



16^{èmes} Journées de l'Hydrodynamique

27-29 novembre 2018 - Marseille



APPLICATION DES RESEAUX DE NEURONES AU PROBLEME DE TENUE A LA MER HYDRO-ELASTIQUE

NEURAL NETWORKS APPLIED TO THE HYDRO-ELASTIC SEAKEEPING PROBLEM

C. MONROY⁽¹⁾, A. ANDONIU⁽¹⁾, A. BENHAMOU⁽¹⁾

charles.monroy@bureauveritas.com ; alexandru.andoniu@bureauveritas.com ; alexis.benhamou@bureauveritas.com

⁽¹⁾ Département recherche, Bureau Veritas, Paris La Défense

Résumé

Nous explorons dans cet article la possibilité d'utiliser un réseau de neurones pour prédire de façon déterministe le moment fléchissant d'un navire dans un état de mer irrégulier. Pour plusieurs états de mer, le comportement hydro-élastique du navire est évalué grâce à un logiciel d'hydro-structure. Grâce à cette base d'apprentissage, le réseau de neurones est capable de faire des prédictions pour des conditions de houle non rencontrées auparavant. Le potentiel intérêt de cette méthode est de pouvoir analyser la réponse sur plein d'états de mer en un temps relativement court.

Summary

We investigate in this article the possibility to use a neural network in order to predict deterministically the vertical bending moment of a ship in irregular waves. For a few sea states, the hydro-elastic behaviour of the ship is assessed with a non-linear hydro-structural tool. Based on this dataset, the network is able to make predictions for random wave conditions. If proved successful, it could considerably fasten the design process as the neural network is much faster than the direct computation for new sea states.

I – Introduction

I – 1 Problématique

Récemment, les algorithmes d’intelligence artificielle et plus précisément d’apprentissage profond ont enregistré des succès notables dans les domaines allant de la reconnaissance d’image au traitement automatisé du langage. Les bases théoriques de ces méthodes sont connues depuis longtemps et leur soudaine popularité est due à plusieurs facteurs concomitants : augmentation de la puissance de calcul, maturité et facilité d’utilisation des bibliothèques d’intelligence artificielle (disponibles en *open source*) et présence de grandes bases de données d’apprentissage.

Dans cet article, nous nous intéressons à la possibilité d’appliquer des réseaux de neurones à la prédiction de la tenue à la mer hydro-élastique. Ceci est fait grâce à la bibliothèque Python *open source* Keras [7] permettant d’interagir avec les algorithmes de neurones profonds disponibles dans TensorFlow [8], un outil *open source* d’apprentissage automatique. Dans le cadre de notre étude, les bases de données d’apprentissage sont générées par le biais de simulations numériques préalables. Le modèle numérique utilisé doit être à la fois suffisamment rapide pour pouvoir générer de longues séries temporelles, mais aussi suffisamment lent pour que l’utilisation d’un réseau de neurones pour la prédiction reste intéressante. Le cas d’application, qui constitue un compromis raisonnable, consiste en une base de données de calculs hydro-structure non-linéaires (effectués par le logiciel HOMER) sur plusieurs états de mer. Si le réseau de neurones permet de prédire le comportement hydro-élastique du navire sur n’importe quel état de mer à partir d’un entraînement sur un petit nombre d’états de mer, ceci pourrait considérablement accélérer la méthodologie d’analyse statistique de la réponse long-terme du navire.

Au-delà de ce problème précis, la volonté d’expérimenter avec les réseaux de neurones est aussi née de l’importance croissante qu’ils semblent prendre dans notre discipline. À titre d’exemple, lors de la dernière conférence OMAE 2017 (*36th International Conference on Ocean, Offshore and Arctic Engineering*), une dizaine de papiers traitait de l’utilisation des réseaux de neurones dans le contexte offshore. En tant que société de classification, Bureau Veritas se doit de monter en compétence sur des méthodes qui pourraient devenir standard dans un futur proche. Dans notre domaine, une des sources d’inspiration pour cet article est l’utilisation d’un réseau de neurones pour prédire les tensions dans les lignes d’ancrage [2], mais à la différence du cas qui nous intéresse ici, le modèle dispose de la connaissance de l’état du système au pas de temps précédent.

I – 2 Les réseaux de neurones, qu’est-ce que c’est ?

Il est hors de propos d’expliquer en détail ici le fonctionnement des réseaux de neurones. Nous pouvons conseiller quelques documents qui nous ont permis, sans connaissances préalables, d’acquérir une compréhension pratique suffisante du sujet pour l’étude qui suit. Tout d’abord, les premiers cours de Yann Le Cun dans sa chaire consacrée à l’intelligence artificielle offrent une introduction intéressante en vidéo [10]. Les tutoriels pour expliquer l’utilisation de Keras sont assez nombreux, mais nous pouvons recommander [9] pour une prise en main rapide. Enfin, l’ouvrage sur l’apprentissage profond rédigé par le créateur de Keras [1] est très bien fait pour un ingénieur venant d’une autre discipline.

Cependant, voici une courte introduction au sujet pour permettre de comprendre le reste de l’article. La Figure 1 est une représentation schématique d’un perceptron (ou neurone) qui est l’unité de base d’un réseau de neurones. Un perceptron reçoit un vecteur d’inputs $[z_1, z_2, \dots, z_n]$. À chaque input z_i est associé un poids v_i . À la somme pondérée $net = \sum z_i v_i$ est ôté un biais constant θ choisi aléatoirement pour chaque neurone. Le perceptron prend en entrée cette différence $net - \theta$ et lui applique une fonction d’activation qui renvoie un nombre réel. Il existe plusieurs fonctions d’activation, mais l’une des plus populaires, qui a principalement été utilisée dans cette étude, est la

fonction RELU (pour *Rectified Linear Unit*) qui se définit simplement par l'équation suivante $f(x) = \max(0, x)$.

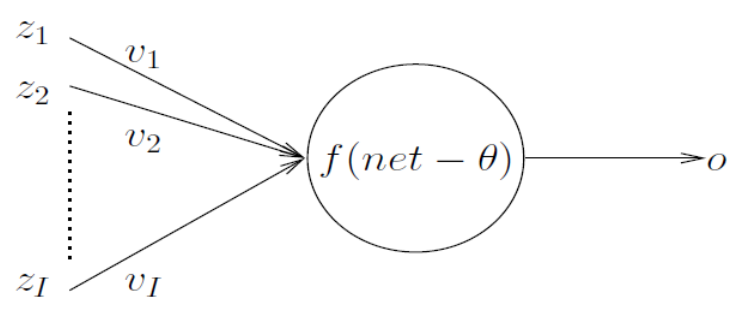


Figure 1 – Perceptron. Image issue de [6].

La Figure 2 est la représentation schématique d'un perceptron multi-couches (ou plus exactement d'un réseau de perceptrons multi-couches) qui est le type de réseau de neurones qui a été utilisé dans le cadre de cette étude. Le vecteur d'inputs mis en entrée du réseau de neurones consiste en la liste d'attributs, tandis que le réseau renvoie une unique valeur cible en sortie. Ce type de réseau est utilisé pour le problème de régression qui nous intéresse ici. Lorsqu'il y a plusieurs couches dans le réseau (avec des « couches cachées » - « *hidden layers* »), il est commun de parler d'apprentissage profond (« *deep learning* »).

