

SlownessBuddy

Courbes de lenteur polaires :

SAW et PSAW sur surface homogène et dans un réseau périodique

(Calcul du beam-steering et du paramètre de diffraction)

(calcul de l'atténuation et du couplage)

Vincent Laude

12 juillet 2000

Ce papier résume la procédure employée pour calculer le beam-steering et le paramètre de diffraction à partir d'une courbe de lenteur polaire. On décrit également comment les courbes de lenteur des ondes de surface pour les conditions aux limites surface libre et surface métallisée, ainsi que dans un réseau, sont obtenues.

Le but ultime est de disposer d'un outil logiciel (SlownessBuddy) qui permette de tracer les courbes de lenteur et d'en déduire les grandeurs de diffraction importantes dans les cas :

1. Ondes de volume ;
2. SAW sur surface homogène ;
3. PSAW sur surface homogène ;
4. SAW/PSAW dans réseau périodique sans tenir compte de la surcharge massique (Blotekjaer) ;
5. SAW/PSAW dans réseau périodique en tenant compte de la surcharge massique (FEM/BEM).

1. Calcul des courbes de lenteurs polaires

L'objet du programme est de tracer des courbes de lenteur, c'est-à-dire des fonctions du type $s(\vartheta)$, où ϑ est l'angle de propagation dans le plan du substrat. On discute dans la suite de cette section l'existence des différentes ondes, ainsi que les méthodes pratiques employées pour les localiser.

• Ondes de volume

Les ondes de volume, au nombre de 3, sont déterminées par une méthode aux valeurs propres très efficace (une fraction de seconde suffit pour générer des milliers de points). Elles existent toujours quel que soit l'angle.

Les 3 courbes de lenteur de volume sont sauveées dans 3 fichiers appelés **volume1.dat**, **volume2.dat** et **volume3.dat**. Les courbes sont classées en assurant au mieux la continuité de la courbe et de sa dérivée (plutôt que celle du vecteur de Poynting comme dans la thèse de L. Boyer). Dans certains cas pathologiques et pour un nombre de points insuffisant (par exemple quartz (YZt)/5, cf. thèse de L. Boyer p. 104), la classification peut échouer, mais c'est rare.

Le beam-steering et le paramètre de diffraction sont calculés directement à partir de la courbe de lenteur.

• SAW sur surface homogène

Deux courbes de lenteur sont sauveées sous les noms **sawsl.dat** et **sawsm.dat** (sl et sm représentent respectivement la surface libre et métallisée).

Les lenteurs sont déterminées en recherchant les zéros des déterminants donnés dans la thèse de Laurent Boyer (équations (4.11) et (4.12), page 114). Cette méthode est équivalente à trouver les

zéros et les pôles de la permittivité effective, mais elle est plus élégante. La fonction de recherche de zéro employée est DZANLY de IMSL ; cette fonction est très performante si le zéro existe. La recherche se fait par continuité à partir de la valeur précédente, d'où l'importance de la valeur initiale.

A noter qu'on ne cherche qu'un seul zéro à chaque fois. Peut-il exister plus d'une SAW à la fois ? Je n'en sais rien, mais je ne l'ai jamais observé.

A noter également que pour certaines coupes, la SAW surface libre n'existe pas pour tous les angles (du fait de la présence d'une SSBW qui peut être plus lente que la SAW). Dans le cas surface métallisée, rien de tel n'a été observé. Pour résoudre ce problème, on minimise d'abord le module du déterminant (fonction DUVMI de IMSL) ; si la valeur trouvée est proche de 0 (en pratique inférieure à un seuil arbitraire est fixé à 1.d-4), on considère qu'un zéro existe ; sinon on passe à l'angle suivant. Cette méthode est peu élégante mais efficace. De plus, dans le cas où la lenteur initiale est choisie comme étant la plus lente des SSBW, on calcule la lenteur de l'onde de volume pour l'angle initial, puis on recherche la lenteur projetée maximale des ondes de volume dans le plan sagittal. Cette méthode permet dans certains cas de sortir d'un minimum local causé par une SSBW placée avant la SAW, mais plus lente que l'onde de volume pour l'angle considéré. Dans le cas où toute la courbe de lenteur n'est pas obtenue, il est toujours possible de fixer manuellement la lenteur initiale, ou simplement de changer l'angle initial (les SSBW gênantes sont généralement très localisées autour de symétries particulières observables sur les ondes de volume).

