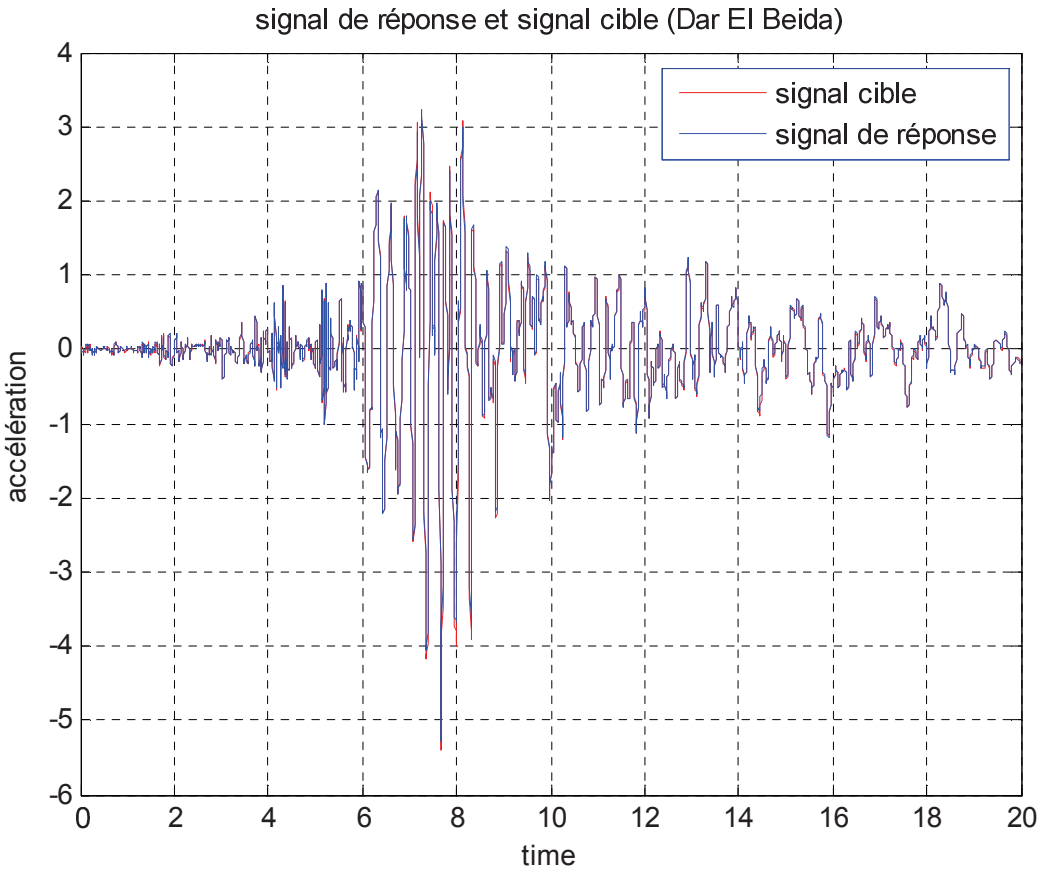


Figure 1.5.2. Différence entre le signal de réponse et le signal cible du modèle N°5.

1.5-2. Accélérogramme de Kedara : R=0.99

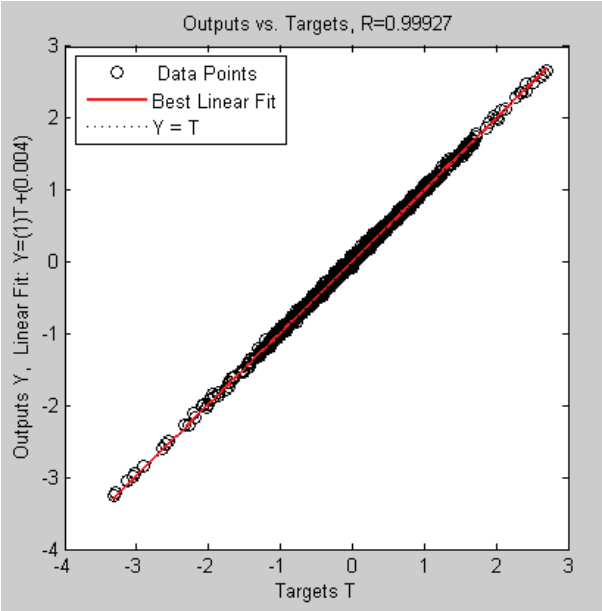


Figure 1.5.3. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°5.

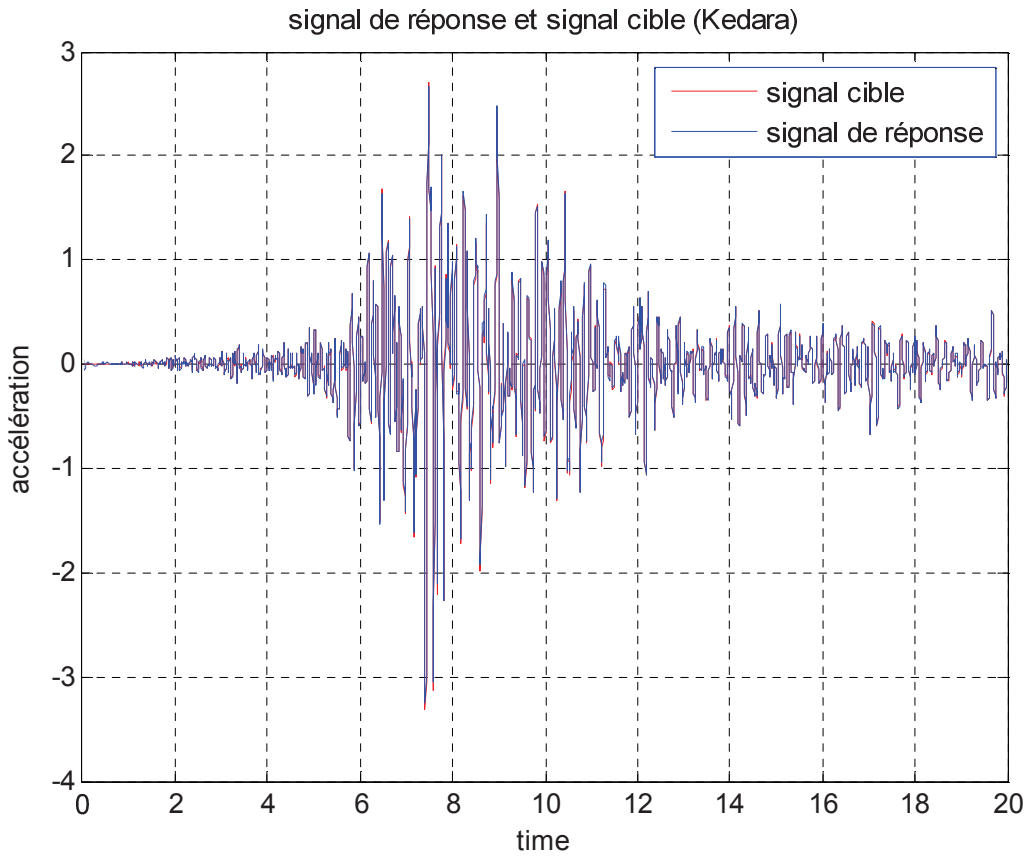


Figure 1.5.4. Différence entre le signal de réponse et le signal cible du modèle N°5.

1.5-3. Accélérogramme de Blida : R=0.99

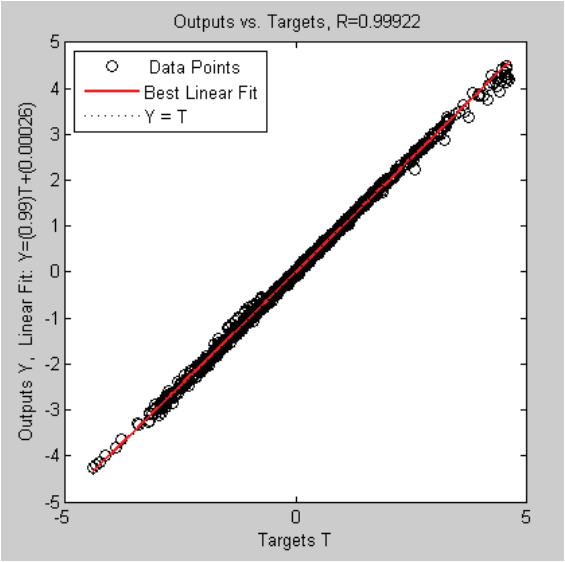


Figure 1.5.5. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°5.

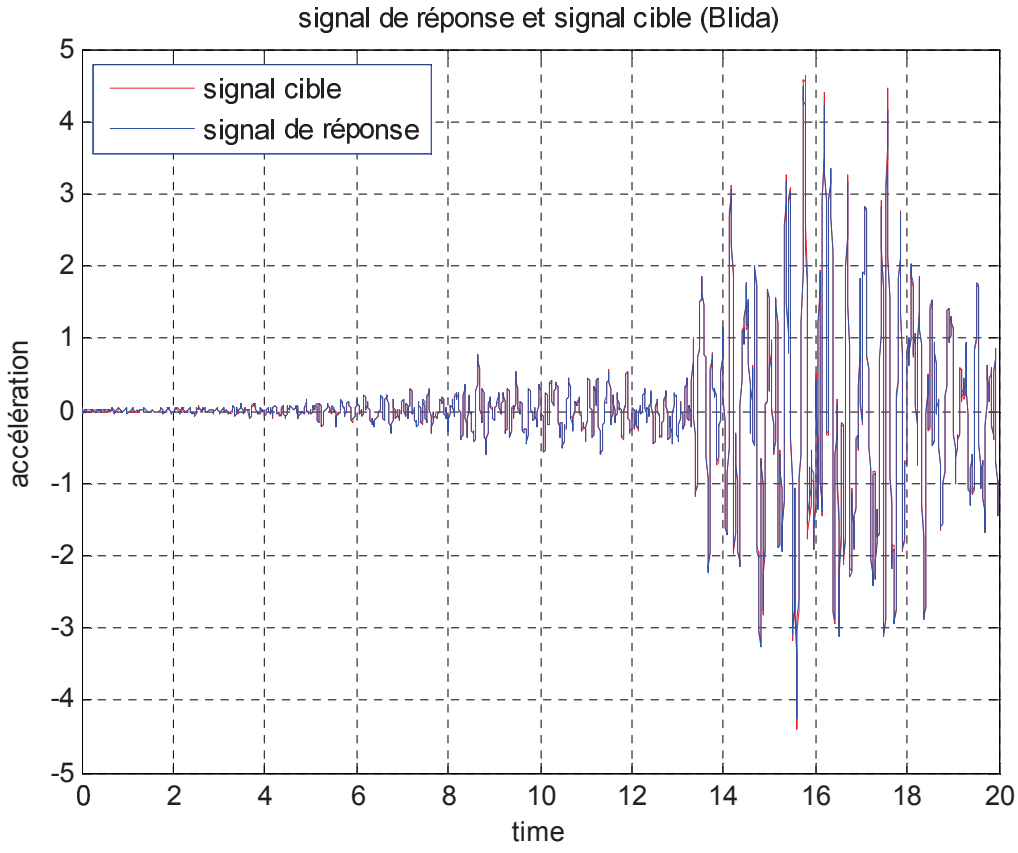


Figure 1.5.6. Différence entre le signal de réponse et le signal cible du modèle N°5.

1.5-4. Accélérogramme d'Hussein Dey : R=0.99

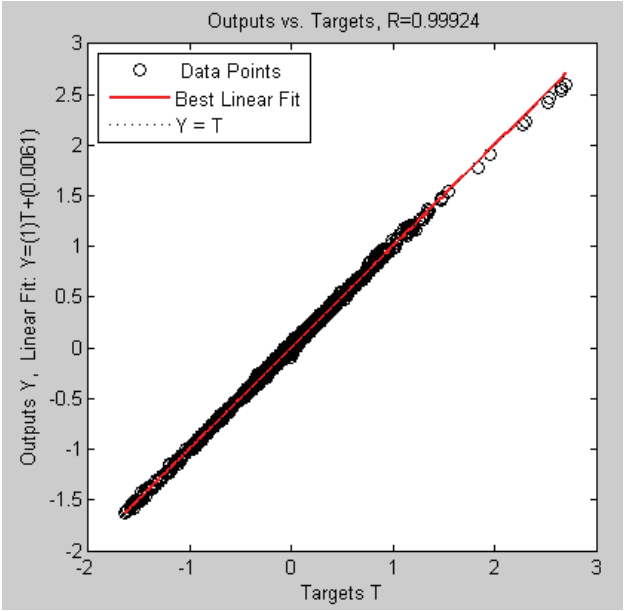


Figure 1.5.7. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°5.

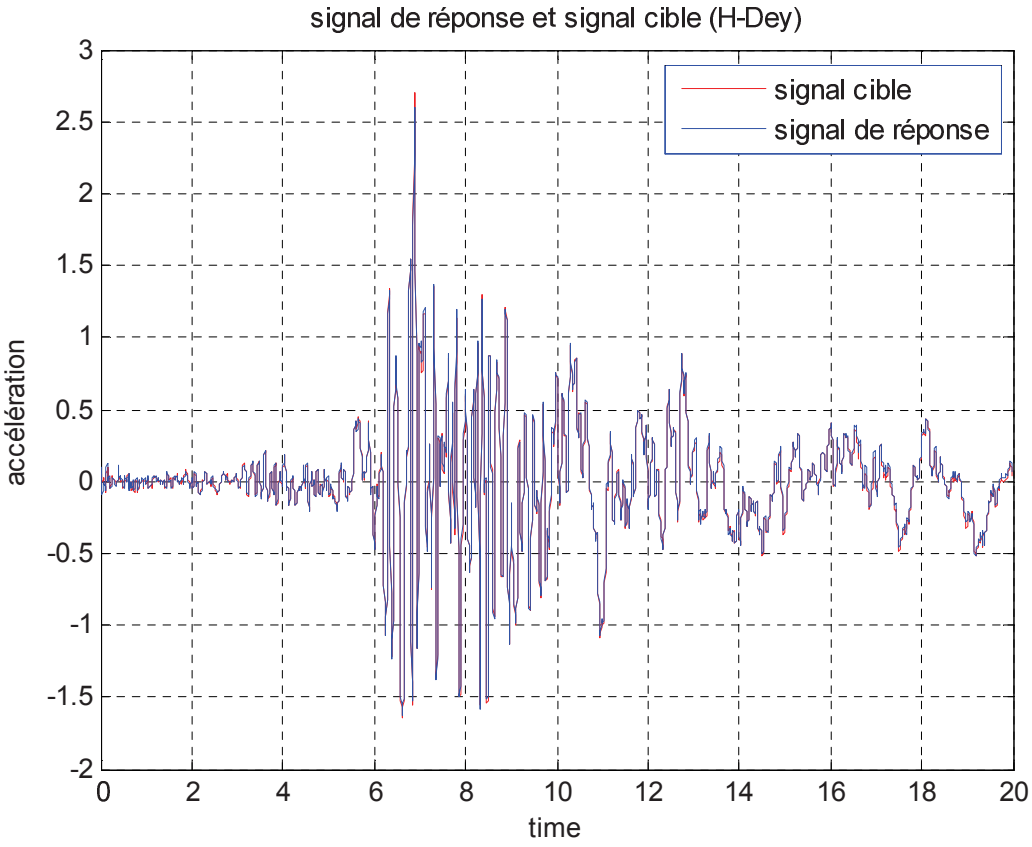


Figure 1.5.8. Différence entre le signal de réponse et le signal cible du modèle N°5.

