
Approximation de calculs éléments finis par réseaux neuronaux : quelques applications en optimisation

Jean-Luc Marcelin — Assad Kallassy

*Laboratoire Sols, Solides, Structures
UMR CNRS C5521
BP 53, 38041 Grenoble cedex 9*

RÉSUMÉ. Une stratégie d'optimisation (algorithmes génétiques) utilisant des analyses approchées par approximation des calculs éléments finis à l'aide de réseaux neuronaux est présentée. Les exemples montrent la bonne adéquation de cette approximation neuromimétique qui a pour conséquence des gains importants en temps de calcul pour l'optimisation.

ABSTRACT. This work examines the possibility of using some genetic algorithms and some neural networks to optimize mechanical structures ; examples show that using this stochastic method and neural networks is efficient.

*MOTS-CLÉS : optimisation, algorithmes génétiques, éléments finis, réseaux neuronaux.
KEYWORDS : optimization, genetic algorithms, finite elements, neural networks.*

1. Introduction

Ce travail s'inscrit dans le cadre de l'optimisation des structures ou des systèmes mécaniques. Parmi les méthodes d'optimisation utilisées actuellement, les méthodes dites stochastiques ou probabilistes connaissent un développement important. Cet intérêt est dû à leurs nombreux avantages, sur lesquels on revient par la suite; leur inconvénient majeur est le volume des calculs qu'elles peuvent générer, surtout dans le cas d'analyses par les éléments finis. Le but de cet article est de montrer que certains calculs par éléments finis peuvent être approximés par des réseaux neuronaux et rendre ainsi l'optimisation stochastique plus rapide.

On a utilisé à plusieurs reprises une des méthodes stochastiques les plus connues, à savoir la méthode des algorithmes génétiques, dans le cadre de divers problèmes d'optimisation de systèmes ou de structures mécaniques. Ces algorithmes se sont généralement révélés efficaces, comme par exemple pour la maximisation de l'amortissement de poutres ou plaques composites [MAR 94], [MAR 95c], [MAR 95d], [MAR 95e], pour l'optimisation des engrenages [MAR 95a], ou l'optimisation des conditions aux frontières avec des applications pour le bridage en usinage [MAR 95b].

Rappelons que ces méthodes stochastiques ont en commun le fait d'utiliser des processus ou propriétés probabilistes pour assurer la convergence de l'algorithme. Généralement, elles utilisent l'évaluation de la fonction objectif sur un échantillon aléatoirement choisi, puis l'exécution de procédures pouvant donner lieu à des calculs prohibitifs si l'analyse des fonctions coûts nécessite des temps de calcul importants.

L'objectif de cet article est de montrer que l'efficacité de ces algorithmes peut dans ce cas être considérablement améliorée, en utilisant des réseaux neuronaux [JOD 94] pour l'analyse du problème à la place d'une analyse par éléments finis classique. L'utilisation de réseaux de neurones pour la modélisation de structures mécaniques semble donner de bons résultats [BER 92] et [SZE 93]. En effet, l'apprentissage d'un réseau neuromimétique peut se faire en parallèle avec le calcul EF des premières générations (c'est-à-dire en utilisant les résultats des analyses par éléments finis). Une fois l'apprentissage terminé, le réseau neuronal remplace complètement les calculs par éléments finis, d'où des évaluations plus rapides, d'autant que la méthode des algorithmes génétiques, au contraire des méthodes déterministes, n'exige pas de calculs très précis de la fonction objectif. Après une description des méthodes utilisées, on donne des exemples d'utilisation des réseaux neuronaux en modélisation de structures mécaniques, et des exemples d'utilisation des algorithmes génétiques couplés avec les réseaux neuronaux pour optimiser ces mêmes structures mécaniques.

2. Les méthodes utilisées

2.1. Les algorithmes génétiques

Les algorithmes génétiques [GOL 89] font partie des méthodes d'optimisation stochastiques. Ils ne nécessitent pas de fortes régularités de la fonction objectif et des contraintes, et permettent la localisation de plusieurs optima. Le principe général est de maximiser une fonction ($\text{Max } F(x)$, x appartenant à un ensemble A) à partir de l'évolution d'une population P (p -uplet) d'individus (points de A) suivant un principe caricaturalement darwinien. Le codage des individus sous forme de chromosomes (nombres binaires par exemple) est une caractéristique des A.G.

La mise en oeuvre des algorithmes génétiques commence par une initialisation aléatoire de la population, de taille p , puis par la construction d'une succession de générations. La sélection et la reproduction se font par copie des parents performants de la population P_i ; la sélection peut-être stochastique ou déterministe. La création des enfants se fait par application d'opérateurs génétiques (croisement, mutation); elle est toujours stochastique. Le remplacement de la population P_i par

la population P_{i+1} s'effectue par sélection des individus performants, soit parmi les enfants, soit parmi les parents et les enfants; ce remplacement est stochastique ou déterministe. La performance d'un individu est mesurée par la valeur que prend $F(x)$. Les bases des algorithmes génétiques sont décrites dans [GOL 89] et [HOL 92]. On a utilisé ici l'algorithme génétique simple décrit dans [GOL 89].

2.2. Les réseaux de neurones

On trouve dans l'ouvrage [JOD 94] le détail de la théorie générale des réseaux de neurones que l'on ne reprend pas ici. L'utilisation d'un réseau de neurones pour la simulation ou la modélisation se fait en deux temps: une phase dite d'apprentissage utilise des résultats de calculs par éléments finis; elle est suivie d'une phase de calcul ou de généralisation. Dans le cas présent, le réseau de neurones doit pouvoir estimer une fonction objectif ou fonction coût en fonction des variables d'entrée ou de conception. On peut même évaluer plusieurs fonctions si l'on veut calculer par exemple des contraintes d'optimisation. A noter ici que les variables d'entrée sont directement les digits binaires des chromosomes définis pour l'algorithme génétique. Pour décrire un réseau de neurones, il suffit de connaître le modèle de neurone (figure 1), la fonction neurone (figure 2) et l'organisation des connexions entre les neurones (figure 3).

