

REPUBLIQUE ALGERIENNE DEMOCRATIQUE
ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT
SUPERIEUR ET DE LA RECHERCHE
SCIENTIFIQUE



Université Mohamed Boudiaf de M'sila
Faculté des Mathématiques et de l'Informatique
Département des Mathématiques

Mémoire de Master

Domaine : Mathématiques et Informatique

Filière : Mathématiques

Option : EDP et applications

Thème

*Résolution numérique d'un problème inverse de conduction thermique par la
méthode de Levenberg-Marquardt*

Présentée par :

M^{elles} ZIANE Imane et CHARIKH Siham

Soutenu publiquement le : 28/05/2017.

Devant le jury composé de :

Président : *M^r* MERZOUGUI Abdelkrim

M.C.A, Université de M'sila

Encadreur : *M^r* NOURI Brahim

M.C.B, Université de M'sila

Examineur : *M^r* BLIZAK Tahar

M.A.A, Université de M'sila

Année universitaire 2016/2017

Remerciements

Nous remercions avant tous ALLAH pour son aide, ses innombrables dons, ALLAH qui nos a donné la force, la volonté et le moral pour accomplir nos études en master en mathématique.

*Ainsi, nous tenons également à exprimer nos vifs remerciements à notre encadreur Mr **NOURI Brahim** pour avoir d'abord proposer ce thème, pour son suivi continué tout*

le long de la réalisation de ce mémoire et quelle n'a pas cessée de vous donner ses conseils.

Notre remerciements vont au président du jury et aux membres du jury qui nos ont fait l'honneur de participer au jury.

Nous remercions évidemment nos parents, nos frères et soeurs, qui depuis de si longues années, nos ont encouragé et soutenu dans la poursuite de nos études.

Enfin, Nous tenons à exprimer notre reconnaissance à tous nos amis et collègues pour le soutien moral . . .

A tous MERCI

Dédicaces

Je dédie ce modeste travail
A ce qui le plus cher, à ma mère ce qui était un bon soutien pour moi
à mon père qui m'a soutenu et m'aide pour arriver d'ici
Et sans oublier ma petite sœur
Toute ma famille.
et à tous ceux qui se fatiguent pour réussir leur avenir

C.Siham

Je dédie ce mémoire
A mes très chers parents, pour leur amour inestimable, leur confiance, leur soutien qui ont
consenti d'énormes
sacrifices pour mon éducation et mon bien être.
A mes sœurs et mes frères A toute ma famille
A tout mes amis.
A tous mes collègues de la promotion 2017

Z.Imane

Résumé

ملخص: في هذه المذكرة، قمنا باقتراح خوارزمية تعتمد على طريقتي لافنبارغ-ماركار والعناصر المنتهية لحساب معامل في مسألة غير خطية لمعادلة الحرارة في مجال غير متجانس مع شرط ابتدائي وشرطين حديين لدريكلي ونيومان. المعامل تم تقريبه بواسطة كثير حدود وهذا لتقوم الخوارزمية بحساب معاملاته.

كلمات مفتاحية: تربيعات مصغرة غير خطية، عناصر منتهية، طريقة لافنبارغ-ماركار، مسألة عكسية.

Dans ce mémoire, nous avons proposé un algorithme basé sur la méthode de Levenberg-Marquardt et la méthode des éléments finis pour estimer un coefficient dans un problème non linéaire gouverné par une équation de la chaleur dans un milieu non homogène en dimension un avec condition initiale et des conditions aux limites de Dirichlet-Neumann. Le coefficient est rapproché par une forme polynomiale et l'algorithme numérique est employé pour trouver les coefficients de cet polynôme.

Mots-Clés : Éléments finis, Méthode de Levenberg-Marquardt, Moindres carrés non linéaires, Problème inverse.

In this memoir, we proposed an algorithm based on the Levenberg-Marquardt method and the finite element method to estimate a coefficient in a nonlinear problem governed by an equation of heat in a non-homogeneous one-dimensional domain with data condition and boundary conditions of Dirichlet-Neumann. The coefficient is approximated by a polynomial form and the numerical algorithm is used to find the coefficients of this polynomial.

Keywords : Finite Elements, Levenberg-Marquardt method, Nonlinear Least squares, Inverse problem.