1.5-5. Accélérogramme d'Azazga : R=0.99

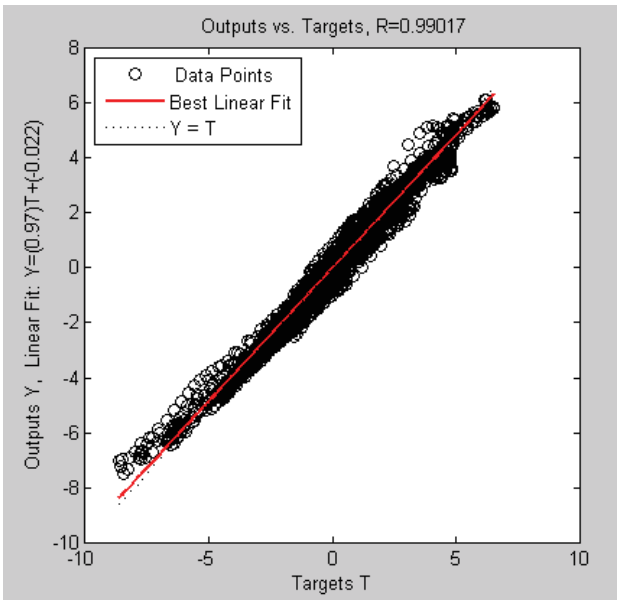


Figure 1.5.9. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°5.

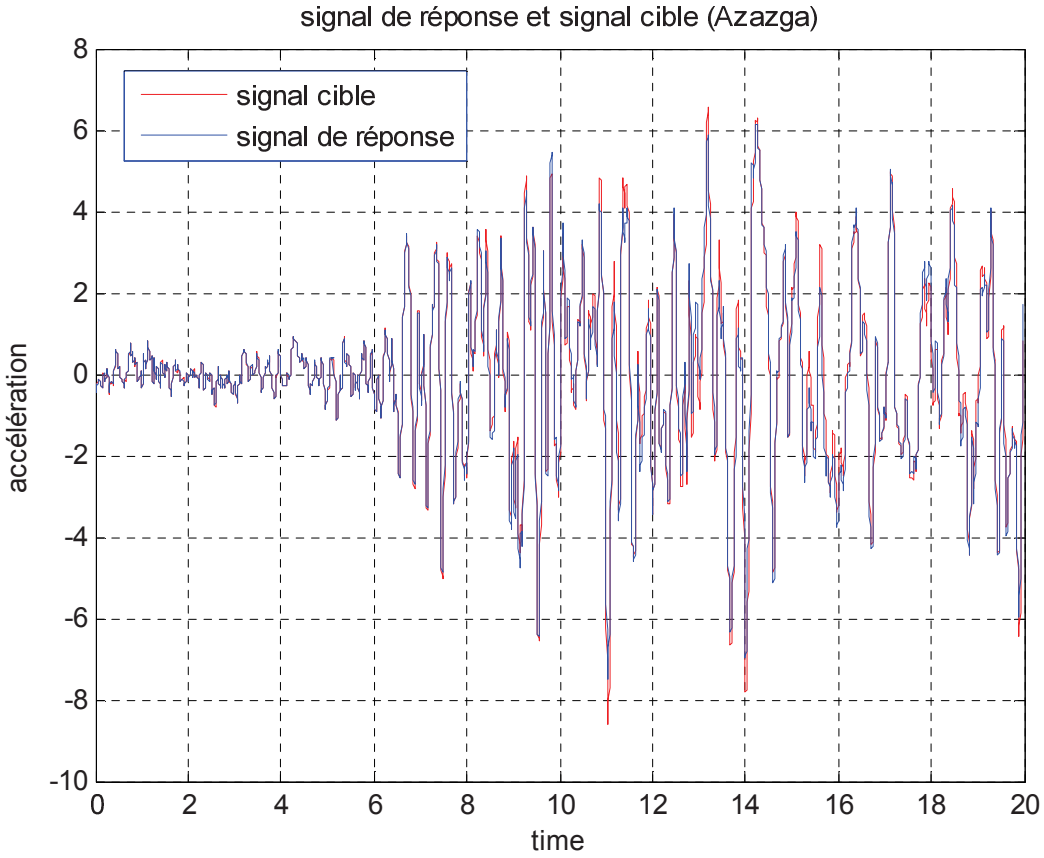


Figure 1.5.10. Différence entre le signal de réponse et le signal cible du modèle N°5.

1.5-6. Accélérogramme d'El Centro : R=0.99

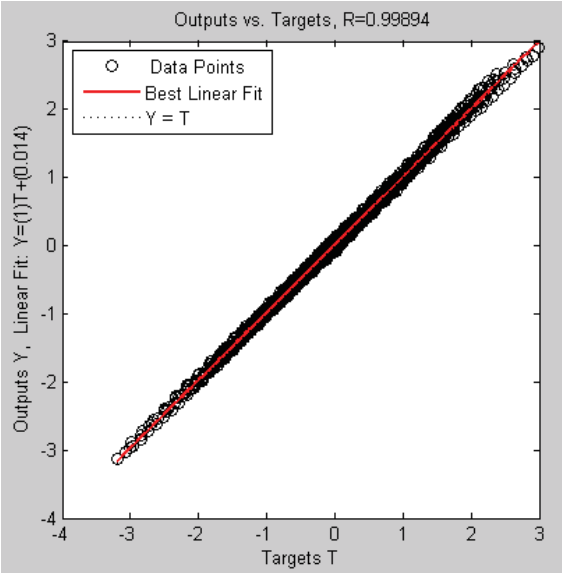


Figure 1.5.11. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°5.

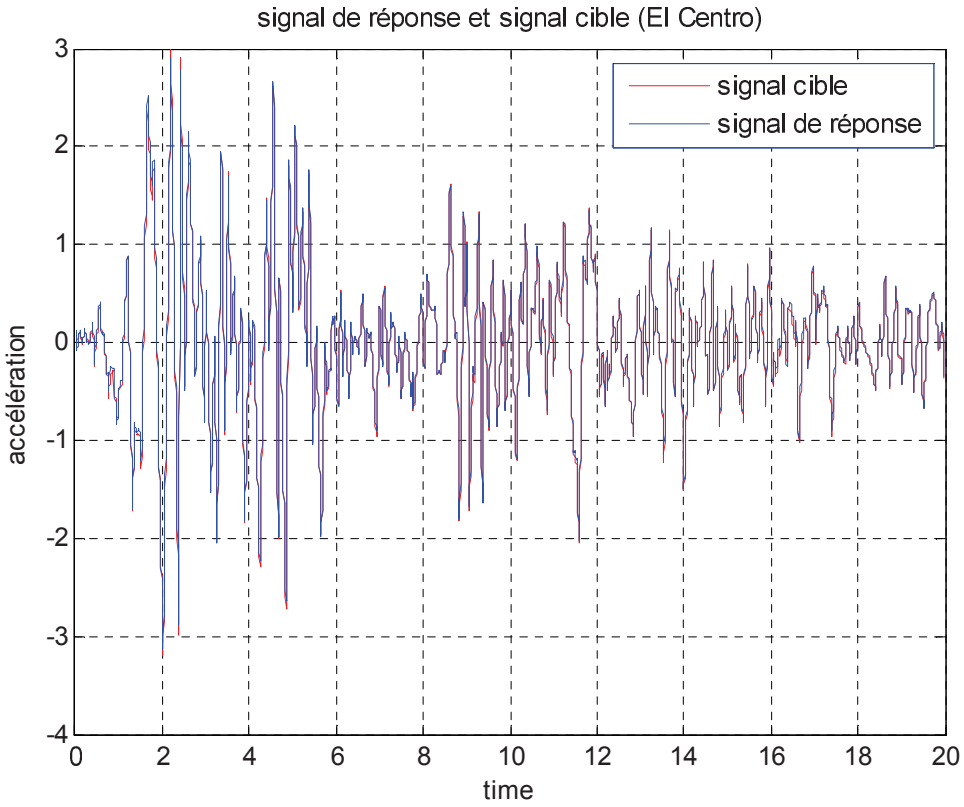


Figure 1.5.12. Différence entre le signal de réponse et le signal cible du modèle N°5.

1.5-7. Accélérogramme de Kobe : R=0.99

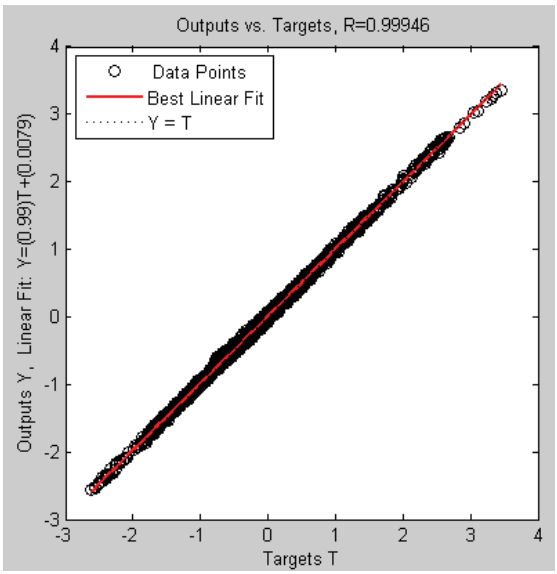


Figure 1.5.13. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°5.

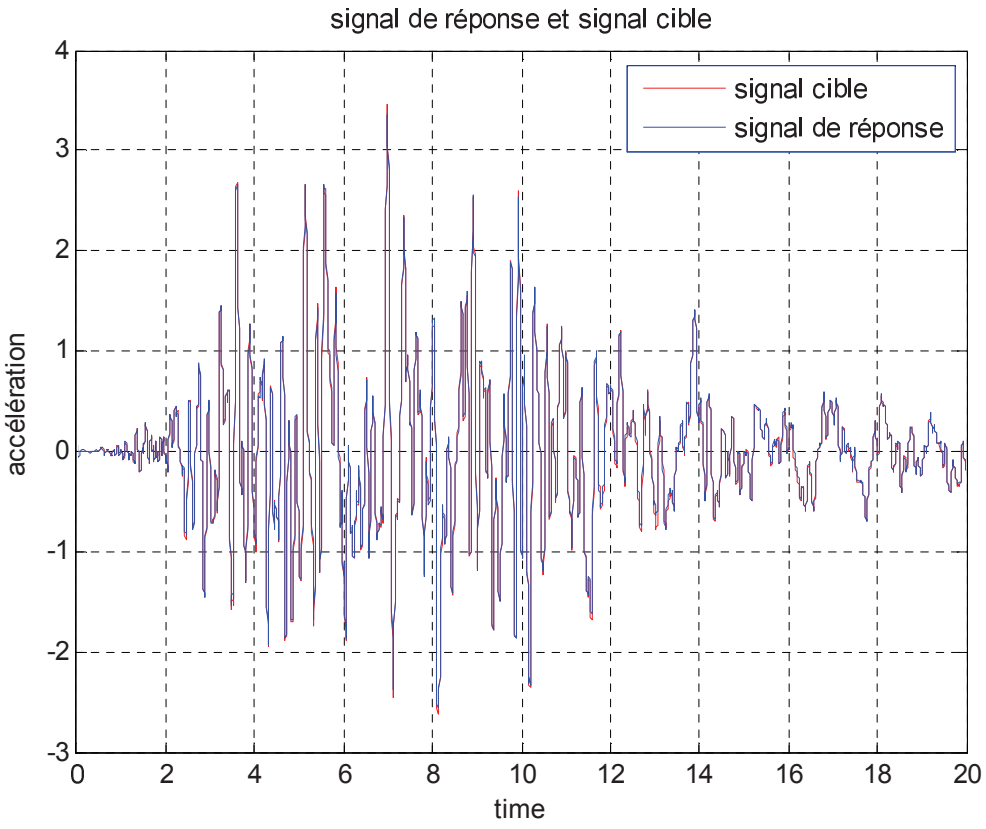


Figure 1.5.14. Différence entre le signal de réponse et le signal cible du modèle N°5.

1.5-8. Signal sinusoïdal : R=0.98

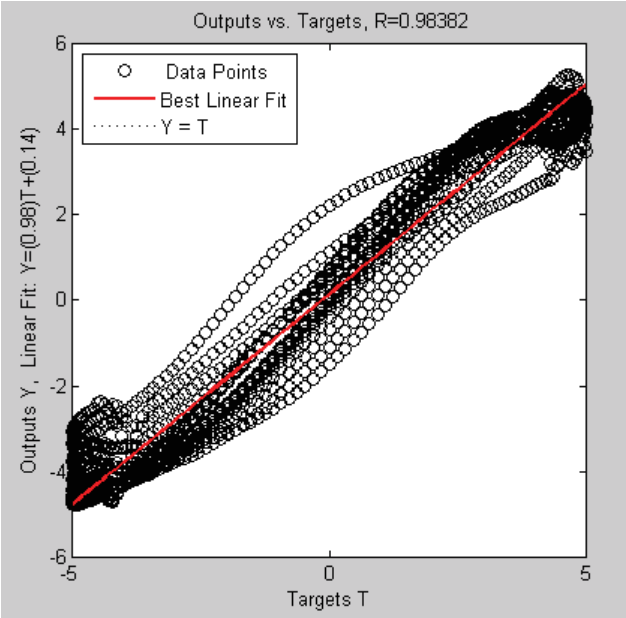


Figure 1.5.15. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°5.