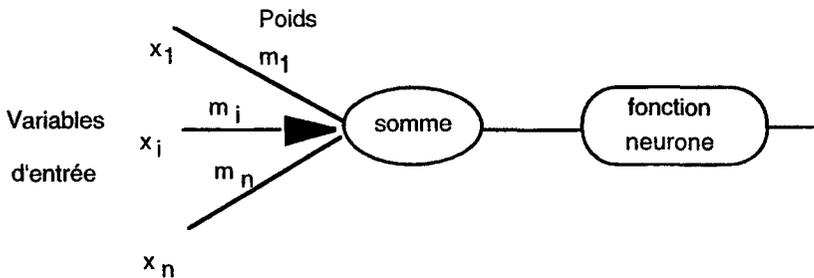


Figure 1. Neurone élémentaire

L'application des réseaux de neurones à la modélisation, notamment à la simulation de calculs de structures mécaniques est relativement récente et semble prometteuse au vu des résultats obtenus ([BER 92], [SZE 93], [SZE 94], [HAJ 94]). Le passage à la modélisation peut sembler naturel car l'action de modéliser un processus ou un comportement nécessite la connaissance des caractéristiques principales de ce processus ou de ce comportement, caractéristiques que le réseau sait extraire et qu'il peut donc facilement mémoriser. Cette aptitude à modéliser exploite d'autre part les qualités d'adaptation des réseaux, en leur permettant de s'améliorer au fur et à mesure de leur exploitation. Dans les articles référencés ci-dessus, les applications se limitent le plus souvent à des calculs de treillis de poutres en statique et donnent de bons résultats. On donne ci-après des exemples un peu différents, notamment en amortissement des vibrations.

Dans cette étude, on utilise deux réseaux de neurones: le réseau Adaline fourni dans l'ouvrage [JOD 94] (Adaline est un réseau élémentaire à une couche et à fonction neurone linéaire) et un Perceptron Multi-Couches (PMC) décrit dans le même ouvrage. Ces réseaux de neurones ont été programmés en langage C++.

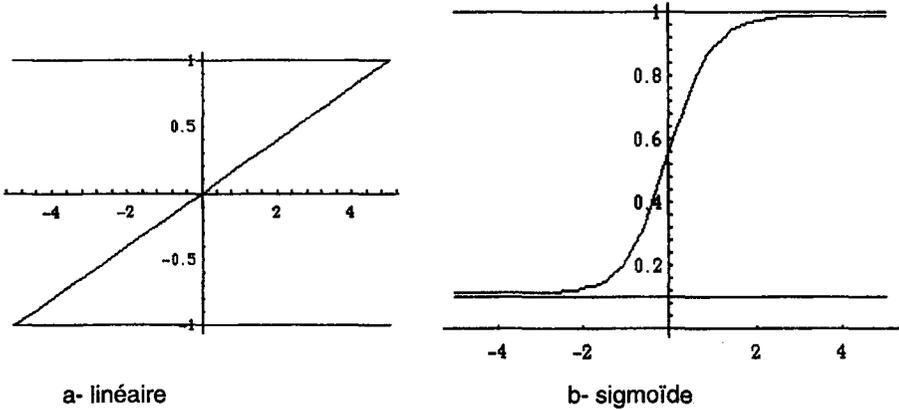


Figure 2. Deux types de fonctions neurone; linéaire pour le réseau Adaline, sigmoïde pour le Perceptron Multi-Couche.

Le PMC se structure en couches où le potentiel de la première couche (couche d'entrée) est fixée à l'entrée du réseau et l'activité de la dernière couche (couche de sortie) est interprétée comme la réponse du réseau. Une ou plusieurs couches intermédiaires (couches cachées) s'interposent entre l'entrée et la sortie. Le calcul de l'activation dans le réseau s'effectue en propageant l'activation initiale de la couche d'entrée jusqu'à la couche de sortie, et la connexion est totale entre deux couches consécutives; ainsi un neurone d'une couche cachée reçoit toutes les activations des neurones de la couche précédente et émet son activation vers tous les neurones de la couche suivante.

La phase d'apprentissage consiste à optimiser ou adapter au moyen d'une règle d'apprentissage les poids affectés à chaque liaison; on utilise pour ce faire un échantillon d'apprentissage, c'est-à-dire des solutions qui sont déterminées au préalable par la méthode des éléments finis. Le principal critère est d'obtenir une erreur minimale, cette erreur est définie comme la moitié de la somme des carrés des différences entre les activations des neurones de sortie et la sortie désirée de ces mêmes neurones. Plusieurs algorithmes ont été utilisés pour l'optimisation; la méthode adoptée ici est analogue à la règle de Widrow-Hoff, applicable en particulier à des réseaux à deux couches complètement interconnectées avec une fonction de neurone linéaire tel que $f(x)=x$ (Adaline) et qui ensuite a été généralisée aux réseaux multicouches (PMC) grâce à l'algorithme de rétropropagation du gradient de l'erreur, l'erreur à la sortie de chaque neurone étant exprimée en fonction de l'erreur calculée pour la couche suivante par un calcul différentiel simple.

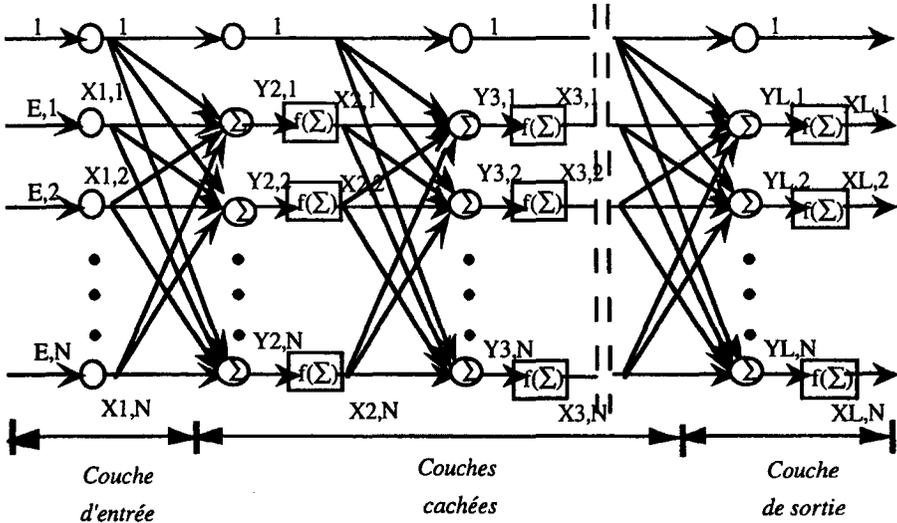


Figure 3. Le réseaux à connexions (PMC) utilisé dans le cadre de cette étude.