Table des matières

1	Problème des moindres carrés non linéaires	1
1.1	Méthode de Gauss-Newton	4
1.2	Méthode de Levenberg-Marquardt	4
2	Problème direct : Problème de conduction thermique	6
2.1	Position du problème	7
2.1.1	Formulation variationnelle	7
2.2	Semi-discrétisation par éléments finis	9
2.3	Implémentation numérique	10
2.3.1	Vecteur de terme source	10
2.3.2	Matrice de rigidité	10
2.3.3	Matrice de masse	11
2.3.4	Méthode d'Euler implicite	11
2.3.5	Méthode de Cholesky	11
3	Méthode de Levenberg-Marquardt pour un problème inverse de conduction thermique	13
3.1	Problème inverse	14
3.2	Schéma itératif	15
3.3	Critère d'arrêt	16
3.4	Algorithme de calcul	17
3.5	Calcul des coefficients de la matrice jacobienne	18
3.5.1	Formulation variationnelle	18
3.5.2	Semi discrétisation par éléments finis	19
3.5.3	Méthode Grank-Nicolson	21

Introduction générale

Un problème inverse est une situation dans laquelle les valeurs de certains paramètres d'un modèle doivent être identifiées à partir d'observations du phénomène. En mathématiques, un problème inverse a la forme d'une équation

$$Ax = y \tag{1}$$

Où y représente les mesures effectuées, x représente les valeurs des paramètres du phénomène et A est un opérateur linéaire ou non linéaire.

Les problèmes inverses généralement sont des problèmes mal posés car si l'on cherche à résoudre l'équation (1); cela nécessite l'inversion de l'opérateur A . Cette opération n'est pas forcément évidente d'un point de vue numérique et d'après Hadamard [2] un problème est bien posé s'il vérifie les trois conditions suivantes :

- ☞ La solution existe ;
- ☞ Elle est unique ;
- ☞ Elle dépend continument des données.

Donc, si l'une des trois conditions n'est pas satisfaite, on dit que le problème est mal posé.

La résolution du problème inverse passe en général par une étape initiale de modélisation du phénomène, dite problème direct qui décrit comment les paramètres du modèle se traduisent en effets observables expérimentalement. Ensuite, à partir des mesures obtenues sur le phénomène réel, la démarche va consister à approximer au mieux les paramètres qui permettent de rendre compte de ces mesures. Cette résolution peut se faire par *simulation numérique* ou de façon analytique.

Dans ce mémoire, on s'intéresse au problème de conduction thermique en dimension un suivant :

$$\frac{\partial u}{\partial t} = \frac{\partial}{\partial x} \left[a(x) \frac{\partial u}{\partial x} \right] + f(x, t), \quad (x, t) \in]0, \ell[\times]0, T], \tag{2}$$

avec condition initiale

$$u(x, 0) = 0, \quad 0 \leq x \leq \ell, \tag{3}$$

et les conditions aux limites de Dirichlet-Neumann

$$u(0, t) = 0 \quad \text{et} \quad a(\ell) \frac{\partial u}{\partial x}(\ell, t) = h(t), \quad \text{pour tout } 0 \leq x \leq \ell. \tag{4}$$

où $f(x, t)$, $h(t)$ et $a(x)$ sont des fonctions continues. Nous considérons le problème (2)-(4) comme un problème direct. Donc, si $f(x, t)$, $u_0(x)$, $g(t)$ et $h(t)$ sont des fonctions continues et $a(x)$ connaît, le problème (2)-(4) a une solution unique.

Pour le problème inverse, le coefficient de conduction thermique $a(x)$ est considéré comme étant inconnu. De plus, on considère aussi une condition supplémentaire sur la frontière $x = \ell$ donnée par :

$$u(\ell, t) = g(t), \quad 0 \leq t \leq T. \quad (5)$$

Ce mémoire se décompose en trois chapitres de la manière suivante :

Dans le premier chapitre, nous avons donné un rappel sur le problème des moindres carrés non linéaires, ainsi que les algorithmes des méthodes suivantes :

- ☞ Méthode de Gauss-Newton,
- ☞ Méthode de Levenberg-Marquardt.

Dans le second chapitre, nous avons utilisé la méthode des éléments finis pour résoudre le problème de conduction thermique avec conditions aux limites de Dirichlet-Neumann. La méthode des éléments finis est basé sur trois idées principales. D'abord, nous obtenons la formulation variationnelle du problème direct. Ensuite, nous discrétisons la formulation variationnelle par éléments finis \mathbb{P}_1 et nous dérivons un système d'équations différentielles linéaires de premier ordre. Avec la méthode d'Euler implicite et la méthode de Cholesky, nous avons proposé une algorithmes pour résoudre ce système.