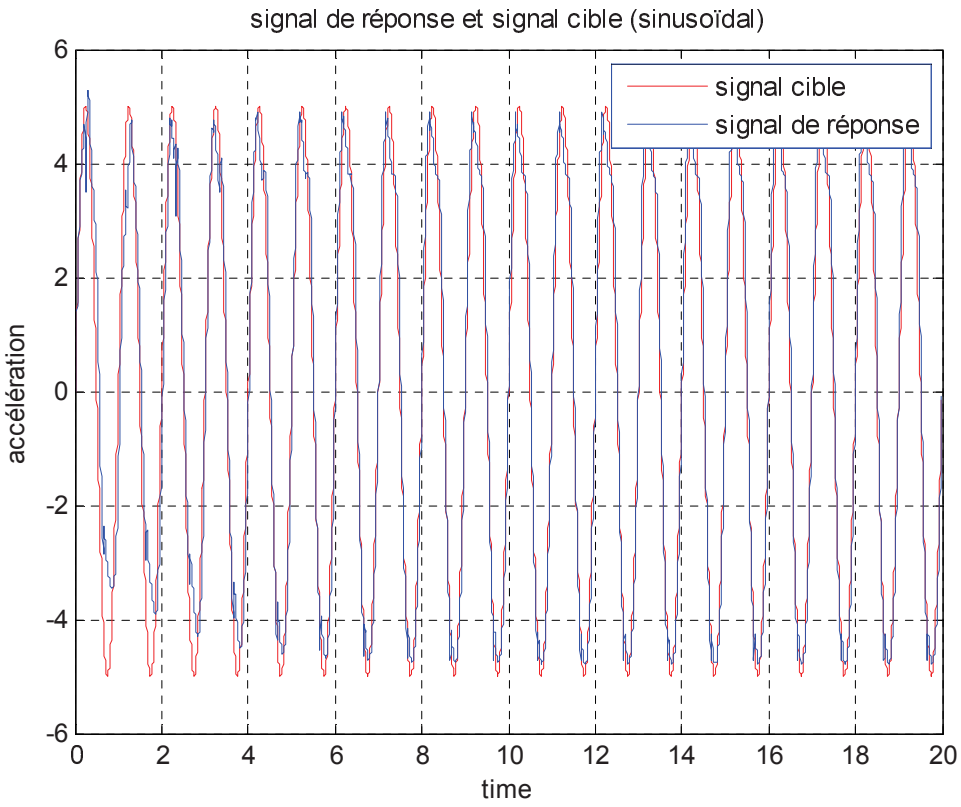


Figure 1.5.16. Différence entre le signal de réponse et le signal cible du modèle N°5.

1.6-Modèle N°6 :

1.6-1. Accélérogramme de Dar El Beida : R=0.99

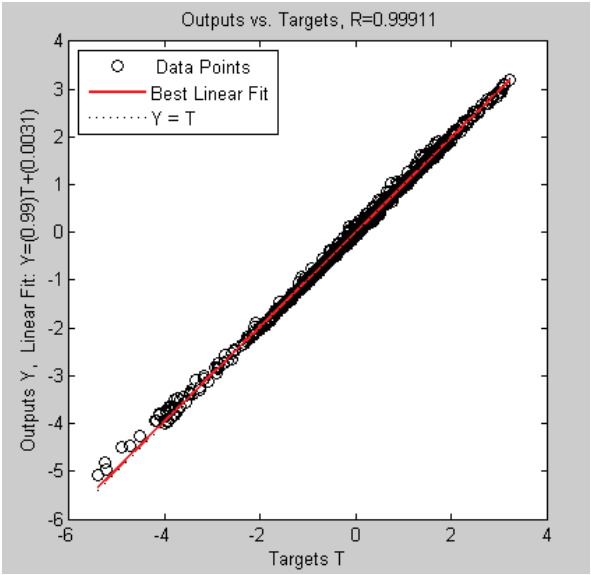


Figure 1.6.1. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°6.

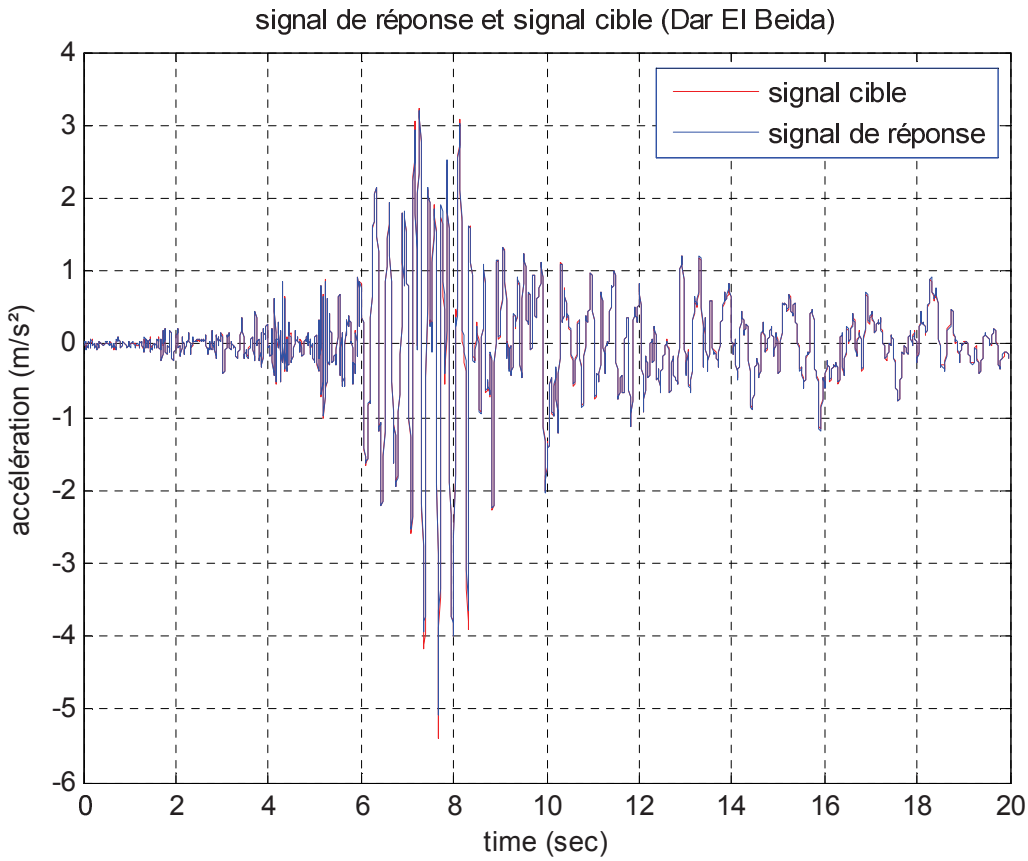


Figure 1.6.2. Différence entre le signal de réponse et le signal cible du modèle N°6.

1.6-2. Accélérogramme de Kedara : R=0.99

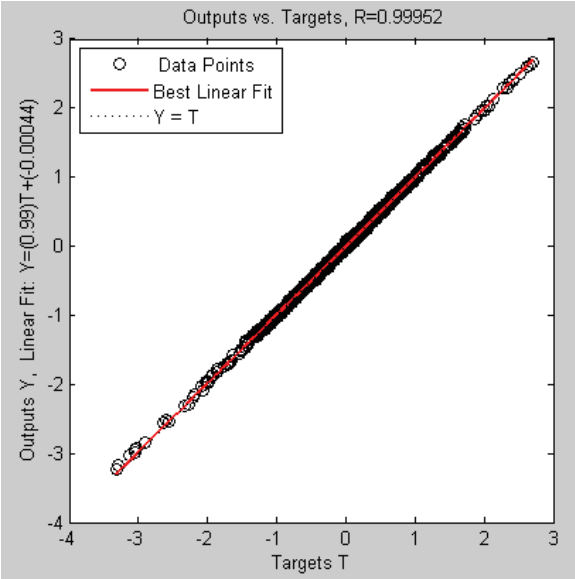


Figure 1.6.3. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°6.

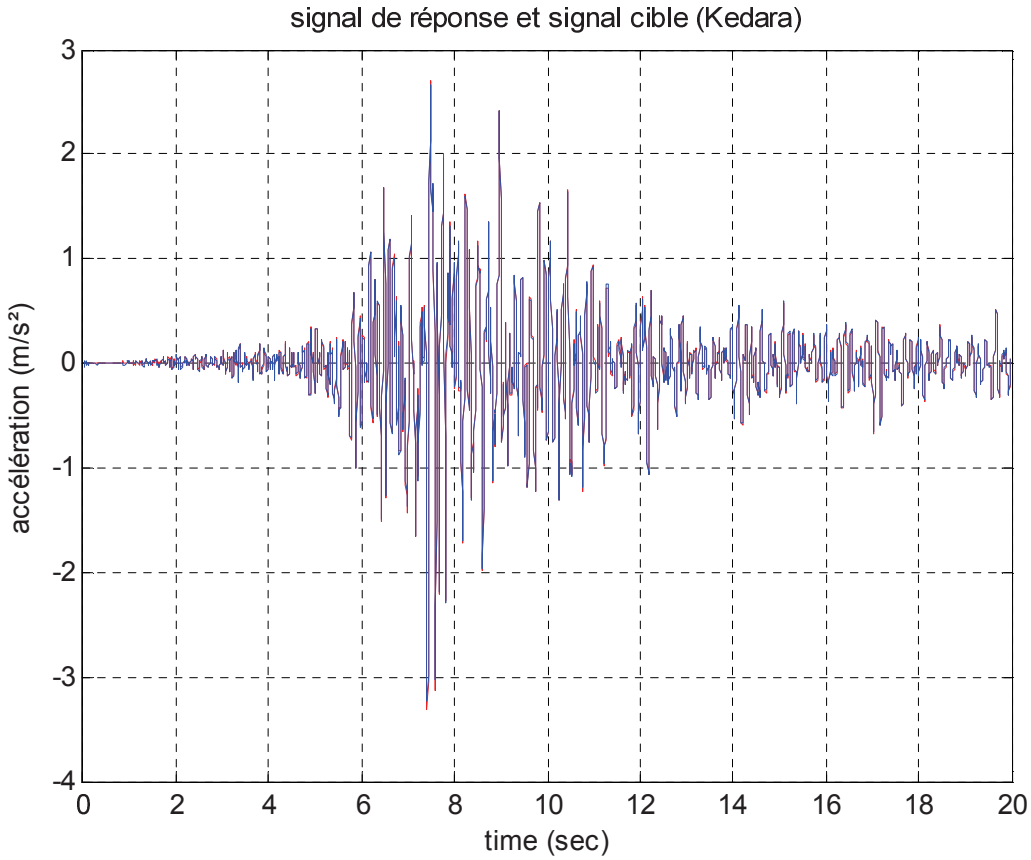


Figure 1.6.4. Différence entre le signal de réponse et le signal cible du modèle N°6.

1.6-3. Accélérogramme de Blida : R=0.99

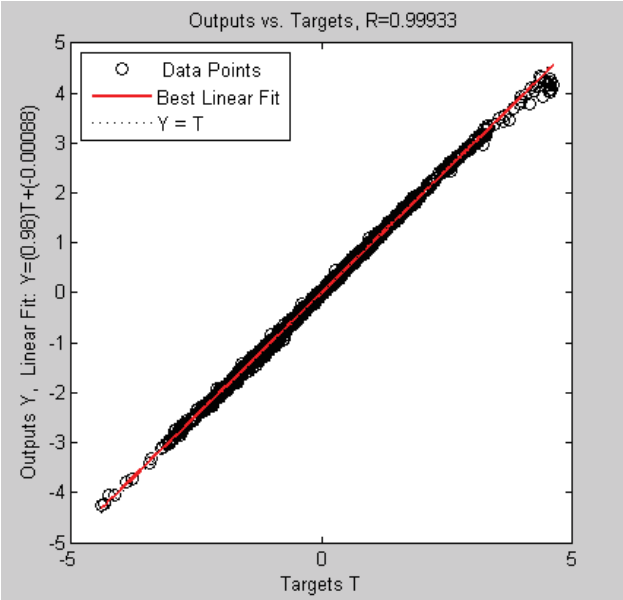


Figure 1.6.5. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°6.

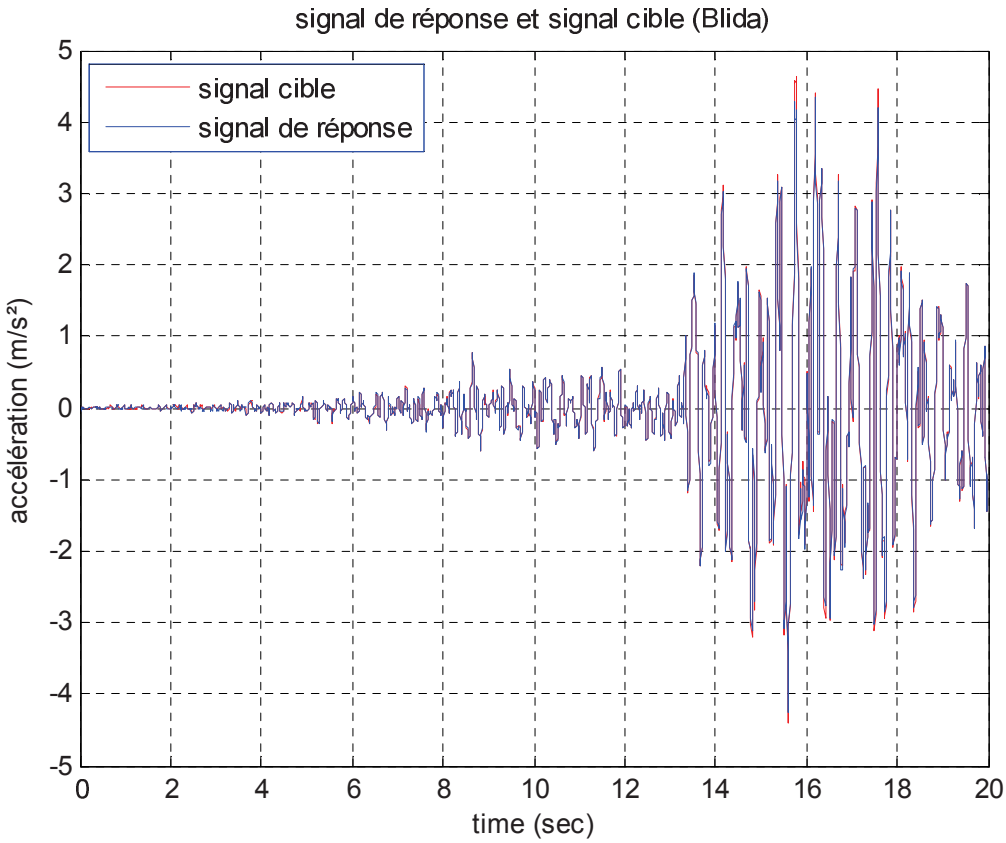


Figure 1.6.6. Différence entre le signal de réponse et le signal cible du modèle N°6.