3. Les résultats obtenus

Deux cas tests dans le domaine des vibrations des structures mécaniques sont présentés ici pour les deux raisons suivantes: les techniques d'optimisation déterministes sont souvent difficiles à mettre en oeuvre en vibrations à cause de calculs délicats et souvent difficiles des sensibilités; par ailleurs les temps de calcul sont souvent plus longs que lors d'une optimisation en statique. Ces difficultés sont levées par les algorithmes génétiques qui n'utilisent pas de dérivées et qui peuvent traiter des fonctions non continues. L'optimisation des structures vibrantes est donc un domaine tout indiqué pour montrer l'intérêt de la stratégie d'algorithmes génétiques alliés aux réseaux neuronaux.

3.1. Test 1: optimisation des sections pour maximiser la première fréquence d'un treillis rigide

Le premier exemple a pour objectif la maximisation de la première fréquence de résonance d'un treillis rigide en acier dont la déformée du premier mode est donnée sur la figure 4. Ce premier exemple est vraiment un exemple académique, d'autant plus académique qu'il s'agit en fait d'un petit treillis expérimental utilisé par des élèves-ingénieurs dans le cadre de travaux pratiques de vibrations (c'est pour cela que les dimensions ne semblent pas bien réalistes). Ce treillis permet une première expérimentation numérique. On aurait tout aussi bien pu choisir un exemple test d'optimisation connu (par exemple le treillis à 10 barres), mais en général ces cas tests en optimisation sont traités en statique, et le problème de la maximisation de la première fréquence n'est pas abordé; il n'y aurait donc pas eu de toutes façons de solution de référence pour comparer avec les résultats obtenus ici. On peut remarquer que la maximisation de la première fréquence revient à chercher

un compromis entre une raideur maximale et un poids minimum, puisque la fréquence est proportionnelle au rapport raideur sur masse.

Le problème n'est pas contraint par souci de simplicité, le but essentiel étant la mise en oeuvre des réseaux de neurones. Si l'on veut par exemple prendre des contraintes sur le poids ou la flèche, on peut le faire en utilisant des fonctions de pénalisation. Cela a déjà été fait dans [MAR 95c] et [MAR 95d] en pénalisant la fonction coût de manière multiplicative par une fonction tendant vers 0 quand le critère n'est pas respecté et tendant vers 1 quand le critère est respecté, ce qui permet d'affecter une chance de survie faible aux chromosomes qui ne respectent pas les contraintes. Les algorithmes génétiques se prêtent relativement bien à ce genre de manipulation. D'autre part, un réseau de neurones est capable d'évaluer plusieurs sorties, c'est-à-dire la fonction objectif proprement dite, ainsi que les contraintes d'optimisation.

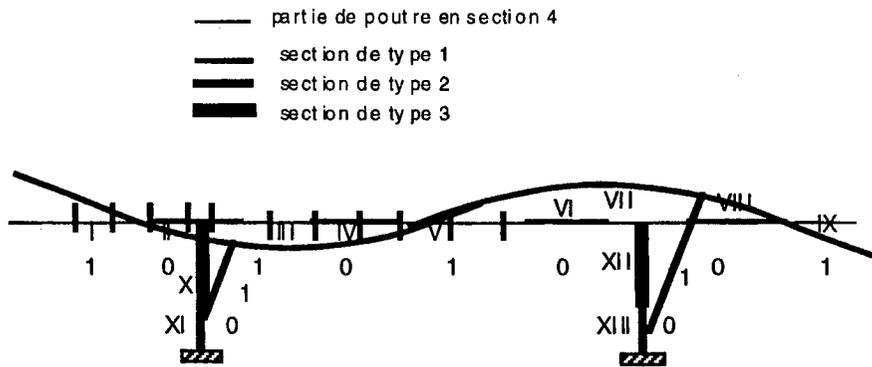


Figure 4. Définition de la déformée correspondant au 1er mode, et des 13 secteurs, dans le cas d'un chromosome 101010101010.

Le treillis est discrétisé par 36 éléments finis de poutre, dont 6 pour chaque poutre verticale. Les poutres ont une section rectangulaire de largeur b et de hauteur h . La hauteur totale du treillis est de 0.143 m et sa longueur totale de 0.68 m. Ce treillis est subdivisé en 13 zones dont les sections peuvent varier et vont être les variables d'optimisation. C'est ainsi que 9 zones sont discrétisées chacune par deux ou trois éléments finis pour la poutre horizontale, les sections de ces 9 zones pouvant être choisies parmi 2 possibilités: section de type 1 ($b=0.03$ mm, $h=0.0059$ mm) ou section de type 4 ($b=0.02$ mm, $h=0.003$ mm). De même les poutres verticales sont subdivisées chacune en 2 zones de 3 éléments finis dont les sections peuvent être choisies parmi 2 autres possibilités: section de type 2 ($b=0.02$ mm, $h=0.01$ mm) ou section de type 3 ($b=0.02$ mm, $h=0.02$ mm). Ce problème est intéressant car il comporte un nombre important de combinaisons possibles, à savoir 2^{13} . Les chromosomes choisis ont pour longueur 13 digits binaires; les 9 premiers digits correspondent au codage des 9 premiers secteurs définis de la manière suivante: le digit 0 correspond à la section de type 1 et le digit 1 correspond à la section de type 4; les 4 derniers digits correspondent au codage des 4 secteurs pour les poutres verticales, et ceci de la manière suivante: le digit 0 correspond à la section de type 2 et le digit 1 à la section de type 3. C'est ainsi que le chromosome 111111110000 correspond à un choix de la section de type 4 pour

les 9 secteurs de la poutre horizontale et à un choix de la section de type 2 pour chacun des deux secteurs des deux poutres verticales. La figure 4 donne la configuration pour le chromosome 101010101010.