Dans le dernier chapitre, nous avons étudié le problème inverse (2)-(5) pour estimer la fonction inconnue $a(x)$. Dans ce cas, on propose la forme polynomiale pour la fonction inconnue $a(x)$. Donc, $a(x)$ est rapproché comme

$$a(x) = p_0 + p_1x + p_2x^2 + \dots + p_mx^m,$$

où p_0, p_1, \dots, p_m sont des constantes. Les coefficients inconnus p_0, p_1, \dots, p_m peuvent être déterminés en utilisant la méthode des moindres carrés non linéaires suivante :

$$F(p_0, p_1, \dots, p_m) = \frac{1}{2} \sum_{i=1}^n [u(1, t_i) - \varphi(t_i)]^2 \quad (6)$$

En se basant sur la méthode de Levenberg-Marquardt et la méthode des éléments finis, nous avons proposé un algorithmes pour résoudre le problème (6).

Problème des moindres carrés non linéaires

*D*ans ce chapitre, on s'intéresse au problème qui consiste à adapter un modèle à des observations qui sont entachées d'erreurs. Nous nous intéresserons ici au cas où le nombre d'observations est supérieur au nombre de paramètres. Ceci nous conduira à résoudre, suivant le choix du modèle, des systèmes linéaires ou non-linéaires surdéterminés. Ce problème est appelé le problème des moindres carrés. Nous avons donné un rappel sur la méthode des moindres carrés non linéaires, ainsi que les algorithmes de Gauss-Newton et de Levenberg-Marquardt.

Définition 1.1. Le problème des moindres carrés non linéaire consiste à minimiser la fonction

$$g(x) = \frac{1}{2} r^T(x) r(x) = \frac{1}{2} \sum_{i=1}^m r_i^2(x). \quad (1.1)$$

avec

$$r_i(x) = y_i - f(t_i, x), \quad i = 1, \dots, m$$

où $f(t_i, x)$ est une fonction non linéaire, t_i les variables indépendantes et $x \in \mathbb{R}^n$ le vecteur de paramètres à estimer.

Afin d'écrire le modèle quadratique pour la minimisation de (1.1) nous avons besoin des dérivées premières et secondes de g . La dérivée première s'écrit

$$\nabla g(x) = \sum_{i=1}^m r_i(x) \cdot \nabla r_i(x) = \nabla^T r(x) r(x) \quad (1.2)$$

avec

$$\nabla r(x) = \begin{bmatrix} \frac{\partial r_1(x)}{\partial x_1} & \cdots & \frac{\partial r_1(x)}{\partial x_n} \\ \vdots & & \vdots \\ \frac{\partial r_m(x)}{\partial x_1} & \cdots & \frac{\partial r_m(x)}{\partial x_n} \end{bmatrix}$$

la matrice dite Jacobienne. Le vecteur

$$\nabla r_i(x) = \begin{bmatrix} \frac{\partial r_i(x)}{\partial x_1} \\ \vdots \\ \frac{\partial r_i(x)}{\partial x_n} \end{bmatrix}$$

correspond à la ligne i de la matrice Jacobienne.

La dérivée seconde s'écrit

$$\begin{aligned} \nabla^2 g(x) &= \sum_{i=1}^m \left[\nabla r_i(x) \cdot \nabla^T r_i(x) + r_i(x) \nabla^2 r_i(x) \right] \\ &= \nabla^T r(x) \nabla r^T(x) + S(x) \end{aligned} \quad (1.3)$$

avec

$$S(x) = \sum_{i=1}^m r_i(x) \nabla^2 r_i(x).$$

Exemple 1.1. Soit la fonction non linéaire

$$f(t, x) = x_1 e^{x_2 t}.$$

et les données suivantes :

t_i	0.0	1.0	2.0	3.0
y_i	2.0	0.7	0.3	0.1

On a le vecteur des résidus

$$r(x) = \begin{bmatrix} y_1 - x_1 e^{x_2 t_1} \\ y_2 - x_1 e^{x_2 t_2} \\ y_3 - x_1 e^{x_2 t_3} \\ y_4 - x_1 e^{x_2 t_4} \end{bmatrix}$$