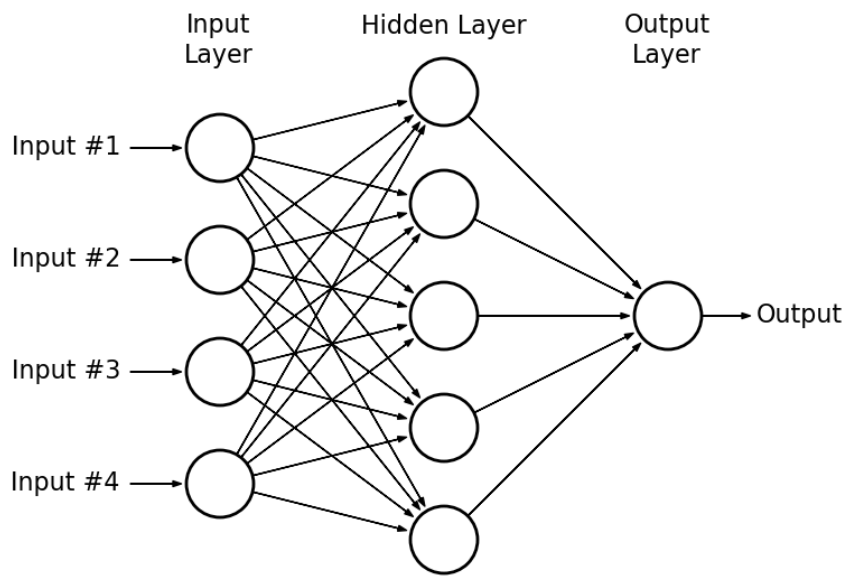


Figure 2 - Représentation schématique d'un réseau de neurones. Image issue de [6].

Pour entraîner un réseau de neurones, il faut disposer d'une grande quantité de vecteurs d'attributs et de valeur cible associée. Pour estimer la qualité de sa prédiction et mettre à jour les poids associés à chaque neurone, le réseau doit disposer d'une fonction de coût, qui pour chaque lot d'attributs et de valeurs cibles renvoie l'erreur associée à la nouvelle prédiction par rapport à la valeur cible connue. Mathématiquement, la seule partie complexe du processus intervient dans la mise à jour des différents poids à chaque itération (à chaque fois que de nouveaux couples attributs-cible sont fournis au réseau). Un algorithme de descente de gradient (tirant profit du fait que les fonctions d'activation sont connues et différentiables) permet de trouver les valeurs des paramètres qui minimisent la fonction de coût. Ainsi, chaque nouvel échantillon présenté fait varier légèrement les poids associés aux inputs de chaque neurone.

Une fois que tous les échantillons (couples de vecteur d'attributs et de valeur cible associée) sont passés une fois dans le réseau de neurones, il est possible de les refaire passer plusieurs fois à nouveau pour mettre à jour plus finement les poids. On parle alors d'*époques* successives.

II – Calculs hydrodynamiques

II – 1 Cas-test

Le navire considéré dans cette étude est un porte-conteneurs de capacité 14400 EVP. Ses caractéristiques sont données dans le tableau ci-dessous.

Longueur (L_{PP})	350 m
Largeur (l)	48.2 m
Creux (C)	29.85 m
Tirant d'eau (T)	14.5 m

Tableau 1 – Caractéristiques du navire étudié

Dans les calculs considérés ici, le navire a une vitesse de 5 nœuds et la houle vient de face.

II – 2 Modèle linéaire

Le moment fléchissant au maître-couple du navire est un critère dimensionnant important pour s'assurer de la résistance du navire. À partir de la géométrie et de la distribution longitudinale de masse, un calcul potentiel linéaire (ici avec HydroStar [11]) permet de calculer la fonction de transfert des efforts internes, en particulier le moment fléchissant au niveau du maître-couple (section 13) qu'on peut voir sur la Figure 3. Le navire est ici considéré complètement rigide.

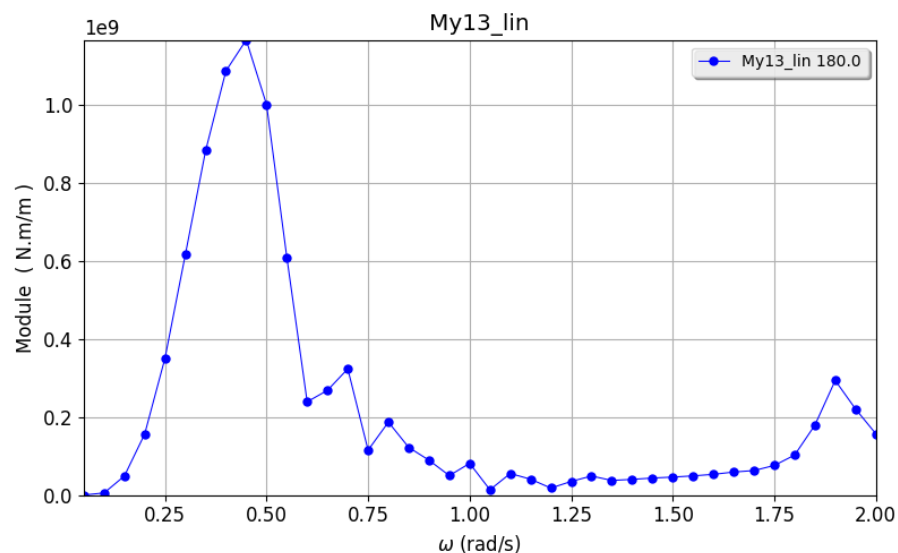


Figure 3 - Fonction de transfert du moment fléchissant linéaire au maître-couple pour une houle de face

II – 3 Modèle hydro-élastique

Le navire a fait l'objet d'investigations détaillées concernant son comportement hydro-élastique dans [3]. En effet, pour les porte-conteneurs d'aussi grande taille, Bureau Veritas recommande de prendre en compte les effets de la déformation élastique du navire dans le calcul du moment

fléchissant au maître-couple du bateau. À cette fin, le logiciel HOMER est utilisé. Un calcul structurel préalable permet de déterminer la forme des principaux modes de vibration ainsi que les matrices de masse et de raideur généralisées. Lors de la simulation hydro-élastique, en plus de se mouvoir rigidement autour de ses six degrés de liberté, le navire se déforme élastiquement le long des modes de vibration pré-calculés. Le chargement consiste en l'intégration de la pression due à la houle incidente jusqu'à la déformée non-linéaire de la surface libre, l'effort de diffraction linéaire dû à la houle, les efforts de radiation linéaires calculés à l'aide de fonctions de retard et les efforts liés aux impacts de tossage (*slamming*). Plus de détails sur l'implémentation et la théorie de HOMER peuvent être trouvés dans [4].