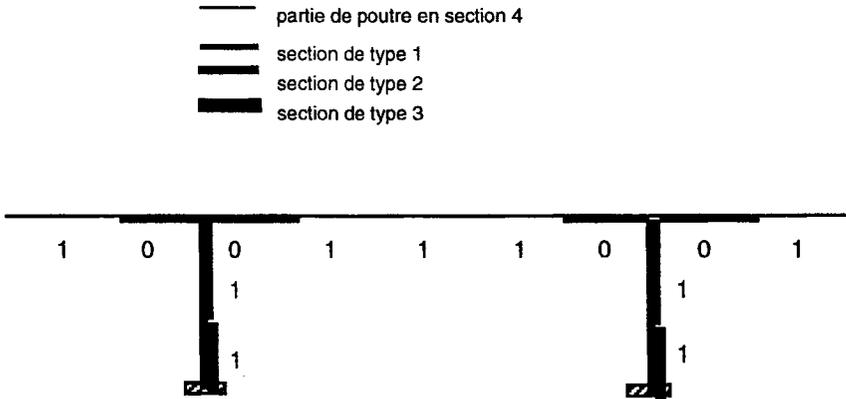


Figure 5. Solution optimale.

L'optimisation par algorithme génétique est exécutée deux fois avec des paramètres différents, les analyses étant faites dans les deux cas par la méthode des éléments finis. Le premier passage correspond à une population de 30 individus, une probabilité de croisement de 0.6 et une probabilité de mutation de 0.03 (ces valeurs sont les valeurs généralement conseillées dans la littérature, le phénomène de croisement devant être plus fréquent que le phénomène de mutation pour une efficacité optimale de l'algorithme). La convergence est atteinte à la 21^{ème} génération, c'est-à-dire au prix de $30 \times 21 = 630$ analyses par éléments finis; le critère de convergence est l'apparition pour la première fois du meilleur individu, qui se révèle être le chromosome 1001110011111 (fréquence 230 Hz) et qui correspond à la solution symétrique de la figure 5. A partir de ce moment là, ce meilleur individu reste toujours présent dans les populations ultérieures. Le deuxième passage correspond à une population de 50 individus, une probabilité de croisement de 0.8 et une probabilité de mutation de 0.02. La convergence est atteinte cette fois-ci à la 28^{ème} génération, c'est-à-dire au prix de 1400 analyses par éléments finis. L'optimum trouvé est le même que pour le premier passage. Remarquons que la solution qui consiste à donner une section de type 3 à l'ensemble du treillis conduit à une première fréquence de 234 Hz, c'est-à-dire équivalente à la fréquence optimale de 230 Hz, mais pour un poids nettement supérieur. Les courbes de convergence pour les deux passages (valeur moyenne des coûts de chaque génération) sont données sur la figure 6.

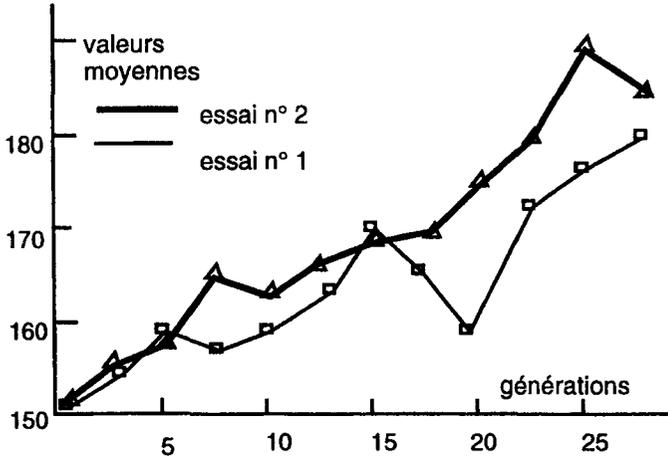


Figure 6. Valeurs moyennes de la première fréquence de vibration de chaque population en fonction des générations

La stratégie d'approximation des calculs par éléments finis est utilisée à l'aide d'abord du réseau Adaline et ensuite du réseau de neurones à trois couches à fonction neurone sigmoïde décrit dans la partie 2.2; les paramètres sont les suivants:

- population de 50 individus
- probabilité de croisement de 0.6
- probabilité de mutation de 0.03.

Les 50 premières analyses de la première génération sont effectuées par la méthode des éléments finis; ces 50 calculs servent à faire l'apprentissage du réseau de neurones, les vecteurs d'apprentissage utilisés étant directement les chromosomes avec la fréquence correspondante. Les calculs d'analyse de l'optimisation sont alors faits de manière approchée dès la deuxième génération. (Remarquons qu'il y a aussi possibilité de réactualiser l'apprentissage au bout de quelques générations car les éléments de la population sont alors plus proches de la solution optimale. L'utilisation de cette possibilité n'a pas été néanmoins nécessaire ici). Les paramètres d'apprentissage et d'évaluation pour le réseau PMC sont les suivants:

- utilisation de 3 couches;
- pas d'apprentissage: 0.002;
- nombre maximum d'itérations pour l'apprentissage: 3000;
- erreur maximale admise: 0.00001 (l'erreur est définie comme la moitié de la somme des carrées des différences entre les activations des neurones de sortie et la sortie désirée de ces mêmes neurones).

On trouvera dans le tableau ci-dessous les 50 vecteurs d'apprentissage utilisés (en premier lieu est donné le chromosome puis la fréquence correspondante en Hz):

chromosome	fréquence HZ	chromosome	fréquence HZ
1011101011101	167	1000000010000	143
1011010101001	121	0010100011101	175
1001000111110	174	0010111000111	164
0110011101011	132	0001011010100	146
0100111111011	135	0000010010000	132
1101001010111	164	0010100010100	151
1011100001000	133	1100001110001	151
0000111011101	190	0110001010001	125
1011101011101	167	0110001110010	117
0100110001000	121	100000001000	134
1001100111111	230	0001001101111	135
1101110000111	166	1101000111011	171
0111111011000	123	0111011100101	120
0001110011000	138	0001010101011	134
0101000111000	123	1010000011011	185
0110100110100	122	0100000111010	124
0000111000000	123	1110100000000	129
0001110011011	192	1100111011110	173
0101110000001	124	0100010010000	118
1100111010111	164	0000001100101	125
1010111011111	171	0011100011111	195
0010110000001	142	0000110110110	150
0010110000111	163	0000111110011	166
0000011111011	172	0010111100011	134
0001100011111	197	0000011101010	121

Une comparaison de certains calculs approchés avec les analyses par éléments finis est présentée ci-dessous:

chromosome	résultat EF	résultat Adaline et erreur	résultat réseau à 3 couches
1000000001011	188.3 Hz	171.3 (9.5% d'erreur)	186.5 (1% d'erreur)
1000000011101	196.2 Hz	183.2 (6.8% d'erreur)	194.9 (0.9% d'erreur)

La définition de l'erreur est le quotient $100(\text{résultat EF} - \text{résultat neuronal}) / (\text{moyenne des 2 résultats})$. Dans tous ces calculs, les erreurs commises sont restées du même niveau et toujours dans le même sens pour Adaline, alors que les estimations étaient bien meilleures pour le réseau PMC à trois couches. Ces erreurs n'ont pas été pénalisantes pour l'algorithme génétique (qui, rappelons-le, n'utilise pas de calcul de dérivées), car celui-ci a retrouvé assez rapidement la solution optimale dans les deux cas, c'est-à-dire en une trentaine de générations, mais avec des temps de calcul sans commune mesure avec le calcul complet par éléments finis, puisque sur un PC 486DX, il a fallu une dizaine d'heures pour les calculs complets par éléments finis contre une quinzaine de minutes pour la stratégie proposée.