et la matrice Jacobienne

$$\nabla r(x) = \begin{bmatrix} \frac{\partial r_1(x)}{\partial x_1} & \frac{\partial r_1(x)}{\partial x_2} \\ \frac{\partial r_2(x)}{\partial x_1} & \frac{\partial r_2(x)}{\partial x_2} \\ \frac{\partial r_3(x)}{\partial x_1} & \frac{\partial r_3(x)}{\partial x_2} \\ \frac{\partial r_4(x)}{\partial x_1} & \frac{\partial r_4(x)}{\partial x_2} \end{bmatrix} = \begin{bmatrix} -e^{x_2 t_1} - x_1 t_1 e^{x_2 t_1} \\ -e^{x_2 t_2} - x_1 t_2 e^{x_2 t_2} \\ -e^{x_2 t_3} - x_1 t_3 e^{x_2 t_3} \\ -e^{x_2 t_4} - x_1 t_4 e^{x_2 t_4} \end{bmatrix}.$$

Les dérivées premières sont :

$$\nabla g(x) = \begin{bmatrix} \frac{\partial g(x)}{\partial x_1} \\ \frac{\partial g(x)}{\partial x_2} \end{bmatrix} = \nabla^T r(x) r(x) = \begin{bmatrix} -\sum_{i=1}^4 r_i(x) e^{x_2 t_i} \\ -\sum_{i=1}^4 r_i(x) x_1 t_i e^{x_2 t_i} \end{bmatrix}.$$

On a les m matrices de dérivées secondes

$$\nabla^2 r_i(x) = \begin{bmatrix} \frac{\partial^2 r_i(x)}{\partial x_1^2} & \frac{\partial^2 r_i(x)}{\partial x_1 \partial x_2} \\ \frac{\partial^2 r_i(x)}{\partial x_2 \partial x_1} & \frac{\partial^2 r_i(x)}{\partial x_2^2} \end{bmatrix} = \begin{bmatrix} 0 & -t_i e^{x_2 t_i} \\ -t_i e^{x_2 t_i} & -x_1 t_i^2 e^{x_2 t_i} \end{bmatrix}$$

et la matrice des dérivées secondes de g est

$$\nabla^2 g(x) = \begin{bmatrix} \sum_{i=1}^4 e^{2x_2 t_i} & \sum_{i=1}^4 x_1 t_i e^{2x_2 t_i} \\ \sum_{i=1}^4 x_1 t_i e^{2x_2 t_i} & \sum_{i=1}^4 x_1^2 t_i^2 e^{2x_2 t_i} \end{bmatrix} - \sum_{i=1}^4 r_i(x) \begin{bmatrix} 0 & t_i e^{x_2 t_i} \\ t_i e^{x_2 t_i} & x_1 t_i^2 e^{x_2 t_i} \end{bmatrix}.$$

Considérons $m_c(x)$, l'approximation quadratique de g dans un voisinage de x_c ,

$$m_c(x) = g(x_c) + \nabla^T g(x_c) (x - x_c) + \frac{1}{2} (x - x_c)^T \nabla^2 g(x_c) (x - x_c)$$

et cherchons le point $x_+ = x_c + s_N$ pour lequel $\nabla m_c(x_+) = 0$ ce qui est la condition du premier ordre nécessaire pour que x_+ minimise m_c . On a :

$$\nabla m_c(x_+) = \nabla g(x_c) + \nabla^2 g(x_c) \underbrace{(x_+ - x_c)}_{s_N} = 0$$

où s_N est le pas de Newton que l'on détermine en résolvant le système linéaire

$$\nabla^2 g(x_c) s_N = -\nabla g(x_c).$$

Dès lors on pourrait envisager la solution du problème (1.1) avec la méthode de Newton dont l'itération k est définie comme

$$\begin{cases} \nabla^2 g(x^{(k)}) s_N^{(k)} = -\nabla g(x^{(k)}), \\ x^{(k+1)} = x^{(k)} + s_N^{(k)}. \end{cases}$$

Remarque 1.1. Dans la pratique on procède cependant différemment étant donné que l'évaluation de $S(x)$ dans (1.3) peut s'avérer très difficile.

1.1 Méthode de Gauss-Newton

Cette méthode utilise une approximation de la matrice des dérivées secondes (1.3) en omettant le terme $S(x)$. Comme $S(x)$ est composé d'une somme de termes $r_i(x) \nabla^2 r_i(x)$. Cette simplification se justifie dans une situation où les résidus $r_i(x)$ sont petits.