1.6-4. Accélérogramme d'Hussein Dey : R=0.99

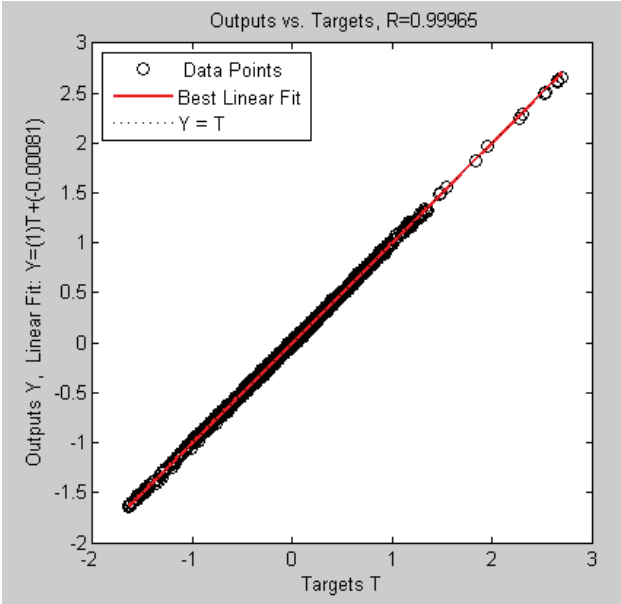


Figure 1.6.7. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°6.

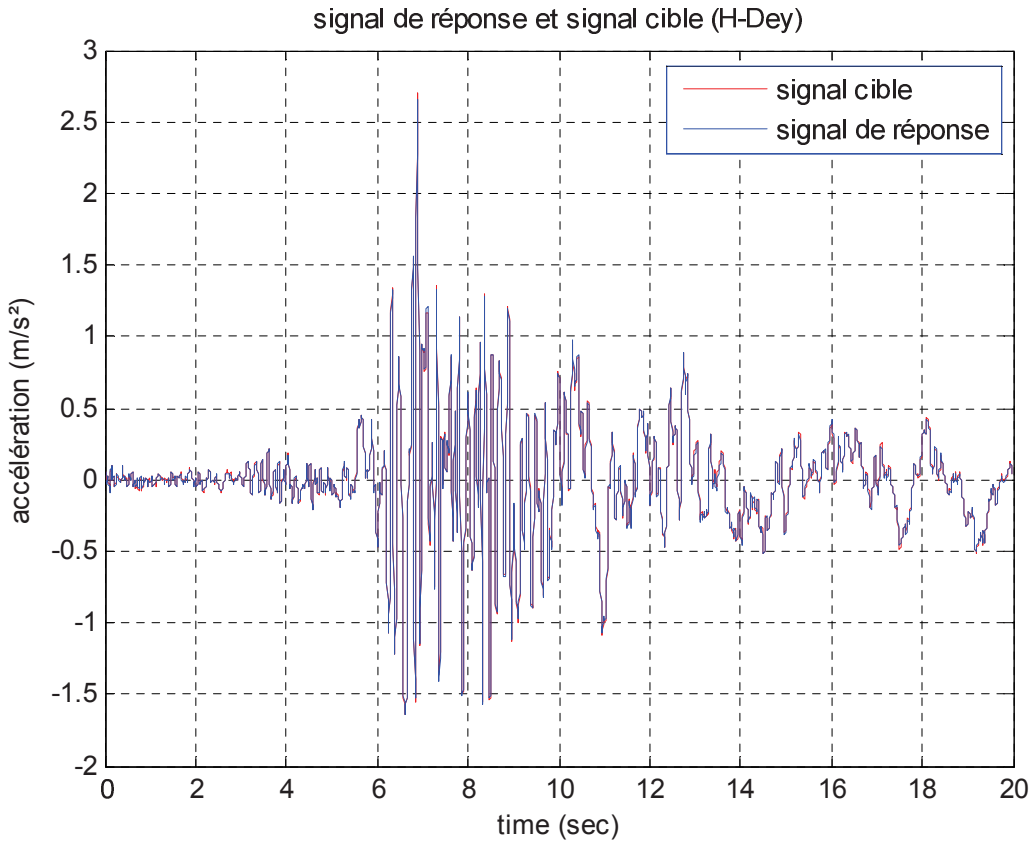


Figure 1.6.8. Différence entre le signal de réponse et le signal cible du modèle N°6.

1.6-5. Accélérogramme d'Azazga : R=0.99

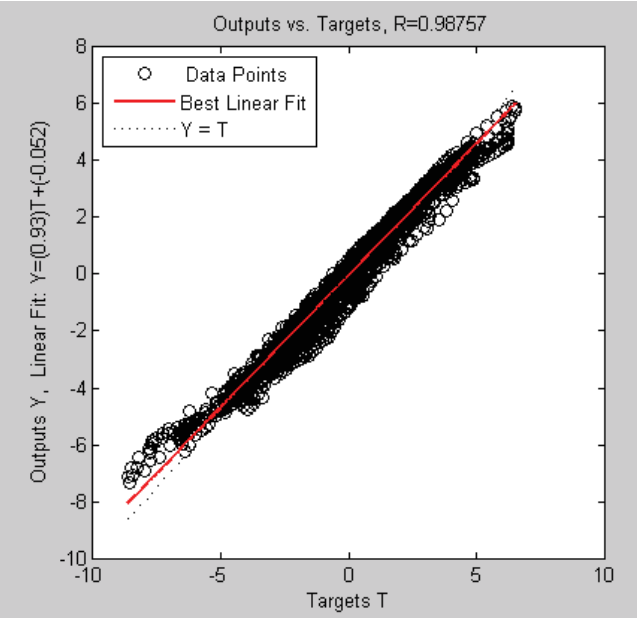


Figure 1.6.9. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°6.

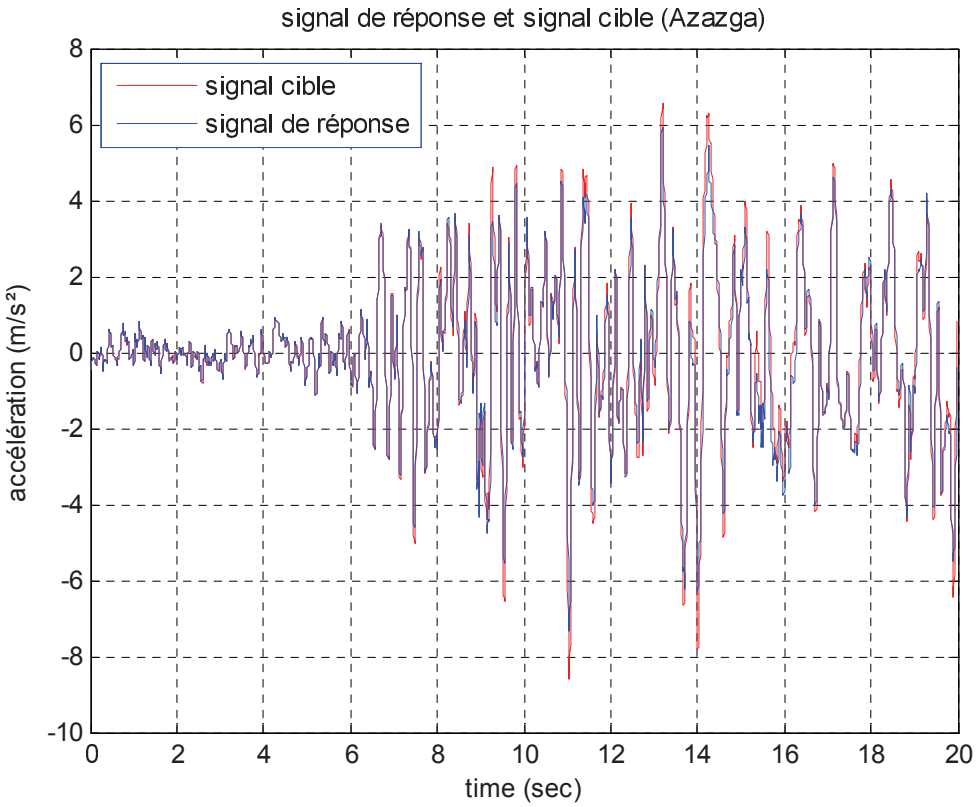


Figure 1.6.10. Différence entre le signal de réponse et le signal cible du modèle N°6.

1.6-6. Accélérogramme d'El Centro : R=0.99

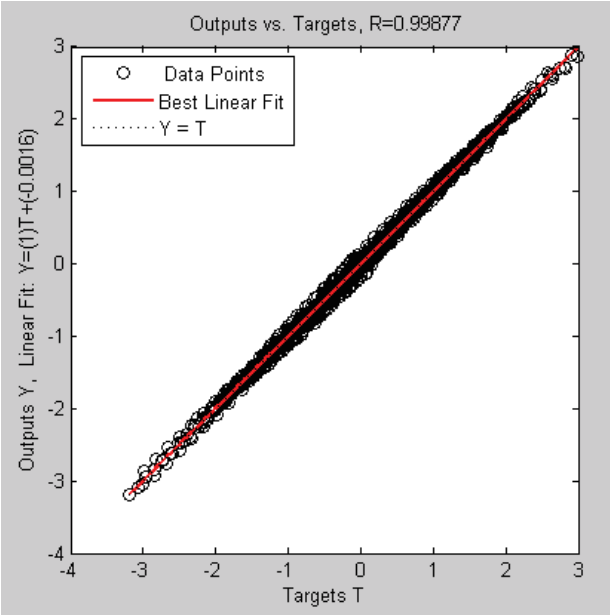


Figure 1.6.11. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°6.

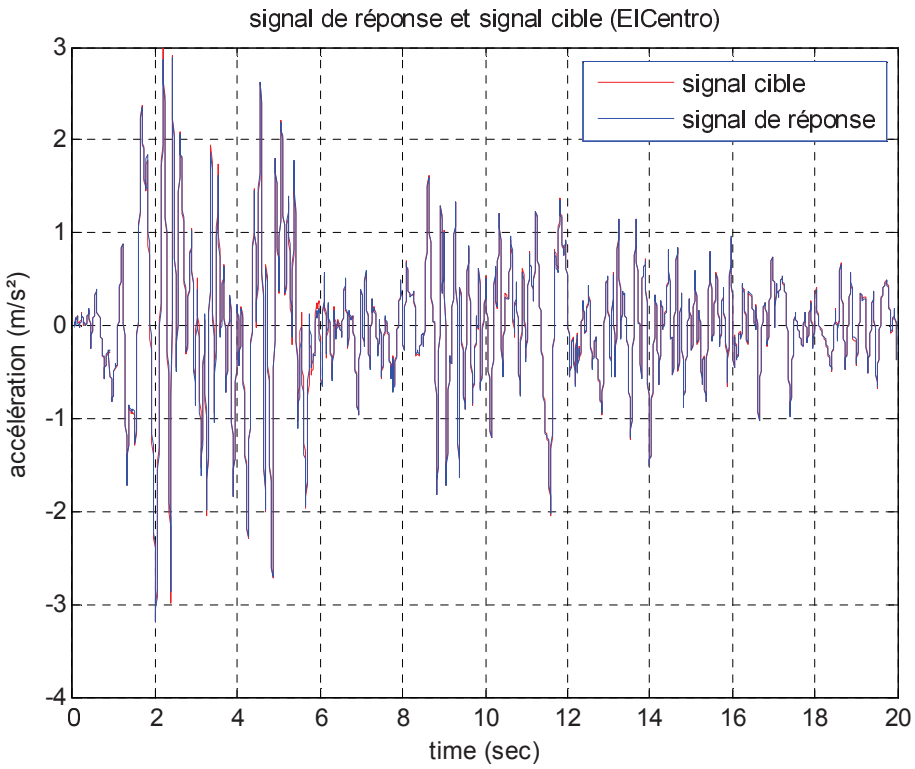


Figure 1.6.12. Différence entre le signal de réponse et le signal cible du modèle N°6.

1.6-7. Accélérogramme de Kobe : R=0.99

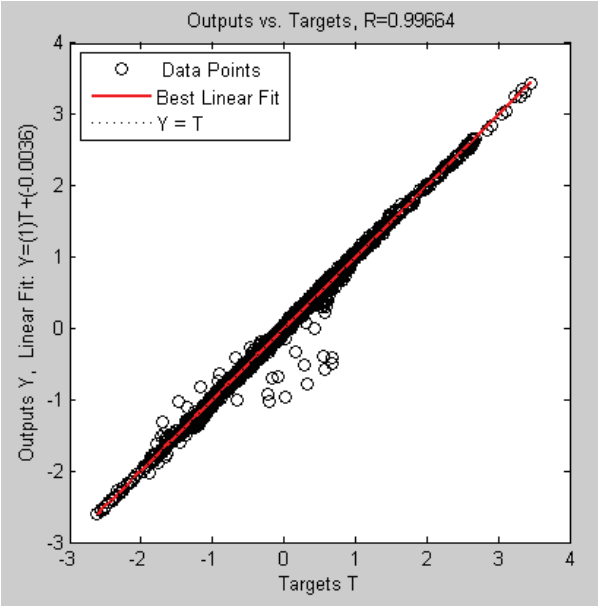


Figure 1.6.13. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°6.