On peut en déduire les résultats suivants en ce qui concerne le rapport de temps de calcul entre éléments finis et évaluation neuronale (pour cet exemple). Sachant

que dans une optimisation par algorithmes génétiques, ce sont les évaluations de la fonction coût qui utilisent la plus grande partie du temps de calcul, on ne tient compte que des calculs de la fonction coût dans les estimations. Dans le cas où les analyses sont faites complètement par éléments finis, il faut 10 heures de temps de calcul pour 1000 analyses environ, ce qui fait 30 secondes environ par analyse. Dans le cas où l'on utilise le réseau de neurones pour remplacer les calculs par éléments finis, il faut 15 minutes pour encore environ 1000 analyses. Ce temps de 15 minutes ne comprend pas la phase d'apprentissage effectuée pendant la première génération de 50 individus (les calculs étant alors faits par éléments finis). Ensuite, il faut 1 seconde environ par évaluation neuronale. Le rapport de temps de calcul entre évaluation neuronale et éléments finis est donc de 1 sur 30 pour cet exemple.

3.2. Test 2: optimisation de l'amortissement par matériaux viscoélastiques contraints

Pour optimiser l'amortissement et contrôler le niveau de vibrations de structures de type plaques (ou coques), une technique couramment utilisée est celle du matériau viscoélastique contraint, c'est-à-dire la plaque de base plus une couche de matériau viscoélastique mince plus une contre-plaque très mince (figure 7). Il est très important de préciser que ces deux dernières couches sont ici très minces par rapport à celle de la plaque de base. Souvent le matériau viscoélastique et sa contreplaque sont vendus sous forme de rouleaux. A titre d'exemples, cette technique est couramment utilisée sur certaines portières de véhicules automobiles, ainsi que sur des skis. La présence en certains endroits bien choisis de telles zones viscoélastiques contraintes entraîne une dissipation d'énergie par cisaillement qui permet d'avoir un amortissement maximum pour un ou plusieurs modes de vibration définis préalablement. En d'autres termes, pour une structure donnée, on cherche à optimiser la position et la géométrie des zones contraintes, sachant qu'un recouvrement partiel est meilleur qu'un recouvrement total. C'est une technique éprouvée qui s'est déjà montrée efficace pour des problèmes de poutres, plaques ou coques [MAR 94]. La fonction objectif est la maximisation de l'amortissement, soit pour un mode donné, soit pour une combinaison linéaire d'un certain nombre de modes pondérés par leurs énergies respectives (par exemple). Des éléments finis spéciaux pour l'analyse des zones viscoélastiques contraintes sont utilisés. Les amortissements sont calculés par la méthode des énergies modales.

3.2.1. La modélisation éléments finis

On rappelle ci-après brièvement le modèle élément fini particulier conçu pour les plaques [MAR 94], et qui a servi de base aux tests effectués. L'analyse dynamique est effectuée par une méthode modale. Les parties homogènes de la plaque sont discrétisées par des éléments finis conventionnels à continuité C1, et les parties hétérogènes contraintes par des éléments finis spécifiques. Les éléments finis des parties amorties et non amorties doivent être compatibles. Il est très important de rappeler ici que les couches de matériau viscoélastique et de la contreplaque sont très minces par rapport à celle de la plaque de base, ce qui particularise vraiment l'élément fini proposé ici. D'autre part, cet élément fini doit être simple mais performant pour garantir la rapidité et l'efficacité de l'optimisation.

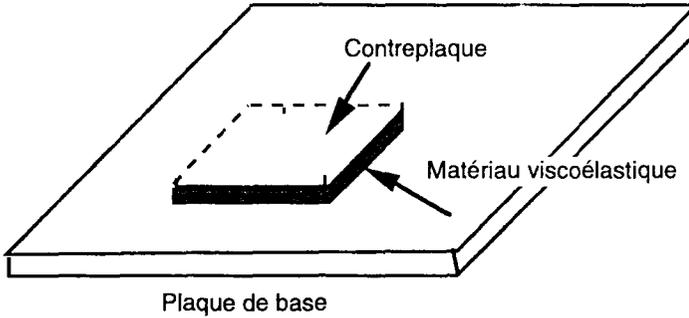


Figure 7. *Plaque de base contrainte*

L'élément fini de plaque homogène est un élément fini classique de plaque en flexion à 4 noeuds obéissant aux hypothèses de Kirchhoff. Les degrés de liberté sont $w, \partial w/\partial x, \partial w/\partial y$ où w est la flèche de la plaque. La fonction d'interpolation est:

$$w = a_1 + a_2x + a_3y + a_4xy + a_5x^2 + a_6y^2 + a_7x^3 + a_8y^3 + a_9x^2y + a_{10}xy^2 + a_{11}x^3y + a_{12}xy^3$$

[1]

Dans la suite les déplacements dans le plan de la plaque principale sont négligés. En fait, c'est la différence de déplacement dans le plan entre la plaque principale et la contreplaque qui importe ici; on prend des déplacements nuls de référence sur la plaque principale; les seuls déplacements dans le plan discrétisés sont ceux de la contreplaque, ou plus exactement cette différence de déplacement dans le plan entre la plaque de base et la contreplaque. Cette hypothèse a été justifiée numériquement dans [MAR 94] et [MAR 95e].