La montre le maillage d'intégration du navire ainsi qu'une représentation des sections de bateau pour lesquelles les efforts de tossage sont calculés à chaque fois que la vitesse relative entre la proue et la surface de l'eau excède une vitesse seuil.

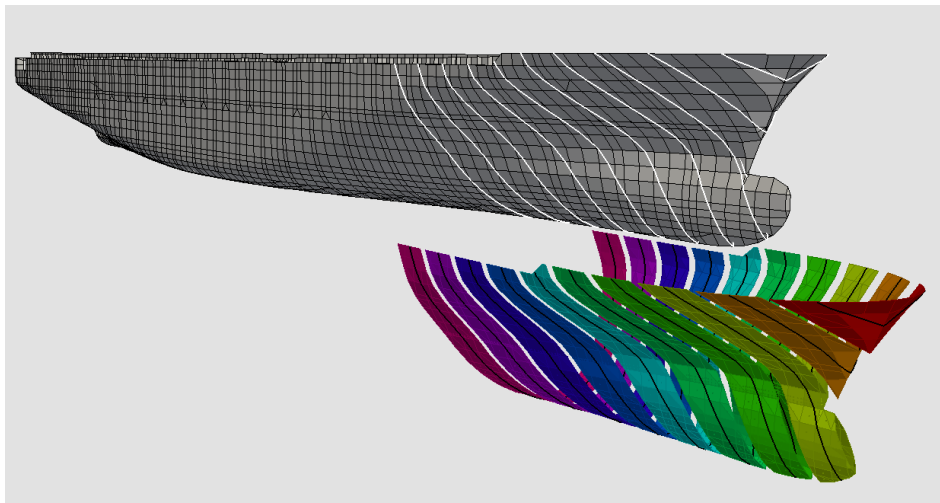


Figure 4 - Modèle HOMER du bateau avec sections de tossage figurées en traits blancs

Le temps de calcul de HOMER sur une architecture classique (un ordinateur de bureau) est de l'ordre du temps réel. Pour trois heures d'état de mer, il faut donc compter trois heures de temps de calcul hydro-élastique. Ceci fait de HOMER un outil idéal pour entraîner un réseau de neurones : c'est suffisamment rapide pour pouvoir générer un grand nombre de données, mais suffisamment lent pour ne pas pouvoir couvrir n'importe quel état de mer instantanément.

II – 4 Etats de mer

Quatorze états de mer d'une durée de six heures chacun ont été reproduits lors de l'étude de référence [2]. La liste est fournie dans le Tableau 2. Pour chaque période de pic considérée (T_p), deux hauteurs significatives (H_S) sont considérées : celle de l'état de mer de design (*DSS*) et celle de l'état de mer de design augmenté (*IDSS* pour *Increased Design Sea State*).

Etat de mer	Spectre	H_S	T_p	γ	N_{freq}	Direction
DSS-11.9	JONSWAP	12.5	11.964	1	200	0
IDSS-11.9	JONSWAP	21.4	11.964	1	200	0
DSS-13.4	JONSWAP	13.5	13.372	1	200	0
IDSS-13.4	JONSWAP	19.46	13.372	1	200	0
DSS-14.8	JONSWAP	14.5	14.784	1	200	0
IDSS-14.8	JONSWAP	18.01	14.784	1	200	0

DSS-16.2	JONSWAP	15.5	16.187	1	200	0
IDSS-16.2	JONSWAP	17.53	16.187	1	200	0
DSS-17.6	JONSWAP	16.5	17.59	1	200	0
IDSS-17.6	JONSWAP	17.68	17.59	1	200	0
DSS-19.0	JONSWAP	16.5	19.002	1	200	0
IDSS-19.0	JONSWAP	18.24	19.002	1	200	0
DSS-20.4	JONSWAP	18.5	20.409	1	200	0
IDSS-20.4	JONSWAP	19.11	20.409	1	200	0

Tableau 2 - Liste des états de mer

La Figure 5 représente la contribution de chaque état de mer à la valeur long-terme de moment fléchissant linéaire à la coupe au maître. L’IDSS correspond à une augmentation de 20% du H_S par rapport au DSS. Comme expliqué dans [5], cette augmentation artificielle de la hauteur de vagues permet de diminuer la période de retour d’un évènement en faisant l’hypothèse de l’indépendance des effets non-linéaires sur la hauteur significative.

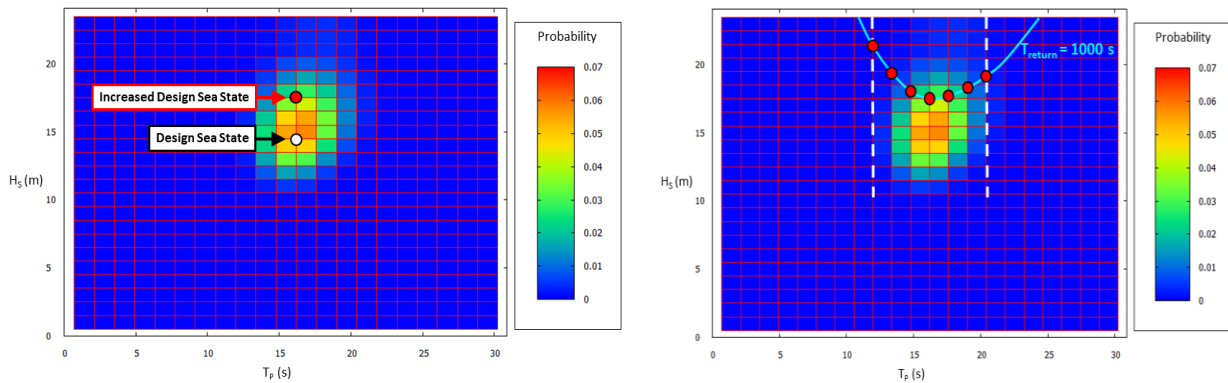


Figure 5 - Etat de mer de design et état de mer de design augmenté

III – Réseaux de neurones

III – 1 Architecture du réseau de neurones

Le réseau de neurones utilisé dans le cadre de cette étude est représenté schématiquement par la Figure 6. Il prend en entrée 26 attributs qui sont les valeurs de la houle incidente (non perturbée par le navire) aux 26 sections du bateau : la section 1 se situant à la poupe, la section 13 au maître-couple et la section 26 à la proue. Un échantillon correspond ici à un pas de temps avec un vecteur d’élévations de houle [wave1, wave2, ..., wave26] et une valeur cible associée : le moment fléchissant au maître-couple M_{y13} .

Nous considérons pendant toute l’étude un réseau constitué de 3 couches cachées successives composées respectivement 100, 200 et 100 neurones. Augmenter le nombre de couches ou de neurones n’améliore pas significativement la qualité des résultats. À ces neurones est associée une fonction d’activation de type RELU. Pour le neurone de sortie, une fonction d’activation linéaire est utilisée. La fonction de coût utilisée est l’erreur moyenne absolue (MSE : *mean squared error*).