La méthode de Gauss-Newton se résume dans l'algorithme qui suit.

Algorithme 1 Méthode de Gauss-Newton pour les moindres carrés non linéaires

- 1: Choisir $x^{(0)}$
- 2: **Pour** $k = 0, 1, \dots$, jusqu'à la convergence **faire**
- 3: Calculer $\nabla r(x^{(k)})$
- 4: Résoudre

$$\left[\nabla r(x^{(k)})^T \nabla r(x^{(k)}) \right] s_{GN}^{(k)} = -\nabla r(x^{(k)})^T r(x^{(k)}) \quad (1.4)$$

- 5: Adapter $x^{(k+1)} = x^{(k)} + s_{GN}^{(k)}$
 - 6: **fin du Pour**
-

Remarque 1.2. ☞ On remarque que le système linéaire (1.4) qui définit le pas de Gauss-Newton $s_{GN}^{(k)}$ est un système d'équations normales et que le pas est aussi la solution d'un problème des moindres carrés linéaires.

☞ En accord à ce qui a été dit précédemment au sujet des moindres carrés linéaires on préfère la solution du système surdéterminé

$$\nabla r(x^{(k)}) s_{GN}^{(k)} \approx -r(x^{(k)})$$

obtenu à partir de la factorisation QR à la solution des équations normales.

☞ L'algorithme peut ne pas converger si le point de départ est choisi trop loin de la solution.

☞ Lorsque les résidus sont grands au point de la solution, l'approximation de la matrice des dérivées secondes peut s'avérer insuffisante, avec comme conséquence soit une convergence très lente ou pas de convergence du tout.

1.2 Méthode de Levenberg-Marquardt

Lorsque la méthode de Gauss-Newton échoue, notamment lorsque la solution du sous-problème de la solution du système linéaire n'est pas de rang complet, la méthode de Levenberg-Marquardt constitue une alternative intéressante. Dans cette méthode on approche la matrice $S(x)$ par une matrice diagonale μI . Nous avons alors l'algorithme suivant :

Algorithme 2 Méthode de Levenberg-Marquardt pour les moindres carrés non linéaires

-
- 1: Choisir $x^{(0)}$
 - 2: **Pour** $k = 0, 1, \dots$, jusqu'à la convergence **faire**
 - 3: Calculer $\nabla r(x^{(k)})$ et μ_k
 - 4: Résoudre

$$\left[\nabla r(x^{(k)})^T \nabla r(x^{(k)}) + \mu_k I \right] s_{LM}^{(k)} = -\nabla r(x^{(k)})^T r(x^{(k)}) \quad (1.5)$$

- 5: Adapter $x^{(k+1)} = x^{(k)} + s_{LM}^{(k)}$
 - 6: **fin du Pour**
-

Remarque 1.3. ☞ La solution du pas $s_{LM}^{(k)}$ dans (1.5) est définie à partir du problème des moindres carrés linéaires suivant :

$$\begin{bmatrix} \nabla r(x^{(k)}) \\ \mu_k^{1/2} I \end{bmatrix} s_{LM}^{(k)} \approx - \begin{bmatrix} r(x^{(k)}) \\ 0 \end{bmatrix}$$

que l'on résoudra en utilisant la factorisation QR. Ainsi on évite le calcul du produit $\nabla r(x^{(k)})^T \nabla r(x^{(k)})$ qui peut introduire des instabilités numériques.

- ☞ μ_k est choisi à partir des considérations sur la "région de confiance". Dans la pratique on choisit la valeur 10^{-2} .
- ☞ Si μ_k est choisi judicieusement la méthode de Levenberg-Marquardt s'avère très robuste en pratique. Elle constitue l'algorithme de base dans un grand nombre de logiciels spécialisés.
- ☞ L'algorithme peut ne pas converger si le point de départ est choisi trop loin de la solution.
- ☞ Lorsque les résidus sont grands au point de la solution, l'approximation de la matrice des dérivées secondes peut s'avérer insuffisante, avec comme conséquence soit une convergence très lente ou pas de convergence du tout.