L'élément fini de plaque hétérogène est donc construit à partir des hypothèses suivantes (figure 8) :

-le déplacement transversal w est indépendant de la coordonnée transversale z , $w = w(x,y)$

-la rigidité en traction et flexion de la couche viscoélastique est négligeable, cette couche est seulement soumise aux cisaillements

-les déplacements longitudinaux (figure 8) sont donnés par:

$$\begin{aligned} u_A &= u_3 + h_3(\partial w/\partial x), & u_B &= -h_1(\partial w/\partial x) \\ v_A &= v_3 + h_3(\partial w/\partial y), & v_B &= -h_1(\partial w/\partial y) \end{aligned}$$

[2]

- le matériau viscoélastique est caractérisé par un module de cisaillement complexe G que l'on considère indépendant de la température et de la fréquence.

-les épaisseurs $h_3 \ll h_1$ and $h_2 < h_1$

-le cisaillement transverse dans les couches élastiques est négligé.

-les déplacements longitudinaux u and v de la surface moyenne de la plaque sont pris égaux à 0. La validité de cette hypothèse a été vérifiée sur des exemples numériques.

Cet élément fini a été déjà présenté en détail dans [MAR 94] et [MAR 95e]. On en rappelle simplement ici les grandes lignes pour aider à la compréhension du texte. Les caractéristiques essentielles sont les épaisseurs de la couche viscoélastique et de la contreplaque qui sont très petites vis-à-vis de l'épaisseur de la plaque de base. Les lecteurs intéressés trouvent dans [MAR 94] une discussion et des justifications sur le fait de négliger les déplacements de traction dans la plaque et pas dans la contreplaque, et sur le fait de positionner le plan neutre au milieu de la plaque principale. Des justifications et des comparaisons avec l'expérience ont aussi été fournies dans [MAR 95e].

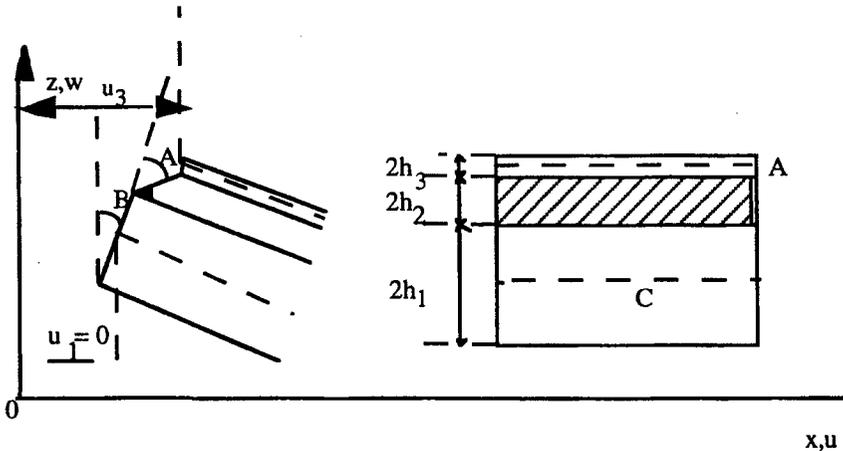


Figure 8. *Elément fini hétérogène*

Les degrés de liberté nodaux sont w , $\partial w/\partial x$, $\partial w/\partial y$ ainsi que les déplacements longitudinaux u_3 et v_3 de la couche contrainte. La fonction d'interpolation pour w est la même que celle de l'élément homogène; les fonctions d'interpolation pour u et v sont celles de l'élément bilinéaire plan. La matrice de raideur élémentaire résulte de la contribution de la flexion, de la traction dans le plan de la couche contrainte et de l'énergie dissipée par cisaillement dans cette couche viscoélastique. Les deux premières contributions sont classiques, la troisième contribution à la matrice de raideur élémentaire est obtenue à partir de l'expression suivante de l'énergie de cisaillement:

$$E = \frac{1}{2} \int_{(A)} \int_{h_1}^{h_1+2h_2} (G\gamma_{xz}^2 + G\gamma_{yz}^2) dz dA \quad [3]$$

avec :

$$\gamma_{xz} = \frac{\partial u_2}{\partial z} + \frac{\partial w}{\partial x} \quad \text{et} \quad \frac{\partial u_2}{\partial z} = \frac{u_A - u_B}{2h_2}$$

donc :

$$\gamma_{xz} = \frac{1}{2h_2} (u_3 + d \frac{\partial w}{\partial x}) \quad \text{avec} \quad d = h_1 + 2h_2 + h_3 \quad [4]$$

de la même manière :

$$\gamma_{yz} = \frac{1}{2h_2} (v_3 + d \frac{\partial w}{\partial y}) \quad [5]$$

On utilise une matrice de masse répartie M qui prend en compte la flexion et la traction des différentes parties. Elle est calculée à partir de l'expression de l'énergie cinétique suivante:

$$T = \frac{1}{2} \int_1 \rho_1 \dot{w}^2 d\tau + \frac{1}{2} \int_2 \rho_2 (\dot{w}^2 + \dot{u}_2^2 + \dot{v}_2^2) d\tau + \frac{1}{2} \int_3 \rho_3 (\dot{w}^2 + \dot{u}_3^2 + \dot{v}_3^2) d\tau \quad [6]$$

La solution en déplacements est cherchée sous la forme périodique: $x = x_0 e^{j\omega t}$. Le module de cisaillement viscoélastique est écrit sous la forme: $G = G_0 (1 + j b)$ et la matrice de raideur correspondante a pour expression:

$$|K_{vj}| = |K_{vjr}| + j |K_{vj}| \quad [7]$$

Le comportement dynamique de la plaque est donné par le système matriciel:

$$(-|M|\omega^2 + |K_1| + j|K_2|)\{u\} = \{0\} \quad [8]$$

où la partie imaginaire K_2 de la matrice de raideur totale est égale à K_{vj} .

Les fréquences et les modes de la structure non amortie peuvent être considérés comme une bonne et simple base modale pour prévoir le comportement dynamique de la structure amortie; ω_j and ϕ_j , $i = 1, n$ sont les fréquences et les modes non-amortis obtenus à partir du système matriciel suivant:

$$(-|M|\omega^2 + |K_1|)\{x\} = \{0\} \quad [9]$$

En faisant le changement de variables classique $\{x\} = |\Phi| \{q\}$, et en prémultipliant par $|\Phi|^t$, on obtient l'équation suivante en vibrations libres:

$$(-|m|\omega^2 + |k_1| + j|k_2|)\{q\} = \{0\} \quad [10]$$

Du fait de l'orthogonalité des modes, $|m|$ and $|k_1|$ sont des matrices diagonales, ce qui n'est pas le cas de $|k_2|$.