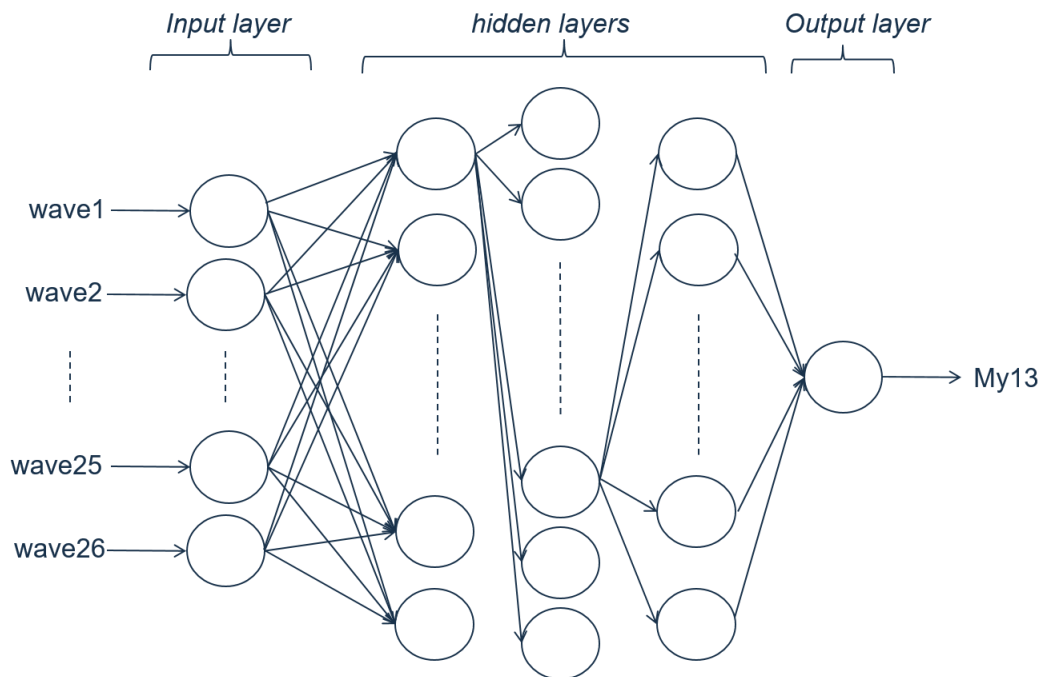


Figure 6- Architecture du réseau de neurones de base

III – 2 Résultats dans le cas linéaire

Le premier essai d'utilisation du réseau de neurones consiste en la prédiction du moment fléchissant linéaire. Associée aux 26 élévations de houle, la valeur cible est le moment fléchissant linéaire obtenu par une simple combinaison linéaire de la discrétisation fréquentielle de l'état de mer et de la fonction de transfert de moment fléchissant linéaire.

L'apprentissage et le test s'effectuent sur le même état de mer choisi comme étant le DSS19.0. Les séries temporelles sur cet état de mer sont scindées en 3 morceaux :

- Les données d'entraînement consistent en 40% de la durée des signaux temporels. Ce sont ces échantillons qui passent et repassent dans le réseau de neurones pour ajuster les paramètres.
- Les données de validation consistent en 10% de la durée des signaux temporels. Ces échantillons sont connus par le réseau lors de la phase d'entraînement, mais seulement à titre d'indicateur de la performance sur un ensemble de données tierces. Ils ne sont pas utilisés pour mettre à jour les paramètres du réseau.
- Les données de test consistent en les 50% restants des signaux temporels. Ces couples vagues incidentes-moment fléchissant sont complètement inconnus du réseau de neurones.

La Figure 7 montre en haut le découpage du signal temporel de moment fléchissant linéaire cible en partie entraînement, validation et test. La prédiction du réseau de neurones à la fin de son entraînement est aussi ajoutée pour l'ensemble du signal. Les trois zooms en bas de la figure correspondent aux trois traits verticaux roses figurant sur la série temporelle entière. Ils se situent respectivement pendant la phase d'entraînement, de validation et de test, et correspondent à des événements extrêmes. Il est clair que même sur des événements que le réseau de neurones n'a jamais vus (sur les données test), la prédiction est quasiment superposée avec les valeurs cibles.

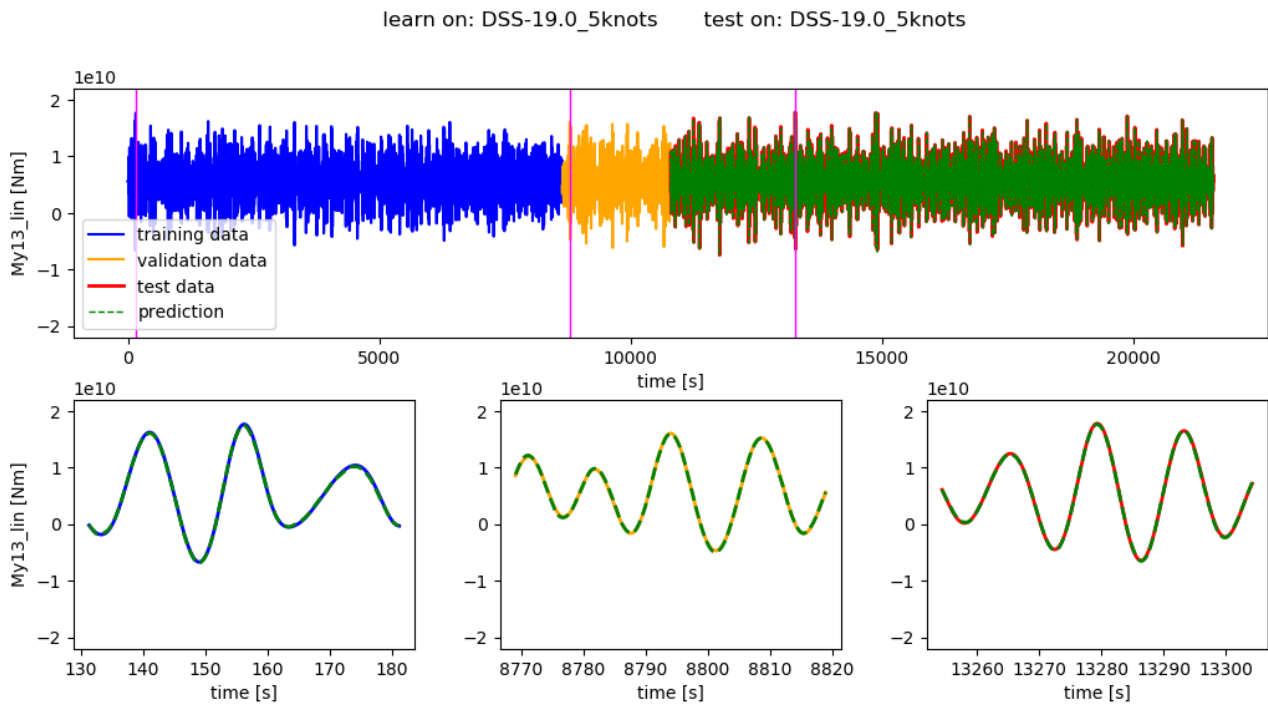


Figure 7 - Prédiction du moment fléchissant linéaire

Ceci revient à dire que grâce à un entraînement sur un peu plus de deux heures de données d'un état de mer, le réseau de neurones est capable de prédire exactement de manière déterministe le moment fléchissant linéaire à partir de n'importe quelle combinaison d'élévations incidentes le long du navire (provenant du même état de mer en termes de distribution d'énergie).