Problème direct : Problème de conduction thermique

Dans ce chapitre, nous avons utilisé la méthode des éléments finis pour résoudre le problème de conduction thermique avec conditions aux limites de Dirichlet-Neumann. La méthode des éléments finis est basée sur trois idées principales. D'abord, nous obtenons la formulation variationnelle du problème direct. Ensuite, nous discrétisons la formulation variationnelle par éléments finis \mathbb{P}_1 et nous dérivons un système d'équations différentielles linéaires de premier ordre. Avec la méthode d'Euler implicite, nous avons proposé un algorithme pour résoudre ce système. Finalement, nous avons programmé cet algorithme par Matlab et nous testons cet algorithme par quelques exemples numériques.

2.1 Position du problème

Considérons la conduction de la chaleur dans un fil métallique non homogène $[0, \ell]$ avec $\ell > 0$. La température initiale de fil est nulle et la limite $x = 0$ est isolée et l'autre limite $x = \ell$ est libre. Pour tout $t \in]0, T]$ avec $T > 0$, une source de la chaleur plane de force $f(x, t)$, appliquée sur $]0, \ell[$.

La formulation mathématique de ce problème de conduction thermique est donnée sous la forme suivante :

$$\frac{\partial u}{\partial t} = \frac{\partial}{\partial x} \left[a(x) \frac{\partial u}{\partial x} \right] + f(x, t), \quad (x, t) \in]0, \ell[\times]0, T] \quad (2.1)$$

avec la condition initiale

$$u(x, 0) = 0 \quad \text{pour tout } 0 \leq x \leq \ell \quad (2.2)$$

et les conditions aux limites de Dirichlet-Neumann

$$u(0, t) = 0 \quad \text{et} \quad a(\ell) \frac{\partial u}{\partial x}(\ell, t) = h(t), \quad \text{pour tout } 0 \leq t \leq T. \quad (2.3)$$

Où $f \in \mathcal{C}(0, T, L^2(0, \ell))$ et $h \in \mathcal{C}(0, T)$ et $a :]0, \ell[\rightarrow \mathbb{R}_+^*$ est une fonction dérivable et bornée c'est à dire :

$$\exists a_1, a_2 > 0, \quad a_1 \leq a(x) \leq a_2, \quad \text{pour tout } x \in [0, \ell]. \quad (2.4)$$

2.1.1 Formulation variationnelle

Soient V et H deux espaces de Hilbert réel définis par :

$$V := \left\{ v \in H^1(0, \ell) : v(0) = 0 \right\} \quad \text{et} \quad H = L^2(0, \ell).$$

et soit \mathcal{A} un sous-ensemble convexe et fermé de $\mathcal{C}^1(0, \ell)$ défini par :

$$\mathcal{A} := \left\{ a \in \mathcal{C}^1(0, \ell) / a_1 \leq a(x) \leq a_2, \quad a_1, a_2 > 0 \right\}.$$

En multipliant l'équation (2.1) par une fonction arbitraire $v \in V$ et en intégrant sur $[0, \ell]$, on obtient :

$$\int_0^\ell \frac{\partial u(x, t)}{\partial t} v(x) dx = \int_0^\ell \frac{\partial}{\partial x} \left(a(x) \frac{\partial u(x, t)}{\partial x} \right) v(x) dx + \int_0^\ell f(x, t) v(x) dx. \quad (2.5)$$

Par intégration par parties, le premier terme de membre gauche de (2.5) devient :

$$\begin{aligned} \int_0^\ell (a(x) u'(x))' v(x) dx &= a(x) u'(x) v(x) \Big|_0^\ell - \int_0^\ell a(x) u'(x) v'(x) dx \\ &= h(t) v(\ell) - \int_0^\ell a(x) u'(x) v'(x) dx. \end{aligned}$$

Par conséquent, la formulation variationnelle du problème (2.1)-(2.3) est donnée par :

$$\begin{cases} \text{Trouver } u : [0, T] \rightarrow V \text{ tel que} \\ \frac{\partial}{\partial t} (u, v)_H + b(a, u, v) = F(t, v) \text{ pour tout } v \in V \\ u(x, 0) = 0. \end{cases} \quad (2.6)$$

Où $b : \mathcal{A} \times \mathcal{C}(0, T; V) \times V \rightarrow \mathbb{R}$ une forme trilinéaire donnée par :

$$b(a, u, v) = \int_0^\ell a(x) \frac{\partial u(x, t)}{\partial x} \frac{\partial v(x)}{\partial x} dx \text{ pour tout } u, v \in V \quad (2.7)$$

et $F : [0, T] \times V \rightarrow \mathbb{R}$ une forme linéaire donnée par :

$$F(t, v) = \int_0^\ell f(x, t) v(x) dx + h(t) v(\ell) \text{ pour tout } v \in V. \quad (2.8)$$