Généralement pour les plaques et les poutres les fréquences sont bien séparées, si bien que la matrice $|k_2|$ peut être considérée comme à diagonale dominante. Dans ces conditions, le système modal [10] donne n équations découplées. Il en résulte qu'une bonne approximation du facteur d'amortissement de la structure peut être

facilement déduit de chaque équation du système [10]. Cette approximation a été justifiée par de nombreux auteurs, en particulier dans [JOH 82].

3.2.2. La fonction objectif

L'objectif est la maximisation de l'amortissement d'un mode simple i ou d'une combinaison linéaire de plusieurs modes:

$$\text{Max}(\text{amortissement}) = SE_{V_i} / SE_i \text{ ou: } \sum_j a_j SE_{V_j} / SE_j \quad [11]$$

Dans [11], SE_{V_i} est l'énergie de déformation stockée dans le matériau viscoélastique pour le mode non amorti i et SE_i est l'énergie de déformation élastique de la structure complète, a_j est le coefficient de pondération du mode i (c'est le rapport de l'énergie de déformation du mode i sur la somme des énergies de déformation des modes considérés). Comme toutes les matrices sont diagonales et que l'on conserve les mêmes modes durant le processus d'optimisation, le calcul des facteurs d'amortissement s'en trouve simplifié. Cette procédure est issue de [JOH 82]. L'accord calcul-expérience que l'on a mis en évidence dans [MAR 95c] est suffisant et permet de justifier l'utilisation de cette approximation dans une procédure d'optimisation.

$$\text{amortissement} = \phi_i^T |K_2| \phi_i / \phi_i^T |K_1| \phi_i = k_{2i} / m_i \omega_i^2 \text{ ou } \sum_j a_j \dots \quad [12]$$

Comme les variables du design sont les positions des éléments amortis, le problème d'optimisation est un problème à variables discrètes pour lequel un algorithme génétique est tout à fait adapté.

3.2.3. Un exemple d'optimisation de plaque par éléments viscoélastiques contraints

Cet exemple met en oeuvre une plaque rectangulaire libre-libre dont les dimensions sont les suivantes : longueur $L = 3$ m, largeur $b = 2.5$ m, $h_1 = 0.003$ m, $h_2 = 0.0015$ m, $h_3 = 0.0002$ m. Afin de créer une dissymétrie dans les modes, on met une masse de 500 kg sur la plaque (Figure 9). Le matériau viscoélastique utilisé est du polyéthylène, de coefficient d'amortissement à température ambiante de l'ordre de 2%. La masse volumique du matériau viscoélastique vaut 1050 kg/m^3 . Le module de cisaillement viscoélastique G_0 vaut 500 MPa, l'amortissement propre ou facteur de perte vaut 1 (ces deux dernières quantités sont considérées comme constantes). La plaque et la contre-plaque sont en aluminium. La discrétisation de la plaque a été faite en utilisant 30 éléments finis de plaque. L'optimisation a été effectuée pour le second mode en libre-libre.

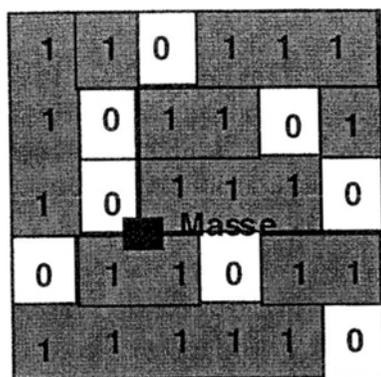


Figure 9. Cas où tous les éléments finis peuvent être couverts (mode 2)

Pour cet exemple, tous les éléments finis peuvent être couverts sachant qu'un recouvrement total donne un amortissement moindre car alors il n'existe pas de zones de discontinuités entre éléments couverts ou non, ces zones étant responsables d'une dissipation accrue d'énergie par cisaillement.

Le problème n'est pas contraint par souci de simplicité, le but essentiel étant la mise en oeuvre des réseaux de neurones. Si l'on envisage la prise en compte des limitations dans les algorithmes génétiques, on trouve dans [MAR 94] le même exemple avec une contrainte sur le nombre maximum d'éléments finis susceptibles d'être couverts; cette contrainte est alors portée par la structure même des chromosomes. On trouve aussi dans [MAR 95c] et [MAR 95d] des exemples de prise en compte d'autres contraintes.

Voici un exemple de chromosome: 101010011101101100011111101010, qui signifie que les éléments 1, 3, 5, 8, 9, 10, 12, 13, 15, 16, 20, 21, 22, 23, 24, 25, 27, 29 sont couverts.

La figure 9 montre le résultat optimal avec analyses complètes par éléments finis quand tous les éléments peuvent être couverts (nombre d'individus de la population 50, probabilité de croisement 0.6, probabilité de mutation 0.03 et 27 générations nécessaires). L'amortissement optimum est de 0.0715. On utilise maintenant exclusivement le réseau de neurones PMC qui donne une bonne approximation, à condition de prendre cette fois-ci 100 vecteurs d'apprentissage (qui sont les résultats des calculs effectués par éléments finis lors des deux premières générations), comme le montre les résultats suivants:

chromosome	amortissement EF	calcul neuronal
111001110101011101111110101110	0.0617	0.0611
111110011011001011101101011111	0.0654	0.0650
100101111101011101111110101110	0.0632	0.0627
101100101101011001111001011110	0.0599	0.0592

Les paramètres d'apprentissage et d'évaluation pour le réseau PMC sont les suivants:

utilisation de 5 couches;
 pas d'apprentissage: 0.005;
 nombre maximum d'itérations pour l'apprentissage: 5000;
 erreur maximale admise: 0.00001 (l'erreur est définie comme la moitié de la somme des carrés des différences entre les activations des neurones de sortie et la sortie désirée de ces mêmes neurones).

L'optimisation par algorithmes génétiques et analyses approchées par réseau de neurones conduit alors à la même solution que dans le cas d'analyses complètes par éléments finis, mais en un temps de calcul sans commune mesure. En effet, pour cet exemple, le rapport de temps de calcul entre éléments finis et évaluation neuronale (sans tenir compte de la phase d'apprentissage) est encore plus intéressant que dans l'exemple précédent, puisqu'il est de 1/100 (les éléments finis sont pénalisés ici par le nombre plus élevé de degrés de liberté).