La Figure 8 montre l'évolution de l'erreur au cours des 50 époques de l'entraînement. On s'aperçoit que l'erreur baisse globalement de la même manière sur les données d'entraînement et sur les données de validation. L'entraînement du réseau de neurones est rapide (une minute sur un PC de bureau) et la phase de prédiction est instantanée.

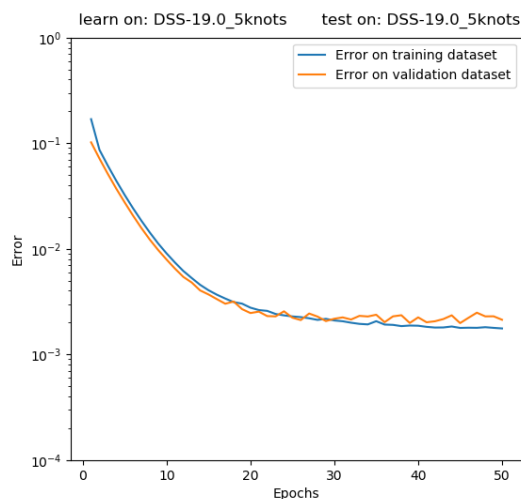


Figure 8 - Evolution de l'erreur pour les époques successives de l'entraînement

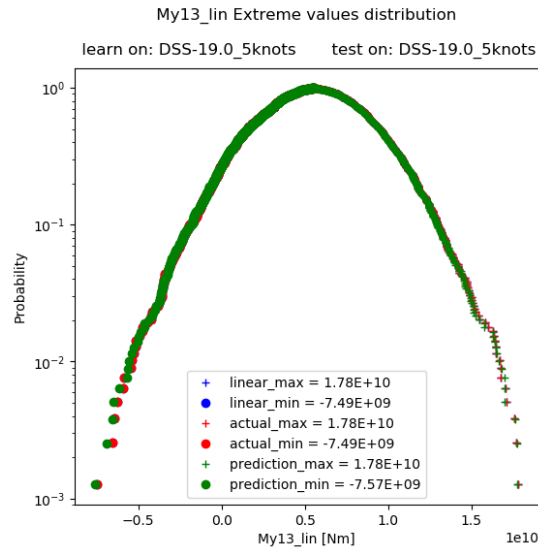


Figure 9 - Distribution des extrema de moment fléchissant linéaire

La Figure 9 procède d'une analyse *up-crossing* du signal de moment fléchissant linéaire et montre que la distribution des extrema est très bien reproduite à la fois en arc (moment positif) et en contre-arc (moment négatif).

III – 3 Résultats dans le cas hydro-élastique non-linéaire

Nous considérons maintenant le cœur du problème qui consiste à prédire le moment fléchissant dans le cas hydro-élastique non-linéaire. En effet, prédire le moment linéaire avec un réseau de neurones n'a pas d'intérêt pratique car à partir de la RAO et de la connaissance de la houle, son calcul est instantané. Il n'en va pas de même pour le cas non-linéaire où le calcul avec HOMER est de l'ordre du temps réel : pour 6 heures d'état de mer, il faut 6 heures de calcul. Ici, l'intérêt d'entraîner un réseau de neurones pourrait prendre tout son sens si l'on souhaite extrapoler sur le même état de mer ou bien apprendre sur un ou plusieurs états de mer et déduire le comportement du bateau sur un autre état de mer.

a) Apprentissage sur DSS, test sur même DSS

Nous reproduisons ici le cas exposé dans la section III-2 à la différence qu'ici le réseau est entraîné et comparé avec le moment fléchissant non-linéaire. Sur la Figure 10, il est visible que le résultat de la prédiction du réseau de neurones est meilleur pour les données d'entraînement que pour les données de test. Cependant, il faut tout de même noter qu'avec une architecture relativement simple le réseau de neurones est capable de prédire de manière plutôt précise le comportement du moment fléchissant non-linéaire et en particulier les oscillations à haute fréquence dues à la vibration hydro-élastique de la poutre-navire. La Figure 11 montre l'évolution de l'erreur pour les différentes époques. On voit ici qu'à partir de 10 époques, l'erreur sur les données d'entraînement continue de diminuer tandis que sur les données de validation l'erreur stagne, voire a tendance à croître. Ceci est un exemple typique de surapprentissage (« *overfitting* »). À partir d'un moment, le réseau de neurones est de plus en plus performant sur les données d'entraînement au détriment de sa généralisation à de nouvelles données indépendantes.