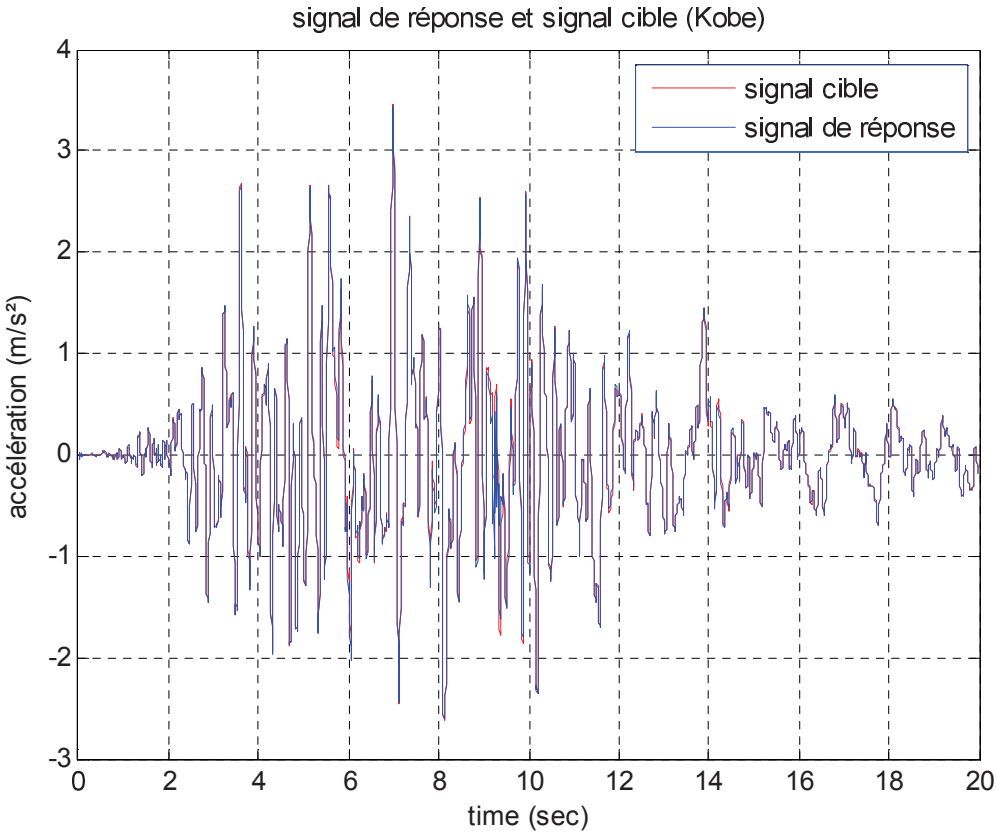


Figure 1.6.14. Différence entre le signal de réponse et le signal cible du modèle N°6.

1.6-8. Signal sinusoïdal : R=0.99

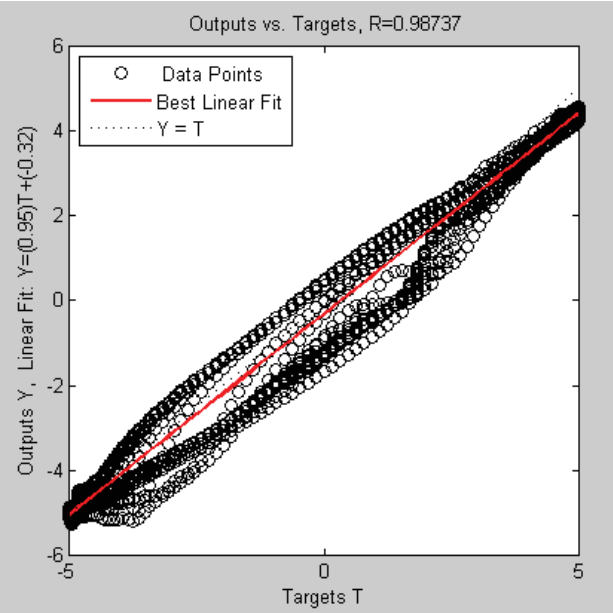


Figure 1.6.15. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°6.

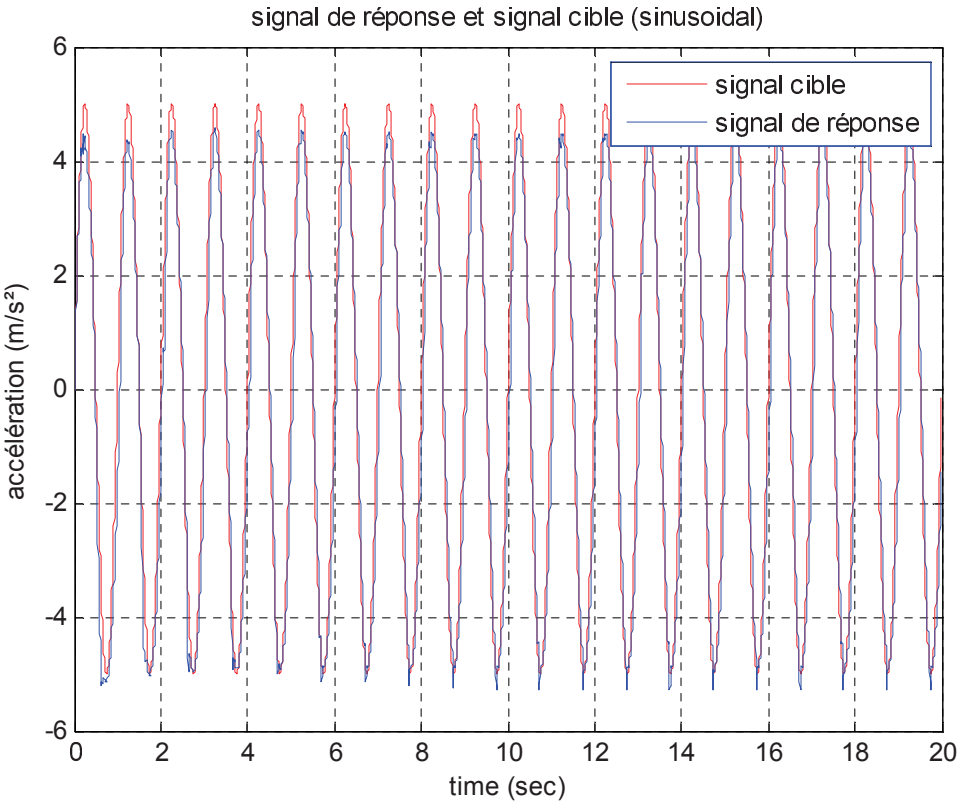


Figure 1.6.16. Différence entre le signal de réponse et le signal cible du modèle N°6.

2- Modèle de la table par fonctions de transfert :

2.1-Modèle N°1 : table non chargée

2.1-1. Accélérogramme d'El Centro : R=0.94

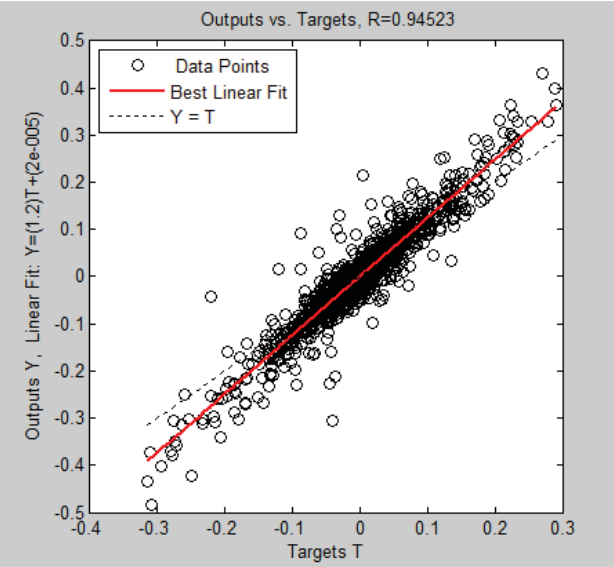


Figure 2.1.1. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°1.

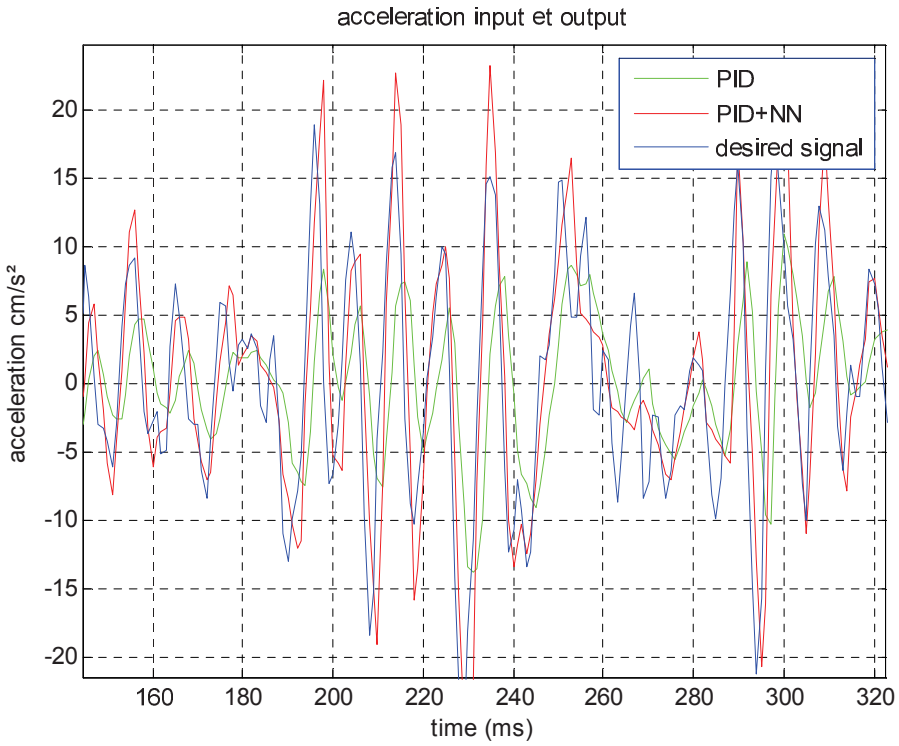


Figure 2.1.2. Différence entre le signal de réponse avec correction par PID, signal de réponse avec correction par PID+NN et le signal cible du modèle N°1 (El Centro)

2.1-2. Accélérogramme de Kobe : R=0.97

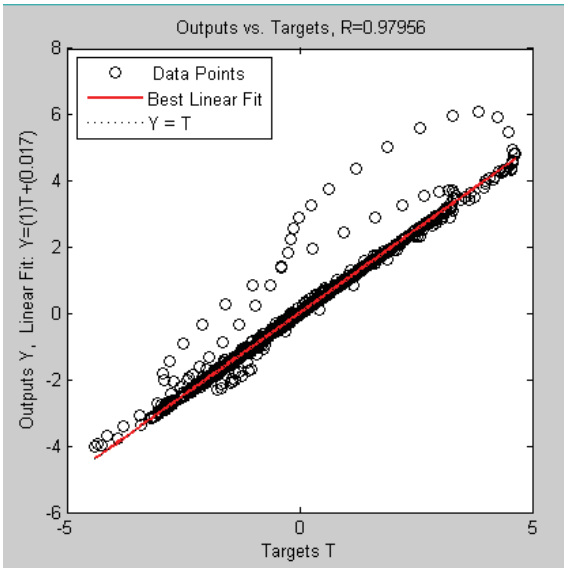


Figure 2.1.3. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°1.

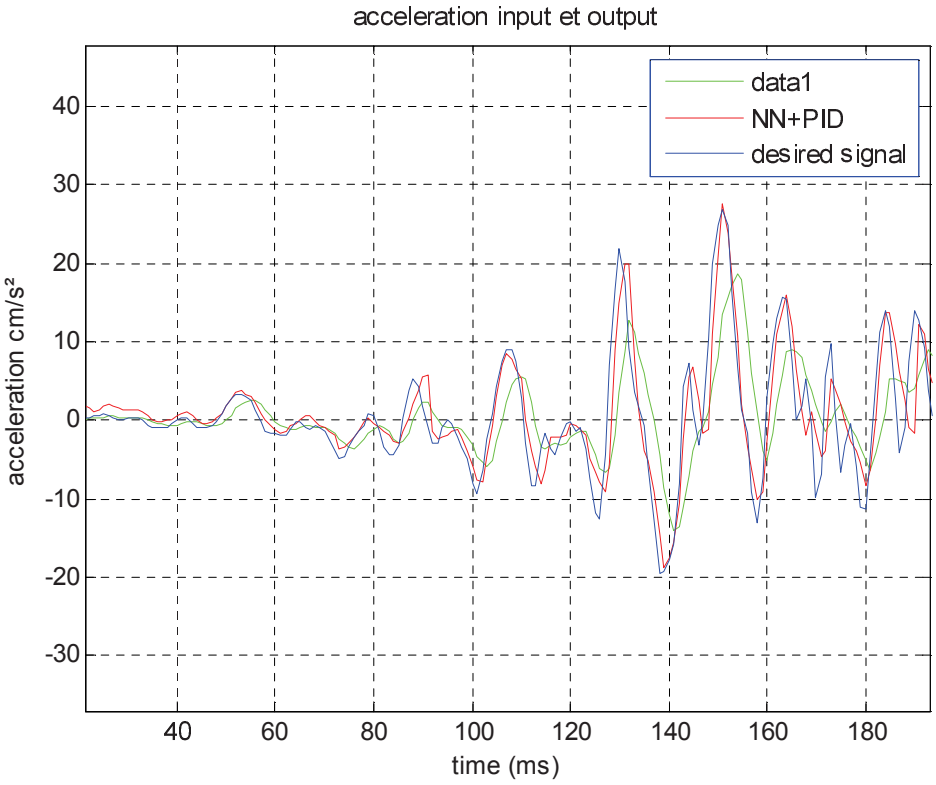


Figure 2.1.4. Différence entre le signal de réponse avec correction par PID, signal de réponse avec correction par PID+NN et le signal cible du modèle N°1 (Kobe).