• **PSAW sur surface homogène**

Les PSAW sont déterminées par le minimum du module des déterminants donnés dans la thèse de Laurent Boyer (équations (4.11) et (4.12), page 114). Cette méthode est nettement plus stable que la précédente, puisque le minimum d'une fonction continue et deux fois dérivable dans un intervalle donné existe toujours, tandis qu'un zéro pas nécessairement. En revanche, la détermination d'un zéro quand il existe est beaucoup plus précise que celle d'un minimum de la valeur absolue de la fonction. Ces détails numériques peuvent avoir des répercussions sur la qualité de l'estimation du beam-steering et du paramètre de diffraction.

Les courbes de lenteurs sont sauvées sous les noms **psawsl.dat** et **psawsm.dat** (sl et sm représentent respectivement la surface libre et métallisée).

• **SAW/PSAW dans réseau périodique (Blotekjaer)**

Pour ce cas, la lenteur est déterminée par fit de l'admittance harmonique. Le calcul de l'admittance harmonique est réalisé par la méthode de Blotekjaer telle que programmée par P. Ventura. Le calcul a été étendu au cas d'un angle de propagation non nul (le réseau est périodique dans la direction $\theta=0$).

La méthode de fit utilisée emploie la fonction DUNLSJ de IMSL (algorithme de moindres carrés non-linéaire). En pratique, on calcule un certain nombre de points (e.g. 32) de l'AH dans un voisinage donné (supposé proche de la lenteur du pôle), et on minimise la distance au modèle théorique suivant :

$$Y(y) = c_1 \sin(|\pi y|) + c_2 \frac{\sin^2(\pi y)}{\sin^2(\pi y) - c_3} \quad (1)$$

Ce modèle est une expression théorique des contributions à l'AH des termes diélectriques et de la SAW/PSAW. *On pourra y introduire un modèle de contribution d'une SSBW une fois celui-ci obtenu.* Les trois paramètres sont supposés complexes *a priori*. En particulier, la partie imaginaire de c_3 n'apparaît théoriquement qu'en cas d'atténuation (donc de PSAW ou SSBW). La lenteur de l'onde est alors estimée par résolution de l'équation implicite (DZANLY de IMSL) :

$$\sin^2(\pi f p s) = c_3 \quad (2)$$

La partie réelle de la solution donne la lenteur, tandis que la partie imaginaire est liée à l'atténuation de l'onde. L'atténuation par longueur d'onde, exprimée en décibels par longueur d'onde, est donnée par (thèse LB, page 132) :

$$\alpha = -20 \log_{10}(e) 2\pi \frac{|\Im(s)|}{\Re(s)} \quad (3)$$

Estimation du couplage. Le courant généré dans les électrodes dans un « vrai » transducteur est obtenu pour $\gamma = 1/2$, et est égal à

$$Y(1/2) = c_1 + \frac{c_2}{1 - c_3} \quad (4)$$

On définit un couplage par le rapport :

$$\Gamma = \left| \frac{c_2}{c_1} \right| \quad (5)$$

La courbe de lenteur est sauvee sous le nom **blotek.dat**.

• SAW/PSAW dans réseau périodique (FEM/BEM)

Un élément fini spécifique, intégré à MODULEF a été développé spécialement. Cet élément est intégré dans la bibliothèque ELAS (projet elas2), sous le nom TRIA 2K2D, et en décrit en section 7. Le calcul de l'admittance harmonique suit la procédure classique de TMX définie par P. Ventura : définition du maillage par CRENOPO, calcul des matrices et seconds membres élémentaires par les exécutables de Modulef (APNOXX, COMAXX, FOMIXX, THEZXX, ASSMXX), calcul de l'impédance mécanique de l'électrode par une version complexifiée de IMPDGEN (PV rév. SB), et enfin calcul de l'admittance harmonique par PolarFemBem (dérivé de PERGEN par PV rév. SB). L'estimation des paramètres est identique au cas Blotekjaer.

En pratique, SlownessBuddy se charge de lancer les exécutables Modulef à mesure des besoins. PolarFemBem peut être lancé directement dans une fenêtre DOS, il prend alors ses données d'entrée du fichier SBPrefs.dat maintenu par SlownessBuddy.

Il faut noter que seules les versions des exécutables de Modulef fournies avec SlownessBuddy connaissent le nouvel élément fini TRIA 2K2D, et qu'il faut donc bien veiller à utiliser ces versions. Il en est de même pour IMPDGEN. De plus, de nombreux fichiers doivent être présents pour que la partie FEM/BEM fonctionne. Tous ces fichiers et programmes sont normalement fournis dans l'installation.