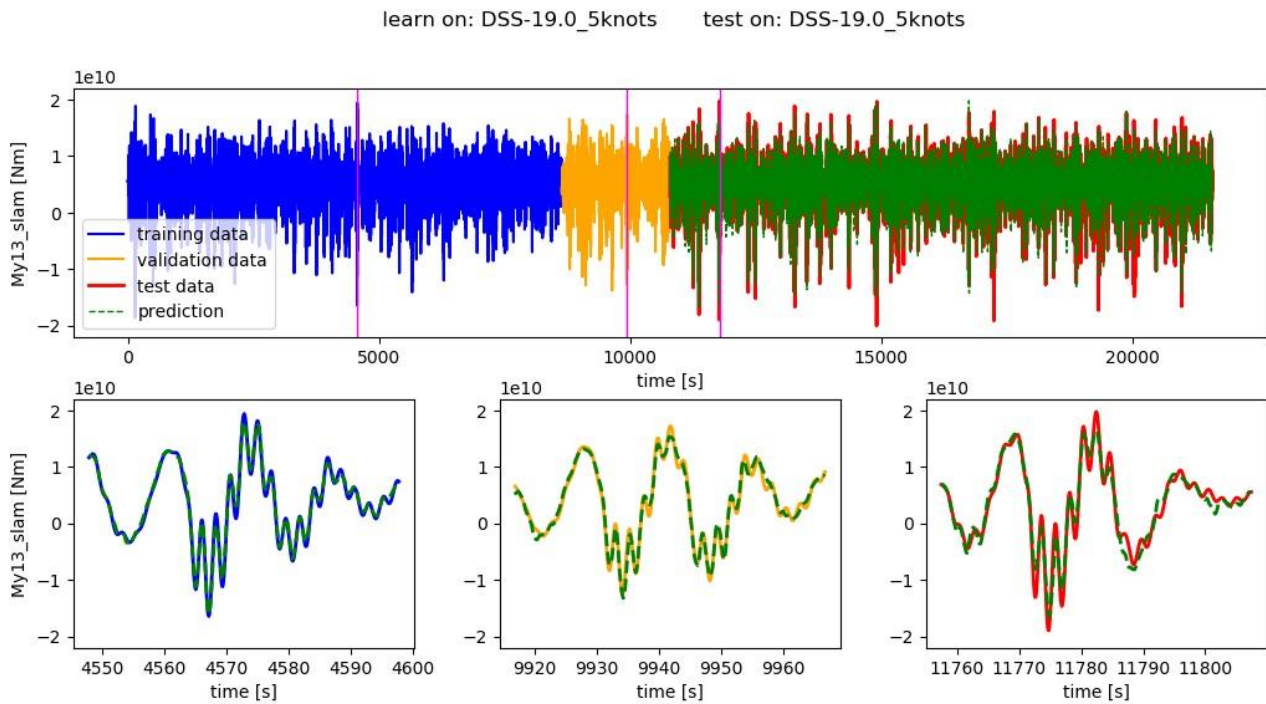


Figure 10 - Prédiction du moment fléchissant hydro-élastique

La Figure 12 montre la distribution des extrema. On voit que les évènements extrêmes sont sous-évalués par le réseau de neurones. Ceci s'explique en premier lieu par le fait que plus l'évènement est extrême, plus le nombre de cas similaires rencontrés dans la phase d'entraînement est faible. Sur la figure, la distribution des extrema du signal de moment fléchissant linéaire est ajoutée en bleu. Elle montre que même si le réseau de neurones sous-estime les extrema de moment fléchissant non-linéaire, sa performance (en un temps très court d'entraînement) est bien meilleure que le modèle linéaire.

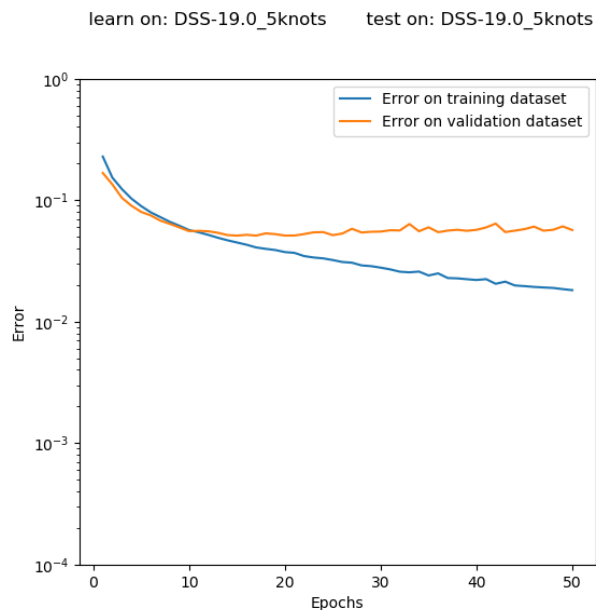


Figure 11 - Évolution de l'erreur pour les époques successives d'entraînement

learn on: DSS-19.0_5knots test on: DSS-19.0_5knots

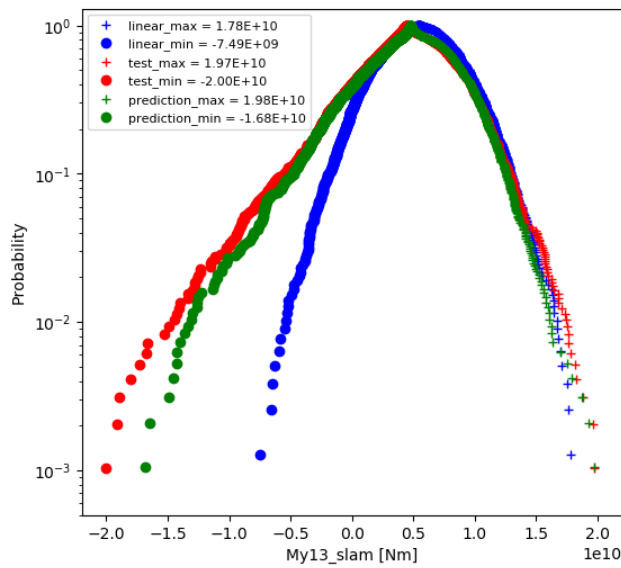


Figure 12 - Distribution des extrema

b) Apprentissage sur tous les états de mer IDSS, test sur un DSS

Pour diminuer le problème de la sous-représentation des événements extrêmes dans la période d'entraînement du réseau de neurones, il a été proposé d'entraîner le réseau de neurones sur les états de mer augmentés (IDSS) pour ensuite faire des prédictions sur les états de mer de design, dans une approche similaire à ce qui a été réalisé dans [5]. Cependant, les résultats du réseau de neurones ne sont que marginalement meilleurs pour les différents états de mer testés. Par exemple, sur le cas du DSS19.0 montré sur les Figure 13 et Figure 14, les résultats sont comparables au cas précédent où le réseau est entraîné sur le même état de mer.