2.1-3. Accélérogramme de Friuli : R=0.96

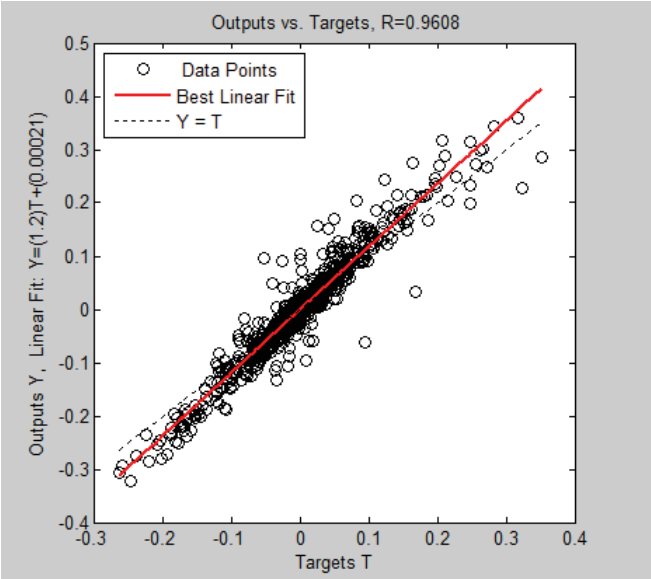


Figure 2.1.5. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°1.

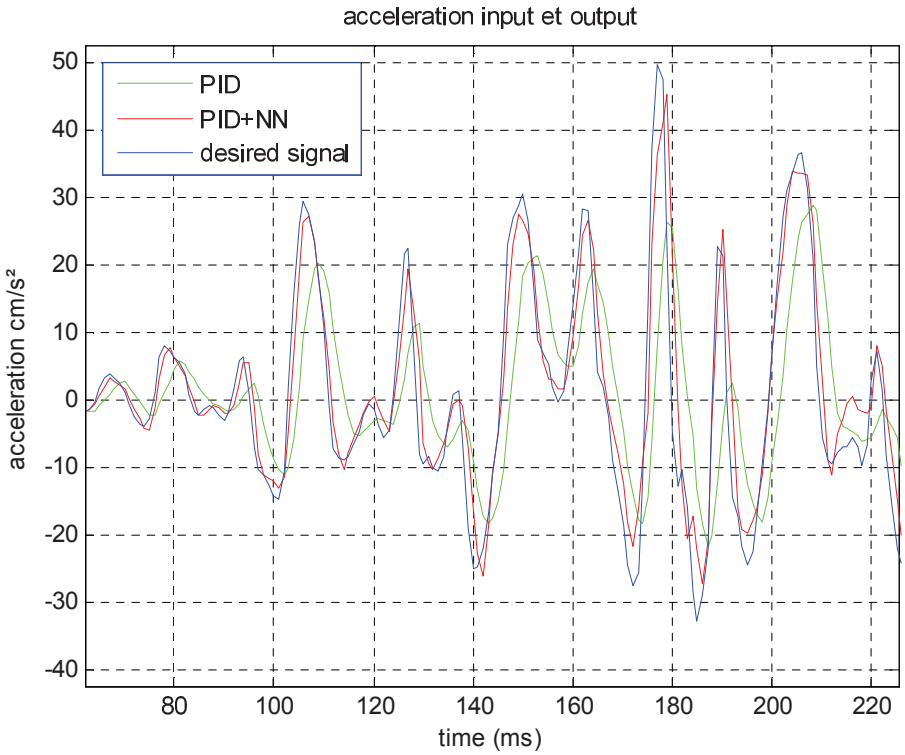


Figure 2.1.6. Différence entre le signal de réponse avec correction par PID, signal de réponse avec correction par PID+NN et le signal cible du modèle N°1 (Friuli).

2.2-Modèle N°2 : table chargée d'un spécimen à 1 DDL.

2.2-1. Accélérogramme d'El Centro : R=0.85

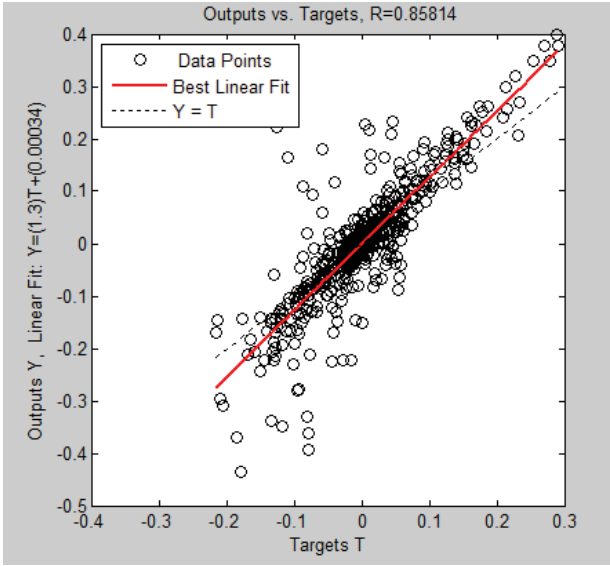


Figure 2.2.1. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°2.

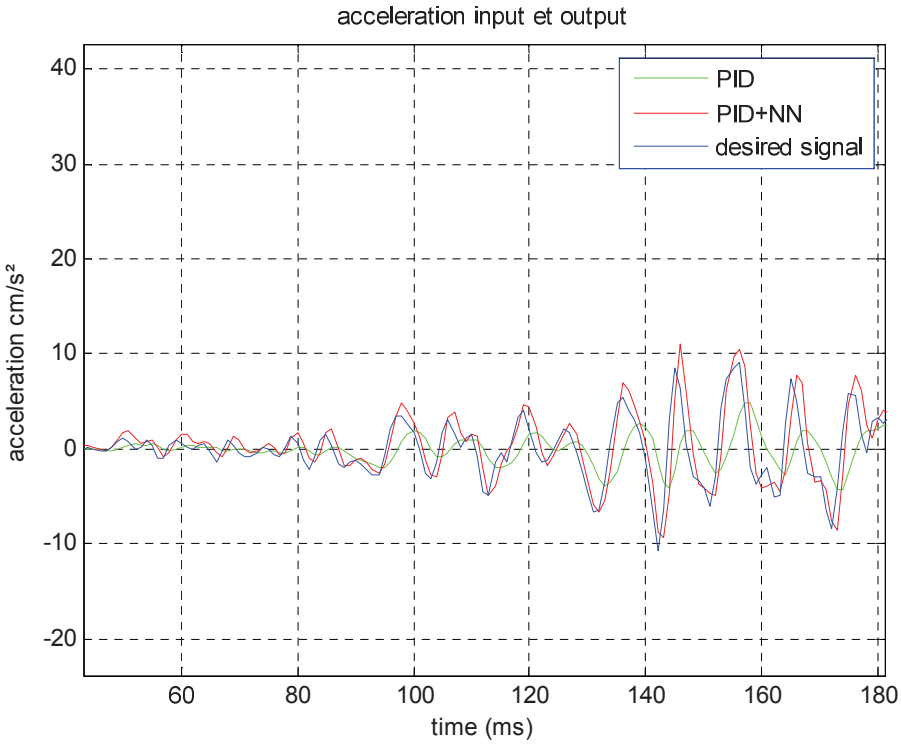


Figure 2.2.2. Différence entre le signal de réponse avec correction par PID, signal de réponse avec correction par PID+NN et le signal cible du modèle N°2 (El Centro).

2.2-2. Accélérogramme de Kobe : R=0.81

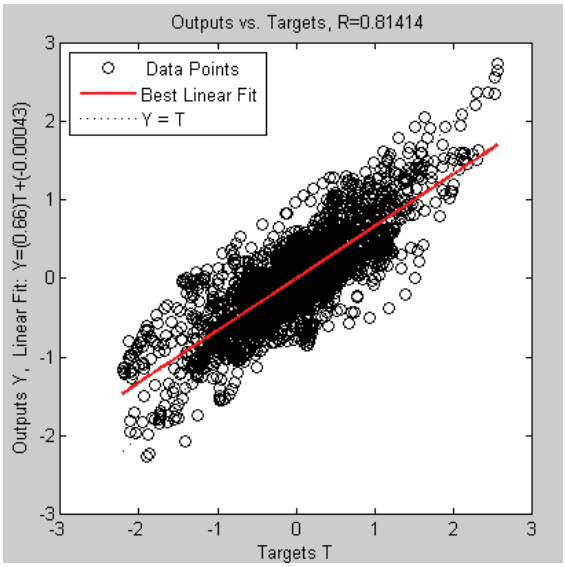


Figure 2.2.3. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°2.

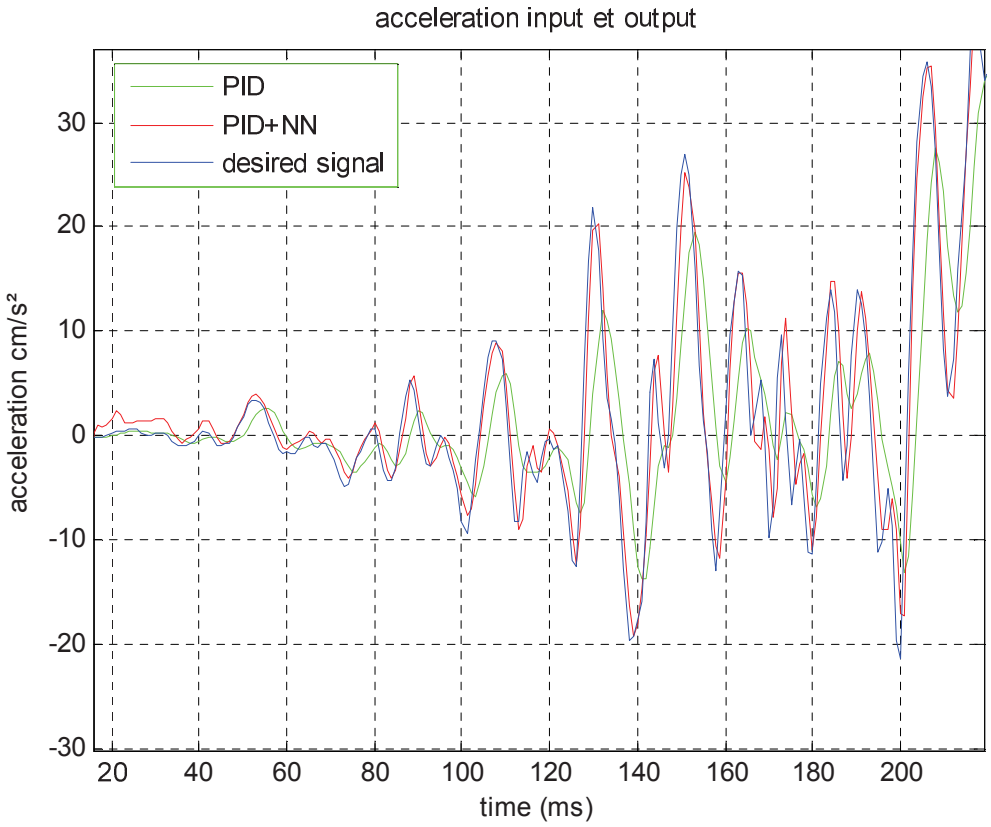


Figure 2.2.4. Différence entre le signal de réponse avec correction par PID, signal de réponse avec correction par PID+NN et le signal cible du modèle N°2 (Kobe).

2.2-3. Accélérogramme de Friuli : R=0.91

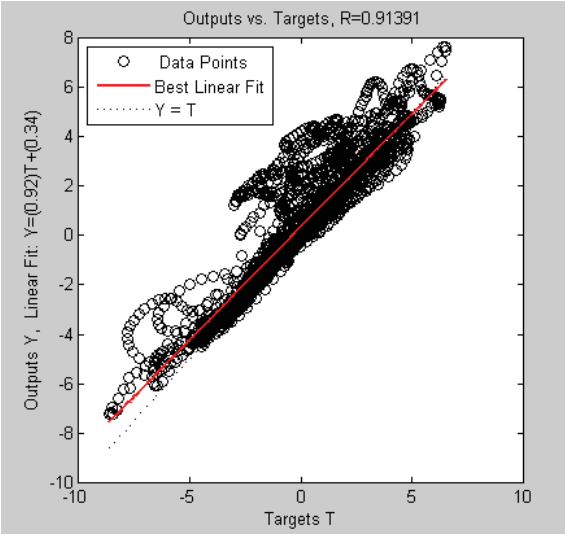


Figure 2.2.5. La régression linéaire entre les signaux cibles et les signaux de réponse du modèle N°2.

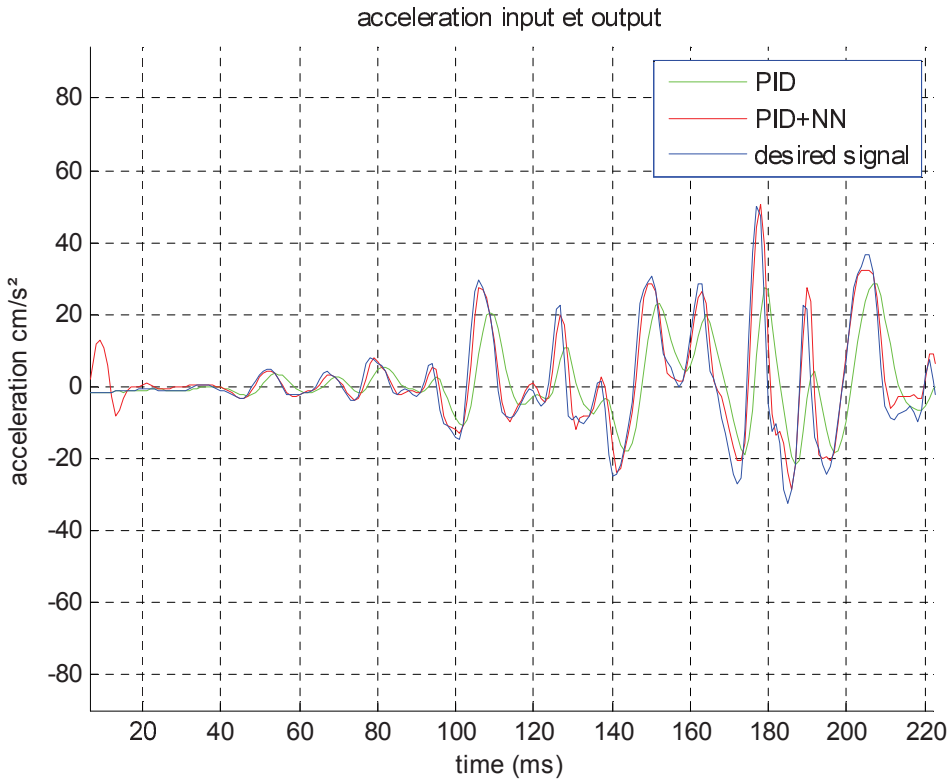


Figure 2.2.6. Différence entre le signal de réponse avec correction par PID, signal de réponse avec correction par PID+NN et le signal cible du modèle N°2 (Friuli).