La courbe de lenteur est sauvee sous le nom **fembem.dat**.

	Lenteur (10^{-4} s/m)	Atténuation (dB/λ)	Couplage (s.u.)	Beam-steering (°) et γ
Volume	Equation de Christofel aux valeurs propres	0.d0	n.a.	Méthode 1 (1 point)
SAW, CL homogènes	Zéros des déterminants	0.d0	$K^2 = \frac{s_m^2 - s_l^2}{2s_m s_l}$	Méthode 2 (3 points)
PSAW, CL homogènes	Minimums des déterminants	???	$K^2 = \frac{s_m^2 - s_l^2}{2s_m s_l}$ (?)	Méthode 2 (3 points)
SAW/PSAW, réseau Blotekjaer	Fit AH (c_3)	Fit AH (c_3)	Fit AH $\left \frac{c_2}{c_1} \right $	Méthode 1 (1 point)
SAW/PSAW, FEM/BEM	Fit AH (c_3)	Fit AH (c_3)	Fit AH $\left \frac{c_2}{c_1} \right $	Méthode 1 (1 point)

Tableau 1 : résumé des méthodes employées dans les différents cas. En grisé les cas programmés, en blanc les cas en attente.

Volume	volume1.dat, volume2.dat, volume3.dat
SAW, CL homogènes	sawsl.dat, sawsm.dat
PSAW, CL homogènes	psawsl.dat, psawsm.dat
SAW/PSAW, Blotekjaer	blotek.dat
SAW/PSAW, FEM/BEM	fembem.dat

Tableau 2 : Noms des fichiers texte dans lesquels sont sauveés les courbes de lenteur.

2. Calcul du beam-steering

On suppose la courbe de lenteur obtenue, c'est-à-dire qu'on dispose d'une fonction strictement positive $s(\theta)$ en coordonnées polaires, l'angle θ étant exprimé en radians. Les coordonnées du point courant sont alors

$$\begin{pmatrix} x(\vartheta) \\ y(\vartheta) \end{pmatrix} = \begin{pmatrix} s(\vartheta) \cos(\vartheta) \\ s(\vartheta) \sin(\vartheta) \end{pmatrix} \quad (6)$$

d'où l'on déduit la tangente à la courbe

$$\begin{pmatrix} x'(\vartheta) \\ y'(\vartheta) \end{pmatrix} = \begin{pmatrix} s'(\vartheta) \cos(\vartheta) - s(\vartheta) \sin(\vartheta) \\ s'(\vartheta) \sin(\vartheta) + s(\vartheta) \cos(\vartheta) \end{pmatrix} \quad (7)$$

On définit l'angle de beam-steering ψ à partir de la normale à la courbe de lenteur par

$$\vec{n}(\vartheta) = \begin{pmatrix} \cos(\vartheta + \psi) \\ \sin(\vartheta + \psi) \end{pmatrix} = \begin{pmatrix} \cos \psi \cos \vartheta - \sin \psi \sin \vartheta \\ \sin \psi \cos \vartheta + \cos \psi \sin \vartheta \end{pmatrix} \quad (8)$$

Ce vecteur devant être orthogonal à la tangente à la courbe, on doit avoir

$$\begin{pmatrix} \cos \psi \\ \sin \psi \end{pmatrix} = \alpha \begin{pmatrix} s(\vartheta) \\ -s'(\vartheta) \end{pmatrix} \quad (9)$$

avec

$$\alpha = \frac{1}{\sqrt{s(\vartheta)^2 + s'(\vartheta)^2}} \quad (10)$$

Au final, en posant

$$f(\vartheta) = \frac{d}{d\vartheta} \ln s(\vartheta) = \frac{s'(\vartheta)}{s(\vartheta)} \quad (11)$$

et

$$d(\vartheta) = \sqrt{1 + f^2(\vartheta)} \quad (12)$$

on obtient

$$\begin{pmatrix} \cos \psi \\ \sin \psi \end{pmatrix} = \frac{1}{d(\vartheta)} \begin{pmatrix} 1 \\ -f(\vartheta) \end{pmatrix} \quad (13)$$

Cette équation permet d'obtenir l'angle ψ modulo 2π par la fonction DATAN2 intrinsèque du Fortran. Cette fonction est à préférer à DATAN qui renvoie un résultat modulo π .