4. Conclusion et perspectives

Au cours de cette expérimentation numérique sur les réseaux de neurones, ceux-ci ont semblé présenter néanmoins quelques limitations. Ces limitations ne sont pas d'ordre informatique: actuellement, grâce à l'amélioration des performances des ordinateurs, il est possible d'utiliser des réseaux de neurones d'une taille significative (environ 1000 neurones et 1000 poids). C'est plutôt l'absence d'une connaissance théorique établie du fonctionnement des réseaux qui rend leur utilisation délicate. On a été confronté à divers problèmes comme la nécessité d'étudier la faisabilité de chaque application en procédant à de nombreuses expérimentations numériques, comme le dimensionnement aléatoire du réseau, ou encore comme l'absence de théorie pour un calcul prévisionnel d'erreur.

Pour les modélisations de structures mécaniques, il est compréhensible finalement de se demander si l'utilisation d'une méthode toute simple dérivée de la méthode de Rayleigh-Ritz (bien connue en vibrations) ne serait pas plus adaptée aux problèmes traités ici. Précisons cette idée sur un exemple concret en statique. Supposons que l'on veuille optimiser par exemple la localisation et le nombre de raidisseurs sur une plaque donnée avec l'objectif de raidir celle-ci. On peut commencer, comme pour les réseaux de neurones, à évaluer par la méthode des éléments finis un certain nombre de solutions représentatives. Dans une stratégie neuromimétique, ces solutions servent d'ensemble d'apprentissage. Dans une stratégie de type Rayleigh-Ritz, on se sert de ces solutions pour trouver la raideur correspondant à une nouvelle configuration de répartition de raidisseurs sans avoir à refaire de calcul par éléments finis et ceci de la manière suivante: la nouvelle solution est cherchée sous la forme d'une somme pondérée des solutions tests ou représentatives calculées au préalable. Pour trouver les coefficients pondérateurs ou poids (comme dans la méthode des réseaux neuronaux) la mécanique offre une théorie solide: les poids sont obtenus par minimisation de l'énergie potentielle totale de la plaque raidie en question. On peut prendre aussi comme critère la minimisation de l'erreur ou du résidu sur les équations d'équilibres, la méthode proposée s'apparentant alors davantage à une méthode de type résidu pondéré de Galerkin. On constate une certaine similitude de ces stratégies avec les réseaux de neurones, la différence étant que la mécanique fournit un critère d'erreur rigoureux.

Par contre les réseaux de neurones ont l'avantage d'avoir des facultés d'adaptation meilleures. Rien n'empêche d'ailleurs de faire fonctionner un réseau de neurones en utilisant les critères d'erreur de la mécanique pour optimiser et contrôler les poids. On se propose de tester ces stratégies dans un proche avenir. Ce premier article ne constitue en fait qu'une première étude de faisabilité.

Néanmoins cet article est consacré aux réseaux de neurones qui présentent un certain nombre de qualités intrinsèques. Ces qualités peuvent éventuellement les rendre supérieurs aux méthodes conventionnelles de la mécanique exposées plus haut. La première qualité qui vient à l'esprit est le parallélisme: les réseaux sont formés d'unités élémentaires qui peuvent calculer simultanément (c'est une des raisons supposées de la supériorité du cerveau). D'autre part, ils sont capables de s'adapter. Enfin, ils peuvent résoudre l'imprécis, reconnaître le flou, et donc faire preuve d'une grande robustesse.

BIBLIOGRAPHIE

- [BER 92] BERKE L., HAJELA P. "Applications of artificial neural nets in structural mechanics". *Structural Optimization, Vol 4*, 1992, p. 90-98.
- [GOL 89] GOLDBERG D. E. *Genetic Algorithm in Search, Optimization and Machine Learning* Addison - Wesley, 1989.
- [HAJ 94] HAJELA P., SZEWCZYK Z. "Neurocomputing strategies in structural design - analysing weights of feedforward neural networks". *Structural Optimization, Vol 8*, 1994, p. 236-241.
- [HOL 92] HOLLAND J. " Les algorithmes génétiques " *Pour la science*, N° 179 Sept 1992.
- [JOD 94] JODOUIN J.F. *Les réseaux neuromimétiques* Hermès, 1994.
- [JOH 82] JOHNSON C.D., KIENHOLZ D.A "Finite element prediction of damping in structures with constrained viscoelastic layers". *AIAA J.*, n° 20, 1982, p. 1284-1290.
- [MAR 94] MARCELIN J.L., TROMPETTE P. "Optimal location of plate damped parts by use of a genetic algorithm". *Shock and Vibration, Vol. 1*, n° 6, 1994, p. 541-547.
- [MAR 95a] MARCELIN J.L. "CAO d'engrenages par algorithmes génétiques", *Revue Internationale de CFAO et d'Informatique Graphique, Vol 10*, n°5, 1995, p.485-498
- [MAR 95b] MARCELIN J.L. "Optimisation des conditions aux frontières par algorithmes génétiques", *Revue Européenne des Elements Finis, Vol. 4*, n° 3, 1995, p. 361-373.
- [MAR 95c] MARCELIN J.L., TROMPETTE P., DORNBERGER R. "Optimisation of Composite Beam Structures Using a Genetic Algorithm". *Structural Optimization, Vol. 9*, 1995, p. 236-244.
- [MAR 95d] MARCELIN J.L., TROMPETTE P., DORNBERGER R. "Optimal Structural Damping of Skis Using a Genetic Algorithm". *Structural Optimization, Vol. 10*, 1995, p. 67-70.

[MAR 95e] MARCELIN J.L., SHAKHESI S., POURROY F. "Optimal constrained layer damping of beams: experimental and numerical studies". *Shock and Vibration*, Vol. 2, n° 6, 1995, p. 445-450.

[SZE 93] SZEWCZYK Z., HAJELA P. "Neural network approximations in a simulated annealing based optimal structural design". *Structural Optimization*, Vol 5, 1993, p. 159-165.

[SZE 94] SZEWCZYK Z., HAJELA P. "Neurocomputing strategies in structural design-decomposition based optimization". *Structural Optimization*, Vol 8, 1994, p. 242-250.

Article soumis le 23 février 1996.
Version révisée le 30 septembre 1996.