ANNEXE 2 : Article paru dans IZIIS-50 Conference Proceedings, Mai 2015.

Offline matching of signals on a shaking table using Neural Networks

H. S. Larbi
National Advanced School of Public Works, 16000 Kouba, Algeria

N. Bourahla,
Civil Engineering Dept. University of Blida, 09000 Blida, Algeria

K. Choutri and H. Benchoubane
Aeronautic Dept. University of Blida, 09000 Blida, Algeria



ABSTRACT
In this study, an offline Neural Network (NN) controller is developed as an additional tool to actual shaking table control technique. First, the shaking table is simulated using a finite element model (FE). A second model is implemented as transfer functions in Matlab. Series of dynamic analysis are carried out to constitute a database to train and test the NN. The response at the center of gravity of the platform is used as the INPUT of the NN and the command signal applied at the fixed ends of the actuators (first model) or the drive signal (second model) is used as the OUTPUT. The proposed NN algorithm is validated using several acceleration signals. The correlation coefficient R between the network response and the target lines 99%. It indicates that the NN control method helps to reduce signal distortions induced by nonlinearities of the shaking table system or nonlinear specimens.

Key words: Shaking table, neural network, control, neural network control.

1. INTRODUCTION

Shaking tables are regarded as preeminent tools in dynamic testing to failure of specimens, especially when they generate the desired signal inputs with high fidelity. However, it is known that the reproduction of the commanded signal remains imperfect (Clark, 1992; Conte & Trombetti, 2000; Crewe & Severn, 2001; Ozcelik & al., 2008). Nowadays, one of the most significant challenge to the shaking table technique is the control of table motions to ensure target accelerations are reproduced (O'Hagan & Ma, 2012). Servo-hydraulic actuator limitations that cause time delays and time lags (Phillips & Spencer, 2011), strong dynamic cross coupling between degrees of freedom (Nowak & al., 2000), inherent dynamic characteristics of several sub-systems and their interactions (Trombetti & Conte, 2002), table-specimen interaction that cause significant disturbances to the system (Thoen & Laplace, 2004), are the main origins that affect significantly the performance of the shaking table controllers.
Several methods have been developed to improve the accuracy of time history matching and the stability of the shaking table control systems. Compensation methods (SK. Lee & al., 2007; Cheng & Ricles, 2009), adaptive control algorithms (Karshenas & al., 2000; Stoten & Gomez, 2001; Stoten & Shimizu, 2007), iterative control techniques (Spencer & Yang, 1998; Luco & al., 2008) and hybrid control strategies (Cheng & Ricles, 2008; Shen & al., 2013; Shen & al., 2011; Vaes & al., 2007) have been developed. Some studies show that the iterative process of adjusting control system parameters may pre-damage the specimen mounted on the platform and found that the offline control could be very useful (Thoen & Laplace, 2004).
In this study, the potential of an offline neural network (NN) controller is investigated. It is intended to predict the commanded signal which will be sent to the online controller to drive the platform to follow the desired trajectory.

2. METHODOLOGY

The neural network proposed in this study is intended to be used on offline mode to correct the original signal to obtain a drive signal which will be input to the command system in such a way that the response signal recorded on the center of gravity of the platen will be similar to the desired one.

The chart of the signal adjustment procedure is illustrated in figure 1.

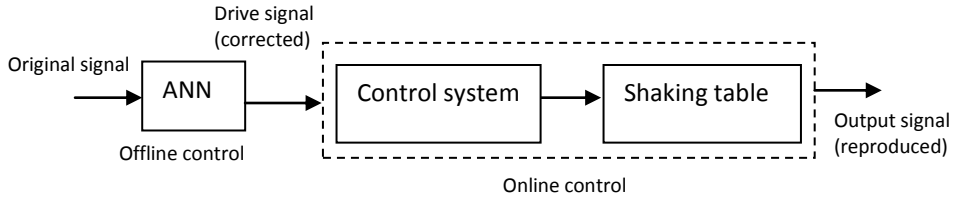


Figure 1. Chart of the signal adjustment procedure.

The NN is trained using database constituted from two models of the shaking table. The first model is a finite element model which represents the mechanical components of the shaking table (i.e: the platen, the actuators and the specimen) without any control system. The second model is implemented as transfer functions in Matlab and it represents a simplified scheme of an entire shaking table system (i.e: the online controller, servovalves, actuators and the platen). The NN input is the response signal at the center of gravity of the platen in displacement, velocity and acceleration the NN output is the command signal in acceleration applied at the fixed ends of the actuators for the first model or the drive signal for the second model.
After the training process, many signals are used for tests and validation of the NN. A linear regression between the network response and the target is performed and a correlation coefficient between the response and the target R is calculated to measure the performance of the NN.

3. THE FINITE ELEMENT MODEL OF THE SHAKING TABLE:

In the first stage, to simplify the shaking table model with dynamic characteristics as similar to the real ones as possible, two finite element models are developed.
The first one represents a bare table (without specimen) and the second one represents a loaded table (with specimen).
It is supposed here that the shaking table has no controller.
The first model includes:
- The platen: a 6.1mx6.1m shell element with an effective mass of 50tons and a natural frequency of 95Hz;
- The actuators: modeled by link elements with a rigidity of 60000KNm, a natural frequency of 16.4Hz (Airouche & al., 2014), a damping coefficient C of 40 KN(s/m) $^{1/2}$ and a nonlinear viscous damping with a damping coefficient α equal to 0.5 (Ozcelik & al., 2008).
The fundamental frequency of the model (bare table) is 5.81 HZ.
In the second model a nonlinear specimen is added with plastic link element at its base.
Two types of specimen are has been used: a rigid one with a natural frequency of 13Hz, and a flexible one with a natural frequency of 5 Hz. Hence, the fundamental frequency of the system (table + specimen) became equal to 2.21Hz and 1.52Hz respectively.

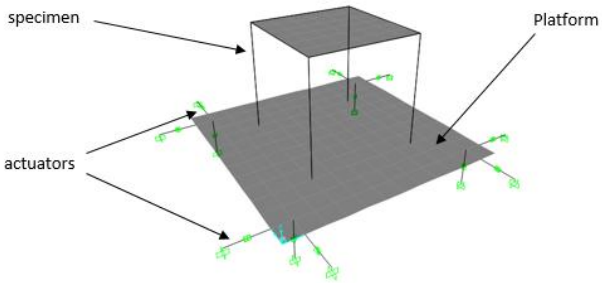


Figure 2. The FE model of the shaking table and a specimen

4. THE TRANSFER FUNCTION MODEL OF THE SHAKING TABLE

In order to model the behaviour of the shaking table with an online control system, another model of one directional shaking table system is developed including:
- Mechanical components: the reaction bloc, the platen and the specimen;
- Hydraulic components: servovalves and actuators;
- Electronic components: the controller.

Because this model will be used to build the data base of training, testing and validating the NN and for simplifying, the TVC controller is replaced by a PID controller. It includes all gains of control system: the proportional, integral, derivative, feed-forward and differential pressure gains (Conte & Trombetti, 2000).

The proposed model uses the total transfer function of the shaking table, obtained between the command signal and the response signal.

The total transfer function account for the following components and sub-systems as shown in figure3:
- $H_t(s)$: transfer function of the three-stage servo-valves
- S(s) : transfer function of the servo-actuator system
- H(s) : transfer function of the servo-hydraulic system (servovalve + actuator + controller)
- T(s) : transfer function of the shaking table system (servo-hydraulic system + platen + base)
- $T_{SDOF}(s)$: transfer function of the shaking table with a SDOF specimen (servo-hydraulic system + platen + specimen + base).

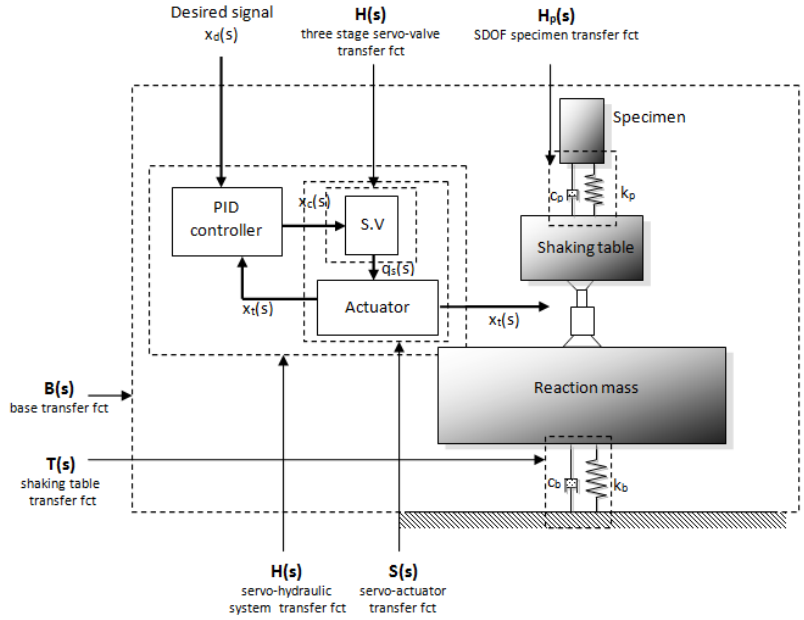


Figure 3: Schematic transfer functions of the system (Conte & Trombetti, 2000).

The total shaking table transfer function T(s) must incorporate the effects of the reaction bloc flexibilities representing by the base transfer function B(s). It is done in the following equation:

$$T(s) = \frac{xta\,(s)}{xd\,(s)} \tag{4.1}$$

$$T(s) = \mathrm{H'(s)[1 + B(s)]} \tag{4.2}$$

Where $x_{ta}(s)$ is the absolute displacement of the table, $x_d(s)$ is the desired absolute table displacement, H'(s) is the servo-hydraulic transfer function which also includes the effect of the reaction bloc flexibilities.

The transfer function of the SDOF specimen is given by:

$$Hp(s) = \frac{xpa\,(s)}{xta\,(s)} \tag{4.3}$$

$$Hp(s) = \frac{s\ 2\xi p\ \omega p + \omega p^2}{s^2 + s.2\xi p\ \omega p + \omega p^2} \tag{4.4}$$

Where $x_{pa}(s)$ is the absolute displacement of the specimen, ξ_p and ω_p are respectively the damping ratio and the natural circular frequency of the specimen.
The total loaded shaking table transfer function becomes as following:

$$Tsdof(s) = \frac{xta\,(s)}{xd\,(s)} \tag{4.5}$$

$$Tsdof(s) = H'sdof(s)[1 + Bsdof(s)] \tag{4.6}$$

Where $B_{SDOF}(s)$ and $H'_{SDOF}(s)$ are the base and the servo-hydraulic transfer functions respectively, both incorporate the effect of the reaction bloc flexibilities.

The expression of B(s), H'(s), $B_{SDOF}(s)$ and $H'_{SDOF}(s)$ could be found in (Conte & Trombetti, 2000).

5. NN ARCHITECTURE

ANNs have been extensively used in complex control processes and nonlinear systems (Mahajerin, 2012). Depending on the data flow, ANN can be divided into two categories: feed-forward neural network (FFNN) which are static networks and Recurrent Neural Network (RNN) which are dynamic networks. The latter have memory, so they can be trained to learn sequential or time-varying patterns as well as nonlinear response. These features of RNNs render them suitable tools in control systems such as the present application.

In this study, the NN chosen is the Layer-Recurrent Network (LRN) which is a new version of the Elman network with an arbitrary number of layers and an arbitrary transfer function is each layer. The training algorithm used is Lavenberg-Marquardt algorithm. The number of nodes and hidden layers are determined by experiments. It has been found that 21 neurons for one hidden layer were sufficient for the network to match the input and target signals (Figure 4).

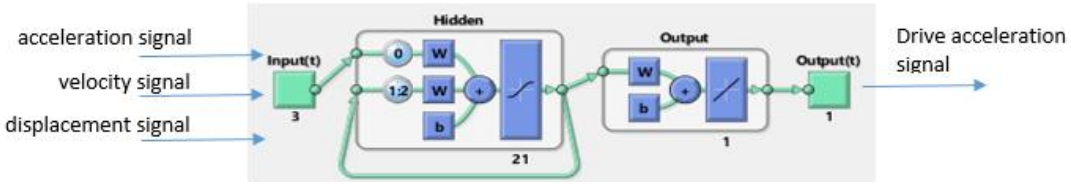
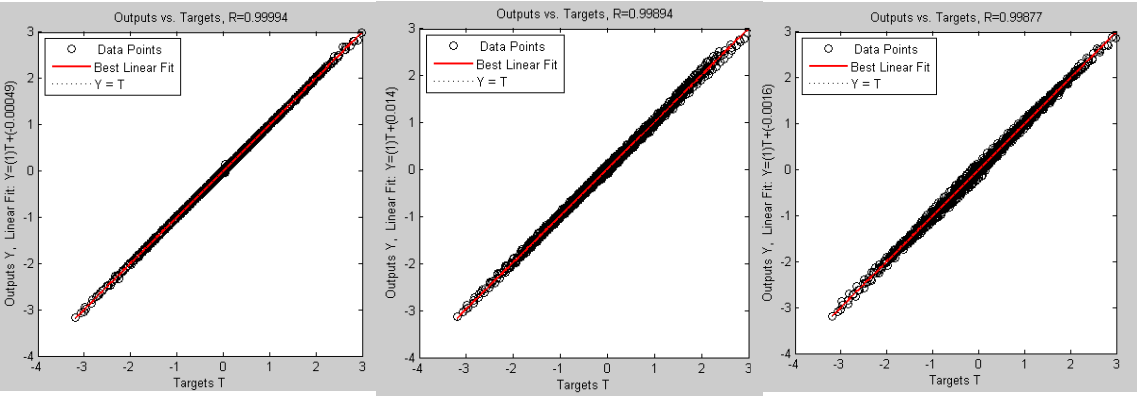


Figure4. Structure of the recurrent neural network (RNN)

6. EVALUATION OF THE NN CONTROL PERFORMANCE FOR A SHAKING TABLE WITHOUT ONLINE CONTROLLER

The database to train the NN is constituted using the nonlinear dynamic responses of the FE models of the shaking table. The signal at the center of gravity of the platen in displacement, velocity and acceleration represents the NN 3-vectors input. The signal at the fixed ends of the actuator in acceleration represents the NN target.
The network is trained with a real earthquake record (Meliana station, Boumerdes earthquake in Algeria, 2003). This signal contains 4000points which were found to be enough for a good training of the NN.