3. Calcul du paramètre de diffraction γ

Par définition, le paramètre de diffraction est

$$\gamma(\vartheta) = \frac{d\psi(\vartheta)}{d\vartheta} \quad (14)$$

L'angle ψ étant obtenu en dérivant la courbe de lenteur, γ sera obtenu comme une dérivée seconde. Par dérivation de l'équation (8), on a

$$\begin{pmatrix} \psi' \sin \psi \\ \psi' \cos \psi \end{pmatrix} = \begin{pmatrix} f' f / d^3 \\ -f' / d + f' f^2 / d^3 \end{pmatrix} \quad (15)$$

et la combinaison des équations (8) et (10) donne

$$\gamma(\vartheta) = -f' / d^2 \quad (16)$$

soit au final

$$\gamma(\vartheta) = \frac{-f''(\vartheta)}{1 + f^2(\vartheta)} \quad (17)$$

4. Considérations numériques pour le calcul des dérivées

Il est nécessaire de calculer les dérivées première et seconde de la lenteur polaire. Pour cela, nous calculons le logarithme de la lenteur pour trois angles $\vartheta - \epsilon$, ϑ et $\vartheta + \epsilon$. On a alors par différence finie

$$f(\vartheta) = \frac{\ln s(\vartheta + \epsilon) - \ln s(\vartheta - \epsilon)}{2\epsilon} \quad (18)$$

et

$$f'(\vartheta) = \frac{\ln s(\vartheta + \epsilon) - 2 \ln s(\vartheta) + \ln s(\vartheta - \epsilon)}{\epsilon^2} \quad (19)$$

En pratique, nous avons deux possibilités pour appliquer les formules ci-dessus :

- Méthode 1 : prendre ϵ comme l'espacement entre deux points de la courbe de lenteur (les calculs sont faits sur N+2 points et sauvés sur N points pour tenir compte des effets de bord).

- Méthode 2 : ajouter deux points de calcul autour de chaque point désiré de la courbe de lenteur ($\vartheta + \epsilon$ et $\vartheta - \epsilon$). Il faut choisir ϵ comme compromis entre résolution (petite valeur) et bruit de calcul (grande valeur). 1 mrad semble un bon compromis. Le défaut de cette méthode de calcul des dérivées est qu'il faut trois fois plus de points de calcul que de points d'affichage.

On aurait aussi pu employer une méthode de FFT pour calculer les dérivées, comme suggéré par JD. Cependant, les deux méthodes élémentaires utilisées permettent d'obtenir les paramètres de diffraction pour une direction quelconque, ce que ne permet pas une méthode de calcul par FFT. De plus, une méthode de type FFT requiert que la courbe de lenteur soit périodique, donc définie de 0 à 2π , ce qui n'est pas toujours le cas (et est impossible pour un réseau).

Pour éviter une dérivation, on pourrait également calculer la direction du vecteur de Poynting, ce qui donnerait directement l'angle de beam-steering. Étant donnée la précision obtenue par la méthode de localisation dans le cas des SAW, cette méthode n'a pas été employée ; elle pourra l'être si la courbe de lenteur est connue moins précisément. A noter cependant que pour la propagation dans un réseau, le vecteur de Poynting calculé pour le substrat seul ne représente pas la direction de propagation de l'énergie.

5. Matrice mixte pour la propagation penchée

Pour une onde se propageant avec un angle de phase ϑ et une lenteur s , tous les champs ont une dépendance de la forme :

$$\exp(-j\omega s(\pm x_1 \cos \vartheta + x_3 \sin \vartheta)) \quad (20)$$

(+ pour une onde incidente et - pour une onde réfléchie). La matrice mixte pour la propagation penchée ($\vartheta \neq 0$) est identique à la matrice mixte classique si on considère qu'elle relie les amplitudes d'ondes de la forme (20). Il suffit donc de considérer que ces termes sont explicitement des fonctions de l'angle de phase ϑ .

Pour obtenir la forme analytique de la contribution de l'onde de surface à l'admittance harmonique (1), on suppose appliqué un potentiel harmonique, et on écrit que les ondes entrantes et sortantes sont identiques à la phase de propagation près (ce qui correspond en d'autres mots au théorème de Floquet). Le potentiel harmonique doit être ici de la forme :

$$V(n, x_3) = \exp(-2j\pi \gamma n) \exp(-j\omega s x_3 \sin \vartheta) \quad (21)$$

pour la n-ième électrode. En effet, la périodicité imposée par le réseau ne s'applique que suivant x_1 . Le terme en exponentielle de la variable x_3 se simplifie, et on retrouve la forme (1) pour la contribution de l'onde de surface à l'admittance harmonique.