learn on: All_IDSS test on: DSS-19.0_5knots

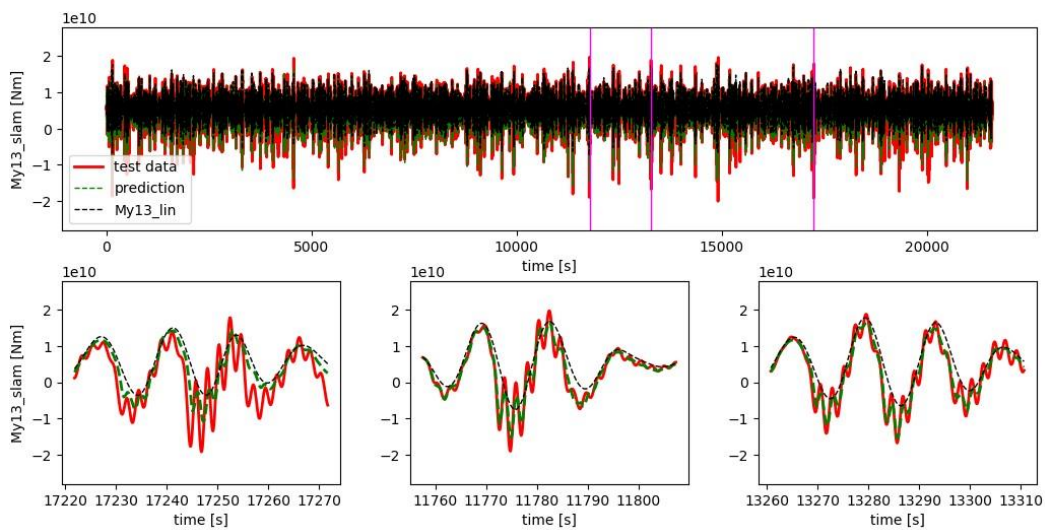


Figure 13 - Prédiction du moment fléchissant hydro-élastique

learn on: All_IDSS test on: DSS-19.0_5knots

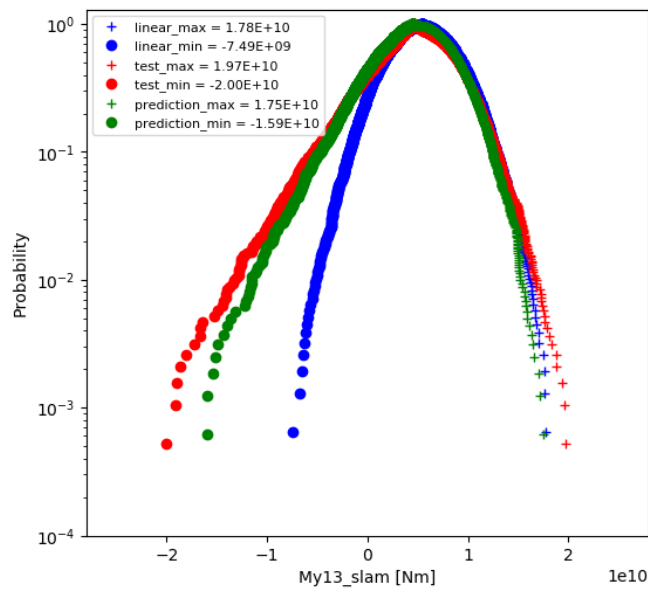


Figure 14 - Distribution des extrema

c) Redéfinition des attributs

Ce qui demande le plus de réflexion dans la mise en place d'un réseau de neurones est le choix correct des attributs pour le problème considéré et éventuellement la construction de nouveaux attributs que l'on estime plus corrélés avec la cible. Ceci est appelé « *feature engineering* » en anglais.

i) Utilisation de signaux temporels issus du calcul linéaire

Le premier essai consiste à créer des attributs issus du calcul linéaire qui sont très rapides à construire. Nous avons ici ajouté comme attributs au réseau de neurones les valeurs du moment fléchissant linéaire, de l'accélération en pilonnement linéaire et de l'accélération en tangage linéaire. La corrélation entre le moment fléchissant linéaire et non-linéaire semble assez intuitive et le fait que le moment fléchissant non-linéaire dépende largement de l'effet du tossage explique qu'une corrélation avec les accélérations en pilonnement et en tangage est attendue.

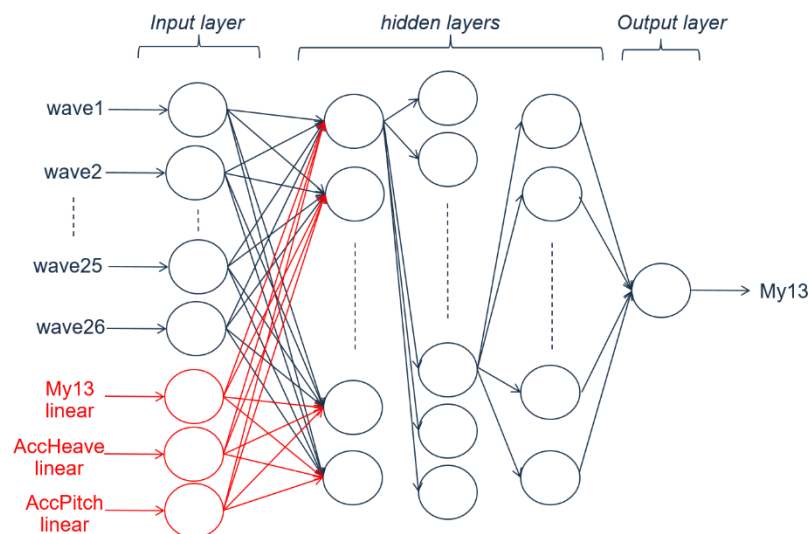


Figure 15 - Inclusion d'attributs issus d'un calcul linéaire dans le réseau de neurones

La Figure 16 montre la distribution des extrema pour ce réseau de neurones (entraîné sur tous les IDSS et testé sur DSS-19.0). Malheureusement, le résultat n'est pas meilleur que dans le cas de base. Ceci peut être expliqué par le fait que les trois nouveaux attributs sont tout à fait bien prédits par le réseau de neurones entraîné avec les 26 élévations de houle (comme montré dans la section III-2 pour le cas du moment fléchissant linéaire).

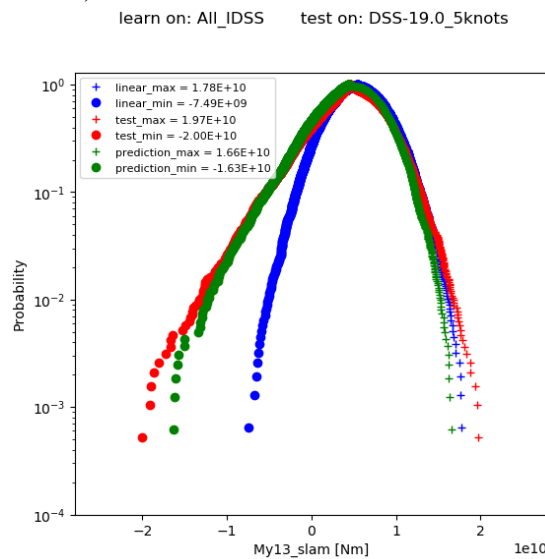


Figure 16 - Distribution des extrema pour le réseau de neurones entraîné avec les attributs issus d'un calcul linéaire

ii) Prise en compte de l'effet mémoire

La limitation évidente des réseaux précédents est le fait d'essayer de prédire le moment fléchissant non-linéaire à partir de l'élévation instantanée le long du bateau. Or, il est clair qu'il existe un fort effet mémoire. La vibration de la poutre-navire (et par là le moment fléchissant) dépend bien sûr de l'historique récent et en particulier de l'importance du pic précédent. Il semblerait que c'est en particulier vrai pour le pic en arc qui dépend de la valeur du pic en contre-arc précédent. Il serait trop lourd en temps de calcul de considérer toutes les élévations à tous les pas de temps précédents comme attributs. Le maximum que nous avons pu réaliser est de considérer l'élévation le long du navire à 8 pas de temps précédents pris à intervalles réguliers lors des dernières 20s (soit environ un T_p du plus grand état de mer). Le nouveau réseau est représenté sur la Figure 17.

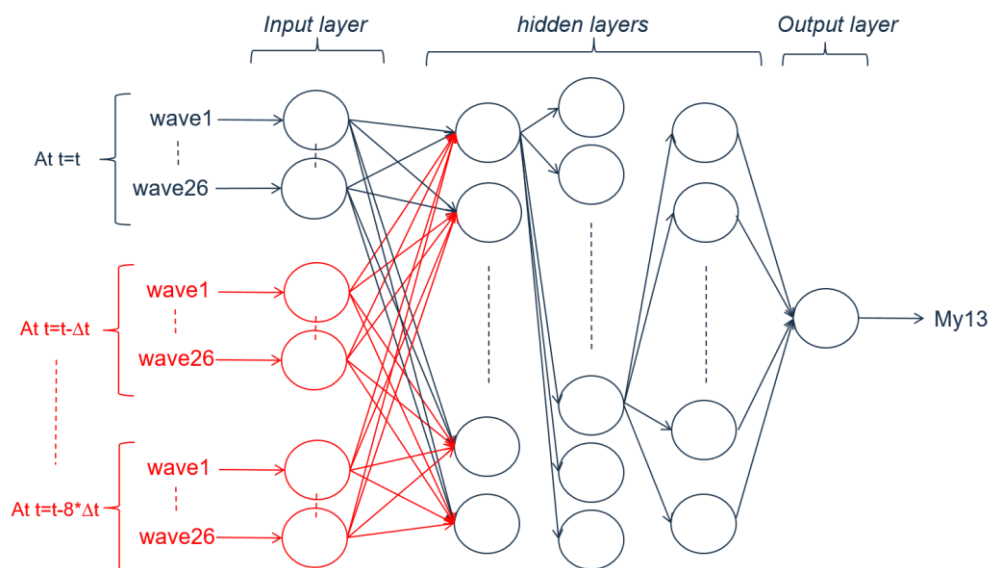


Figure 17 - Inclusion de l'élévation de la houle à certains pas de temps précédents comme attributs

Sur la Figure 18, on peut noter une légère amélioration dans la distribution des extrema. Néanmoins, une sous-estimation des événements extrêmes est toujours notable et l'on touche probablement aux limites du réseau de neurones, tout du moins sous cette architecture.