The error in prediction of the target signal is estimated using the mean square error formula (MSE). The error is set equal to 10-5 and the models converged in all cases (bare table, table with a rigid specimen, table with a flexible specimen).
Then, the network is tested by comparing the output with the target signal using El Centro earthquake record.
To validate the network, several signals are used: five records from five stations representing near and far field records to Boumerdes earthquake (Algeria 2003), in addition to Kobe earthquake record and a sinusoidal signal.
The performance of the NN is evaluated using a linear regression between the network response and the target where a correlation coefficient between the response and the target R is calculated.
The fitting lines shown in figure 5 (a), (b) and (c) are practically superposed with the diagonal and the correlation coefficient obtained is very close to unity, which means that the NN gives very accurate prediction of the drive signals for models representing the table without specimen, the table with a rigid specimen and the table with a flexible specimen respectively.



(a) (b) (c)

Figure 5. Linear regression between the NN response and the target for (a) table without specimen, (b) table with rigid specimen and (c) table with flexible specimen

Figure 6 shows a quasi-perfect matching between the response of the platform and the desired signal (El Centro earthquake record). It indicates that for a bare table, a loaded table with rigid specimen, a loaded table with a flexible specimen, the NN can predict quasi-perfectly the signal that will produce the desired signal at the platform of the shaking table. This is also valid for nonlinear actuators and nonlinear specimen.

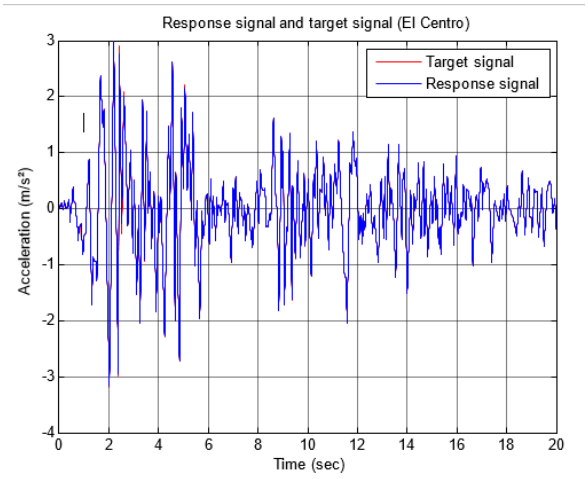
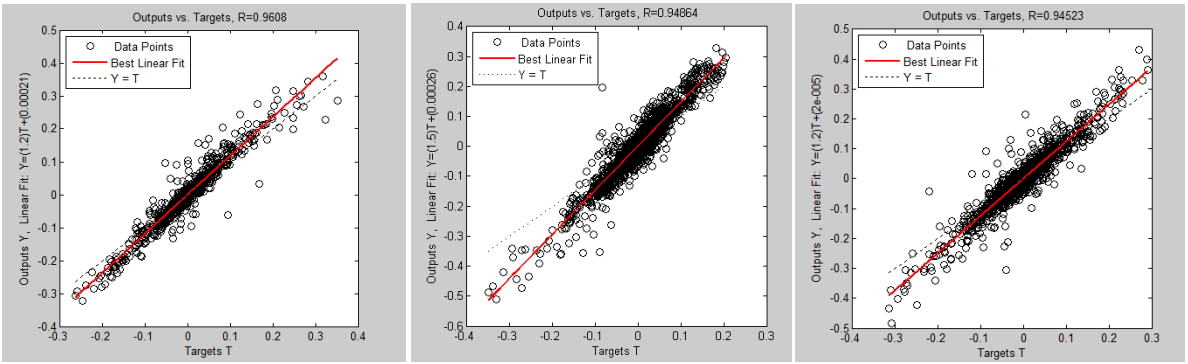


Figure 6. Platform response and target signals.

7. PERFORMANCE OF THE NN CONTROL CORRECTION OF THE SHAKING TABLE WITH ONLINE PID CONTROLLER

The model shown in Figure 3 is used to build the database to train the NN. Similarly to the FE model, the response of the transfer function of the system in displacement, velocity and acceleration are used as input to the NN. The desired signal in acceleration represents the NN target.
Several trials with real earthquake records containing up to 4000 points were not be enough for a good training of the NN. Therefore an artificial earthquake signal of 200 second containing 20000 points has been generated by Seismoartif software (Seismosoft 2013) and used for the training of the NN. The best MSE error in predicting the target signal reached 10^{-3}. Several earthquake record signals were used to test and validate the NN. Figure 7 shows linear regression functions for some of the testing signals, the best R value reached 0.96.



(a) **(b)** **(c)**

Figure 7. Linear regression between the NN response and the target for (a) Friuli earthquake record, (b) Kocaeli earthquake record and (c) El-Centro earthquake record

In terms of signal matching, the NN was not as successful as the shaking table model without online controller. The difficulties for the NN in establishing the relationship between the input and the target signals is attributed to the PID controller which have a variation very dependent on the type of signal. In general, the offline correction using the NN in conjunction with the PID, helps to correct both the delay and amplitudes at low vibrations but may not very efficient to match the peaks at the strongest parts of the signals. Figure 8 compares the output of the NN with the El-Centro earthquake record (target), using as input the computed signal by the transfer function $T_{SDOF}(s)$. Figures 9(b) and 9(c) show the desired signal together with the table responses using PID only control and PID+NN control.

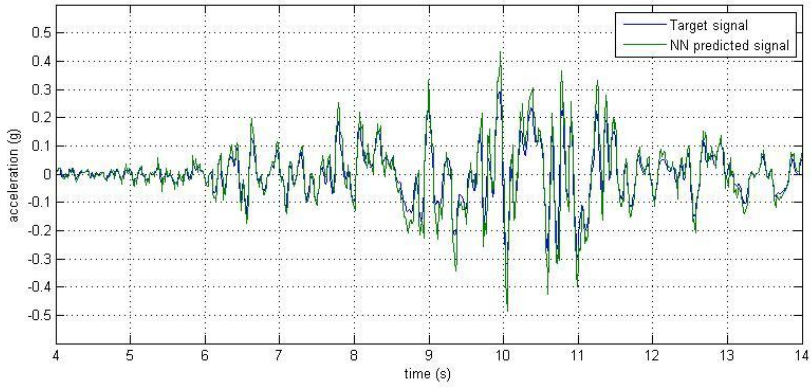


Figure 8. Target and NN predicted signal of El-Centro earthquake record

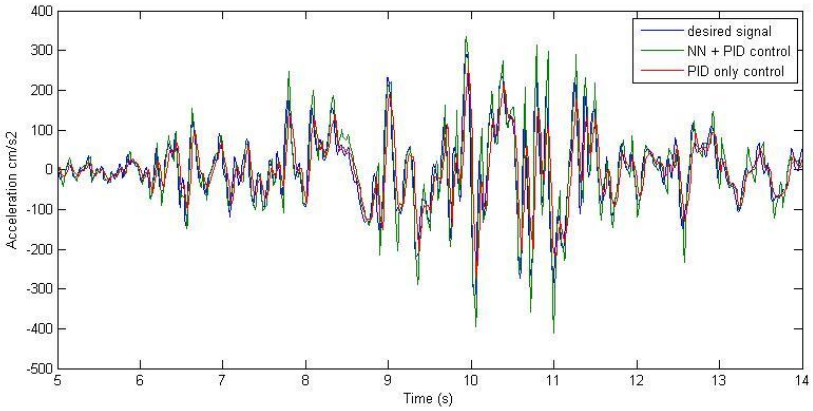


Figure 9. Desired signal, PID only controlled and PID+NN controlled responses of the shaking table under El-Centro earthquake record

8. CONCLUSION

In this paper, a neural network has been developed as an offline controller to reduce the signal distortion between the simulated and the desired signal. In first stage, the NN controller has been applied to a table without any online controller. For this purpose, a finite element model of the shaking table has been used to build the database to train the network. For a bare and loaded table, the NN has been able to predict commanded signals which drive the platform to follow the desired motion with high level of accuracy even for nonlinear actuator and nonlinear rigid or flexible specimens. In the second stage, the NN controller has been applied to a table and a specimen with an online PID controller. The commanded signals predicted by the NN were successful in reducing delay and amplitude distortions at low amplitudes. It should be noted however, that the matching process is very dependent on the type of signal and the peaks during the strongest part of the motion could be exceeded. In general the offline NN control showed a good potential to generate commended signal that help to match target signals, but need to be validated experimentally which will be the future work of this study.

REFERENCES

Airouche A.H., Bechtoula H., Aknouche H., Thoen B.K. and Benouar D., (2014). Experimental identification of the six DOF CGS, Algeria shaking table system. *Smart Structures and Systems*, **13:1**, 137-154.

Cheng C. and Ricles J.M., (2008). Analysis of actuator delay compensation methods for real-time testing. *Engineering Structures*, **31:11**, 2643-2655.

Cheng C. and Ricles J.M., (2009). Servo-hydraulic actuator control for real-time hybrid simulation. *Proceedings, 2009, American Control Conference, St. Louis, Missouri.*

Clark A.J., (1992). Dynamic characteristics of large multiple degrees of freedom shaking tables. *Proceeding of the Tenth World Conference Earthquake Engineering, Balkema, Rotterdam.*

Conte J.P. and Trombetti T.L, (2000). Linear dynamic modeling of a uniaxial servo-hydraulic shaking table system. *Earthquake Engineering and Structural Dynamics*, **29**, 1375-1404.

Crewe A.J. and Severn R.T., (2001). The European collaborative program on evaluating the performance of shaking tables. *Philosophical Transactions of the Royal Society London A.*, **359**, 1671-1696.

Karshenas A.M., Dunnigan M.W. and Williams B.W., (2000). Adaptive inverse control algorithm for shock testing. *IEE Control Theory and Application*, **147:3**, 267-276.

Lee S-K., Park E.C., Min K-W., Lee S-H., Chung L., Park J-H., (2007). Real-time hybrid shaking table testing method for the performance evaluation of the tuned liquid damping controlling seismic response of building structures. *Journal of Sound and Vibration*, **302:3**, 596-612.

Luco J.E., Ozcelik O. and Conte J.P. (2010). Acceleration tracking performance of the UCSD-NEES shake table. *Journal of Structural Engineering*, **136:5**, 481-490.

Mahajerin Nima, (2012). Identification and predictive control using Recurrent Neural Networks. *international master's thesis, university of Orebro, Sweden.*

Neural Network Toolbox User's Guide: For Use with MATLAB (1984-2011), http://www.mathworks.com.
Nowak R.F., Kusner D.A., Larson L.R. and Thoen B.K., (2000). Utilizing modern digital signal processing for improvement of large scale shaking table performance. *Proceeding of The 12th World Conference on Earthquake Engineering.*
O'Hagan J.T and Ma Q.T., (2012). Experimental assessment of PID control for a uniaxial shake table. *Proceeding of The 15th World Conference on Earthquake Engineering, 2012, Lisboa.*
Ozcelik O., Luco J.E. and Conte J.P., (2008). Identification of the mechanical subsystem of the NEES-UCSD shake table by a least-squares approach. *Journal of Engineering Mechanics,* **134:1**, 1-23–34.
Phillips B.M. and Spencer B.F Jr., (2011). Feedforward-feedback tracking control for real-time hybrid simulation. *Proceeding The 6th International Workshop on Advanced Smart Materials and Smart Structures Technology 2011, Dalian, China.*
Seismosoft (2013) "SeismoArtif v2.1 – A computer program for generating artificial earthquake accelerograms matched to a specific target response spectrum," available from http://www.seismosoft.com.
Shen G., Zheng S.T., Ye Z.M., Huang Q.T. Yang Z.D., Zhao. Y. and al., (2011). Tracking control of an electro-hydraulic shaking table system using a combined feed-forward inverse model and adaptive inverse control for real-time testing. *The Institution of Mechanical Engineers, Part I, Journal of Systems and Control Engineering,* **255:5**, 647-666.
Shen G., Zhu Z-C., Zhang L., Tang Y., Yang C-F., Zhao J-S., Liu G-D., Han J-W.(2013). Adaptive feed-forward compensation for hybrid control with acceleration time waveform replication on electro-hydraulic shaking table. *Control Engineering Practice*, **21**, 1128-1142.
Spencer B.F. and G. Yang, (1998), Earthquake simulator control by transfer function iteration. *Proceeding of The 12th ASCE Engineering Mechanics Conference 1998, San Diego.*
Stoten D.P. and Gomez E.G., (2001). Adaptive control of shaking tables using the minimal controller synthesis algorithm. *Philosophical transactions of the Royal Society.A,* **359:1786**, 1697-1723.
Stoten D.P. and Shimizu N., (2007). The feed-forward minimal control synthesis algorithm and its application to the control of shaking tables. *Journal of Systems and Control Engineering,* **221:3**, 423-444.
Thoen B.K. and Laplace P.N., (2004). Offline tuning of shaking tables. *Proceeding of The 13th World Conference on Earthquake Engineering, Vancouver B.C., Canada.* Paper No.960.
Vaes D., Engelen K. and Anthonis J., (2007). Multivariable feedback design to improve tracking performance on tractor vibration test rig. *Mechanical Systems and Signal Processing,* **21:2**, 1051-1075.