Pour l'existence et l'unicité de la solution du problème (2.7), nous avons la proposition suivante :

Proposition 2.1. *Pour tout $a \in \mathcal{A}$ et $f \in \mathcal{C}(0, T, L^2(0, \ell))$, le problème variationnel (2.6) admet une solution unique $u \in \mathcal{C}(0, T; V)$.*

Démonstration. En utilisant l'inégalité de Cauchy-Schwarz, nous obtenons :

$$\begin{aligned} \forall v \in V, |F(t, v)| &\leq \int_0^\ell |f(x, t)| |v(x)| dx + |h(t)| |v(\ell)| \\ &\leq \left(\int_0^\ell f^2(x, t) dx \right)^{1/2} \left(\int_0^\ell v^2(x) dx \right)^{1/2} + |h(t)| \left(\int_0^\ell |v(x)|^2 dx \right)^{1/2} \\ &\leq M \|v\|_V \text{ où } M = \|f\|_{\mathcal{C}(0, T; L^2(0, \ell))} + \|h\|_{\mathcal{C}(0, T)}. \end{aligned} \quad (2.9)$$

Alors, la forme linéaire F est continue sur V .

De même b , on a :

$$\begin{aligned} \forall u, v \in V, |b(a, u, v)| &\leq \sup_{x \in [0, \ell]} |a(x)| \int_0^\ell |u'(x, t)| |v'(x)| dx \\ &\leq \sup_{x \in [0, \ell]} |a(x)| \left(\int_0^\ell (u')^2(x, t) dx \right)^{1/2} \left(\int_0^\ell (v')^2(x) dx \right)^{1/2} \\ &\leq \|a\|_{\mathcal{C}(0, \ell)} \|u\|_{\mathcal{C}(0, T; V)} \|v\|_V. \end{aligned} \quad (2.10)$$

Donc, la forme trilinéaire b est continue sur $\mathcal{C}(0, T; V) \times V$.

Pour la coercivité de b , on a :

$$\forall v \in V, b(a, v, v) = \int_0^\ell a(x) (v'(x))^2 dx \geq \inf_{x \in [0, \ell]} a(x) \int_0^\ell (v'(x))^2 dx \geq \alpha \|v\|_V^2. \quad (2.11)$$

Où $\alpha = \inf_{x \in [0, \ell]} a(x) / \sqrt{2}$.

Enfin, le problème (2.6) admet une solution unique $u \in \mathcal{C}(0, T; V)$. \square

2.2 Semi-discrétisation par éléments finis

Nous construisons un sous espace V_h de V de dimension finie et constitué des fonctions linéaires par morceaux. On se donne un entier naturel N et on pose $h = 1/(N + 1)$, $x_i = ih$ pour $i = 0, \dots, N + 1$ (avec $x_0 = 0$ et $x_{N+1} = 1$) et $I_i := [x_i, x_{i+1}]$ pour $i = 0, \dots, N$.

On définit l'espace vectoriel V_h par :

$$V_h := \left\{ v_h \in \mathcal{C}([0, 1]) : v_h|_{I_i} \in \mathbb{P}_1, v_h(0) = 0 \right\}. \quad (2.12)$$

où \mathbb{P}_1 désigne l'espace des polynômes de degré inférieur ou égale à 1.

On définit, pour $j = 1, \dots, N + 1$, les fonctions ϕ_j par :

$$\phi_j(x) := \begin{cases} \frac{x-x_{j-1}}{h} & \text{si } x \in [x_{j-1}, x_j], \\ \frac{x_{j+1}-x}{h} & \text{si } x \in [x_j, x_{j+1}], \\ 0 & \text{si } x \leq x_{j-1} \text{ ou si } x \geq x_{j+1}. \end{cases} \quad \text{pour tout } j = 1, \dots, N \quad (2.13)$$

$$\phi_{N+1}(x) := \begin{cases} \frac{x-x_N}{h} & \text{si } x \in [x_N, x_{N+1}], \\ 0 & \text{si } x \leq x_N. \end{cases} \quad (2.14)$$