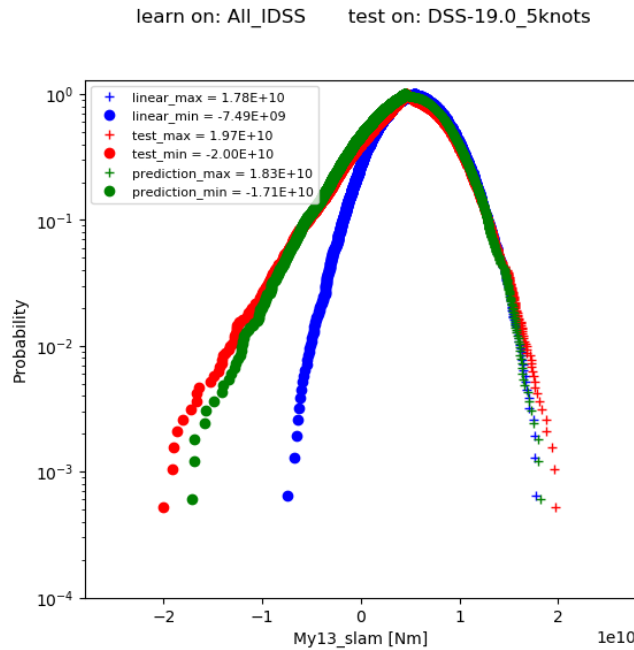


Figure 18 - Distribution des extrema pour un réseau de neurones entraîné avec les valeurs d'élévation de la houle aux pas de temps précédents

IV – Conclusions

Il a d'abord été vérifié qu'un réseau de neurones était capable de prédire quasi-exactement le moment fléchissant linéaire d'un bateau à partir de la connaissance de la houle seule. Ceci valide l'implémentation mais la génération de séries temporelles de moment linéaire a peu d'utilité pratique.

Le potentiel intérêt de l'utilisation d'un réseau de neurones est présent lorsque le temps de simulation du calcul direct devient significatif, comme c'est le cas pour le calcul du moment fléchissant hydro-élastique non-linéaire. Plusieurs configurations ont été testées. Dans un premier temps, le réseau de neurones a été entraîné sur un état de mer pour une durée donnée et sa prédiction a ensuite été comparée avec le résultat de calcul d'HOMER pour le reste de la simulation. Dans un second temps, après avoir été entraîné sur tous les états de mer augmentés, la qualité de la prédiction du réseau de neurones a été évaluée sur un autre état de mer de design. Dans les deux cas, la performance du réseau de neurones est très bonne au regard de son faible temps de calcul. La limitation de cette méthode est à chercher dans la sous-estimation des événements extrêmes, qui sont par définition peu nombreux dans la phase d'apprentissage. De l'avis même des experts du domaine [1], la configuration optimale d'un réseau de neurones résulte d'un travail largement empirique à base d'essais successifs. La prise en compte de l'effet mémoire par l'inclusion dans le vecteur d'attributs de l'élévation de houle incidente à plusieurs pas de temps précédent améliore légèrement la prédiction.

Au-delà du calcul des extrêmes, le réseau de neurones peut aussi être utilisé pour déterminer rapidement le dommage en fatigue sur plusieurs états de mer. Comme le montre le Tableau 3, le dommage est aussi sous-estimé par le réseau de neurones pour DSS19.0, mais des tests plus généraux pourraient certainement permettre d'appliquer un biais correctif sur cette valeur qui est plus rapidement convergée. Le Tableau 3 montre aussi que la prise en compte de l'effet mémoire améliore légèrement la prédiction de tenue en fatigue.

Implémentation	Domage linéaire	Domage test	Domage prédiction	Erreur de la prédiction comparée au domage test
Architecture « de base » (b)	0.00554	0.01319	0.00919	-30%
Inclusion attributs linéaires (c i)	0.00554	0.01319	0.00983	-25%
Effet mémoire (c ii)	0.00554	0.01319	0.0108	-18%

Tableau 3 - Domage en fatigue sur l'état de mer DSS19.0 à partir de l'entraînement sur tous états de mer IDSS

Pour finir, d'autres architectures de réseaux de neurones doivent être investiguées, en particulier les réseaux de neurones récurrents qui prennent naturellement en compte l'état du réseau aux itérations précédentes. Il est possible que leur utilisation (un peu plus complexe) puisse permettre d'améliorer encore un peu plus la précision de la prédiction.

Références

- [1] F. Chollet. Deep Learning with Python. *Manning Publications Co*, 2018.
- [2] N.H. Christiansen, P.E.T. Voie, J. Høgsberg. Artificial neural Networks for reducing computational Effort in active truncated model Testing of Mooring Lines. *Proceedings of the ASME 2015 34th International Conference on Ocean, Offshore and Arctic Engineering*. OMAE 2015.
- [3] A. Benhamou. 14400 TEUs WhiSp2 analysis. *NT 3248*. BV internal document, 2015.
- [4] Derbanne, Q., Malenica, S., Tuitman, J., Bigot, F., & Chen, X.-B. Validation of the global hydroelastic model for springing & whipping of ships. *PRADS*. Rio de Janeiro, 2010.
- [5] Derbanne, Q., Bigot, F., de Hauteclocque, G. Comparison of design wave approach and short term approach with increased wave height in the evaluation of whipping induced bending moment. *OMAE*, 2012.
- [6] Reid, Stuart G., TuringFinance Article: "10 Misconceptions of Neural Networks [in Finance]", May 2014, <http://www.turingfinance.com/misconceptions-about-neural-networks/>
- [7] <https://keras.io/>, "Keras: The Python Deep Learning library"
- [8] <https://www.tensorflow.org/>, "An open source machine learning framework for everyone"
- [9] <https://machinelearningmastery.com/tutorial-first-neural-network-python-keras/>, "Develop Your First Neural Network in Python With Keras Step-By-Step"
- [10] LeCun, Y., <https://www.college-de-france.fr/site/yann-lecun/>, Cours au collège de France
- [11] Chen X.B. "Hydrodynamics in offshore and naval applications - part I". Keynote lecture of 6th Intl. Conf. HydroDynamics, Perth (Australia), 2004.