Cependant, lors du fit de l'AH à son modèle théorique, il faut bien veiller à travailler à ϑ constant. Il faut échantillonner l'AH en lenteur s , et en déduire :

$$\begin{aligned} \gamma &= f p s \cos \vartheta \\ s_1 &= s \cos \vartheta \\ s_3 &= s \sin \vartheta \end{aligned} \quad (22)$$

Le fit de l'AH donne alors $\sin^2(\pi \hat{\gamma}) = c_3$, d'où l'on tire la lenteur de l'onde par $\hat{s} = \frac{\hat{\gamma}}{f p \cos \vartheta}$.

Dit autrement, il ne faut pas travailler à $k_3 = \omega s \sin \vartheta$ constant comme on pourrait le penser naïvement, mais à ϑ constant.

6. Blotekjaer pour la propagation penchée

La théorie de Blotekjaer pour le calcul de l'admittance harmonique doit être légèrement modifiée pour s'appliquer au cas de la propagation penchée dans un réseau. Par souci de simplicité, nous reprenons dans l'ordre original les équations de l'annexe A du rapport DRET TS.ASM

Notations préliminaires :

$$s = \begin{pmatrix} s_1 = s \cos \vartheta = \gamma / fp \\ s_3 = s \sin \vartheta \end{pmatrix}, \quad s_{ln} = (n + \gamma) / fp = s_1 + n / fp \quad \text{et} \quad s_n^2 = (s_1 + n / fp)^2 + s_3^2 \quad (23)$$

Reprise des calculs de PV :

$$\Phi_s(\gamma, x_1, x_3) = \exp(-2j\pi \gamma x_1 / p) \exp(-2j\pi f s_3 x_3) \sum_{m=-\infty}^{\infty} \Phi_{sm}(\gamma, s_3) \exp(-2j\pi m x_1 / p) \quad (A2)$$

$$\sum_{n=-\infty}^{\infty} (n + \gamma) \Phi_{sm}(\gamma, s_3) \exp(-2j\pi m x_1 / p) = 0 \quad ; \quad |x_1 - Np| < b/2, N = E(x_1 / p) \quad (A3)$$

$$\sum_{n=-\infty}^{\infty} D_{sm}(\gamma, s_3) \exp(-2j\pi m x_1 / p) = 0 \quad ; \quad |x_1 - Np| > b/2, N = E(x_1 / p) \quad (A4)$$

$$(n + \gamma) \Phi_{sm}(\gamma, s_3) = \sum_{q=N}^M S_{n-q} P_{n-q} \alpha_q \quad (A5)$$

$$D_{sm}(\gamma, s_3) = \sum_{q=N}^M P_{n-q} \beta_q$$

$$D_{sn} = \omega |s_n| \epsilon_n \Phi_{sn}$$

$$\epsilon_n = \epsilon_V - \frac{j}{|s_n| G_{n44}} = \epsilon_V - \frac{j \epsilon_n^M}{|s_n| \epsilon_n^L} \quad (A9)$$

(Noter l'inversion des déterminants surface libre et métallisée dans l'équation originale)

$$\text{pour } n > N_2, n < N_1, \quad \epsilon_n(s_{ln}, s_3) \rightarrow \epsilon_{\infty} = \epsilon(\infty, 0) \quad (A10)$$

$$\text{pour } n > N_2, n < N_1, \quad \sum_{q=N}^M P_{n-q} \beta_q = \frac{2\pi \epsilon_{\infty}}{p} S_n \sum_{q=N}^M S_{n-q} P_{n-q} \alpha_q \quad (A11)$$

$$\text{pour } N_1 \leq n \leq N_2, \quad D_{sm} = \frac{2\pi \epsilon_{\infty}}{p} \sum_{q=N_1}^{N_2+1} P_{n-q} \alpha_q = \frac{2\pi |s_n|}{p s_{ln}} \epsilon_n \sum_{q=N_1}^{N_2+1} S_{n-q} P_{n-q} \alpha_q \quad (A11')$$

$$B_{n-q} = \left\{ S_{n-q} \epsilon_n^M - j \epsilon_n^L s_{ln} \left(\epsilon_{\infty} - \frac{|s_n|}{s_{ln}} \epsilon_V S_n S_{n-q} \right) \right\} P_{n-q} \quad (A13)$$

Au final, la principale modification est le facteur additionnel $|s_n| / s_{ln}$, qui est surtout important si $n \simeq 0$ et s_3 n'est pas trop petit. La fonction de green est calculée pour l'argument (s_{ln}, s_3) .

7. Élément fini pour la propagation penchée (TRIA 2K2D)

Cet élément fini est fondé sur TRIA 2Z2D écrit par S. Ballandras. 2Z2D est un élément à 3 degrés de liberté (les trois composantes du déplacement) mais 2 dimensions (les coordonnées x_1 et x_2), utilisant des polynômes d'interpolation isoparamétriques. Pour 2Z2D, tous les calculs dans Modulef sont faits en réel. 2K2D est identique à ceci près que l'on suppose une dépendance harmonique pour la troisième dimension, ce qui se traduit par (en reprenant les notations de la documentation HTML de Modulef, FR/Guide7-14/node13.html) :

$$\mathbf{u} = [P(x_1, x_2)] \{u_T\} \exp(-jk_3 x_3) \quad (24)$$

En conséquence, les calculs doivent être faits en complexe, et les dérivées par rapport à la troisième dimension prises en compte. Cela se traduit par :

$$\{\epsilon\} = \left[[D][J^{-1}][DP] - j k_3 [D_3][P] \right] \{u_T\} \quad (25)$$

où [P] est la matrice des polynômes, [DP] la matrice des dérivées des polynômes, [J⁻¹] le Jacobien inverse,

$$[D] = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix} \quad \text{et} \quad [D_3] = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix} \quad (26)$$

Ce qui conduit à :

- Matrice élémentaire de masse (inchangée)

$$[M_T] = \int_T {}^t[P] \rho [P] d\mathbf{x} \quad (27)$$

- Matrice élémentaire de rigidité

$$[K_T] = \int_T {}^h([D][J^{-1}][DP] - j k_3 [D_3][P])[E]([D][J^{-1}][DP] - j k_3 [D_3][P]) d\mathbf{x} \quad (28)$$

(^h) représente la symétrie hermitienne)

- Second membre élémentaire (inchangé)

$$[B_T] = \int_T {}^t[P]\{f^\Omega\} d\mathbf{x} + \int_{\partial T \cap \Gamma_0} {}^t[P]\{f^\Gamma\} d\Gamma \quad (29)$$

- Contraintes élémentaires

$$[\sigma_T] = [E]([D][J^{-1}][DP] - j k_3 [D_3][P]) \quad (30)$$

Il faut noter que le fait de tenir compte de k_3 complique les calculs d'éléments finis. En effet, à la différence de la fréquence qui n'intervient que pour factoriser les matrices de masse et de raideur, k_3 intervient pour chaque calcul de la raideur.

L'inclusion de 2K2D a nécessité de nombreuses (petites) modifications dans le code source de Modulef :

- 9 fonctions spécialisées pour l'élément, plus le « câblage » par GOTO de ces fonctions...
- Version de l'algorithme de Crout (algorithme LU pour les matrices symétriques) pour les matrices hermitiennes (A noter que l'algorithme de Crout implémenté dans Modulef est très basique, et n'effectue aucune recherche de pivot, ce qui est formellement déconseillé dans la littérature...);
- Différentes variations sur le thème du calcul du produit de matrices complexes...

Du fait de la structure antédiluvienne de Modulef (Fortran 77), il est nécessaire de toujours faire les calculs sur le super tableau (et pourtant l'allocation dynamique existe depuis 30 ans au moins maintenant...), le passage des paramètres principaux se fait par des communs obscurs et aucune vérification du type des variables n'est assurée. De plus il est difficile de se servir d'algorithmes tout faits comme ceux d'IMSL (toujours le super tableau...). Il serait extrêmement bénéfique de passer à une librairie d'éléments finis en langage objet (C++ par exemple, ou Fortran90 objet ?) puisque l'essentiel du travail fait par Modulef est de décliner les mêmes fonctions de base pour tous les éléments finis et tous les types de données existants. Le problème est de trouver une bibliothèque aussi riche en éléments finis, au moins ceux utilisés à TMX et au LPMO. Un premier pas serait déjà une version Fortran90 de Modulef...