Les fonctions ϕ_j , $j = 1, \dots, N + 1$, sont des fonctions "chapeau", elles vérifient

$$\phi_j \in V_h \text{ et } \phi_j(x_i) = \delta_{ij} = \begin{cases} 1 & \text{si } i = j, \\ 0 & \text{si } i \neq j. \end{cases}$$

L'approximation par éléments finis de la formulation variationnelle (2.6) est donnée par :

$$\begin{cases} \text{Trouver } u_h \in \mathcal{C}(0, T; V_h) \text{ telle que} \\ \frac{\partial}{\partial t}(u_h(t), v_h)_H + b(a, u_h(t), v_h) = F(t, v_h) \text{ pour tout } v_h \in V_h. \end{cases} \quad (2.15)$$

La solution u_h du problème variationnel (2.15) s'écrit

$$u_h = \sum_{j=1}^{N+1} u_j(t) \phi_j \quad \text{où } u_j = u_h(x_j, t).$$

Et par la substitution $v_h = \phi_i$ pour $1 \leq i \leq N + 1$, le problème (2.15) est équivalent le système linéaire d'équations différentielles de premier ordre suivant :

$$\begin{cases} \text{Déterminer } U = (u_1(t), \dots, u_{N+1}(t))^T \text{ solution de} \\ AU'(t) + KU(t) = R(t) \end{cases} \quad (2.16)$$

Où $K = (k_{ij})$ est une matrice symétrique appelée matrice de rigidité donnée par :

$$k_{ij} = b(a, \phi_j, \phi_i) \quad \forall 1 \leq i, j \leq N + 1,$$

$A = (A_{ij})$ est une matrice symétrique appelée matrice de masse donnée par :

$$A_{ij} = (\phi_j, \phi_i)_H \quad \forall 1 \leq i, j \leq N + 1,$$

et $R(t) = (r_i(t))$ le vecteur de terme source donné par :

$$r_i = F(t, \phi_i) \quad \forall 1 \leq i \leq N + 1.$$

2.3 Implémentation numérique

Pour calculer les composantes du vecteur de terme source et les éléments des matrices de rigidité et de masse, nous utilisons la règle de Simpson suivante :

$$\int_a^b g(x) dx \approx \frac{b-a}{6} \left[g(a) + 4g\left(\frac{a+b}{2}\right) + g(b) \right].$$

2.3.1 Vecteur de terme source

Pour $i = 1, \dots, N$ nous avons :

$$\begin{aligned} r_i &= F(t, \phi_i) = \frac{1}{h} \int_{x_{i-1}}^{x_i} (x - x_{i-1}) f(x, t) dx - \frac{1}{h} \int_{x_i}^{x_{i+1}} (x - x_{i+1}) f(x, t) dx \\ &= \frac{1}{h} \cdot \frac{h}{6} \left[(x_{i-1} - x_{i-1}) f(x_{i-1}, t) + 4 \left(\frac{x_{i-1} + x_i}{2} - x_{i-1} \right) f\left(\frac{x_{i-1} + x_i}{2}, t\right) + (x_i - x_{i-1}) f(x_i, t) \right] \\ &\quad - \frac{1}{h} \cdot \frac{h}{6} \left[(x_i - x_{i+1}) f(x_i, t) + 4 \left(\frac{x_i + x_{i+1}}{2} - x_{i+1} \right) f\left(\frac{x_i + x_{i+1}}{2}, t\right) + (x_{i+1} - x_{i+1}) f(x_{i+1}, t) \right] \\ &= \frac{h}{3} \left[f\left(\frac{x_{i-1} + x_i}{2}, t\right) + f(x_i, t) + f\left(\frac{x_i + x_{i+1}}{2}, t\right) \right]. \end{aligned}$$

et

$$r_{N+1} = F(t, \phi_{N+1}) = \frac{h}{6} \left[2f\left(\frac{x_N + x_{N+1}}{2}, t\right) + f(x_{N+1}, t) \right] + h(t).$$

2.3.2 Matrice de rigidité

Si $|i - j| > 1$, $\text{supp}(\phi_i) \cap \text{supp}(\phi_j) = \emptyset$, donc $k_{ij} = 0$ alors la matrice K est creuse. Pour les éléments de diagonale principale, nous avons pour $i = 1, \dots, N$:

$$\begin{aligned} d_i &= K_{i,i} = \int_{x_{i-1}}^{x_{i+1}} a(x) (\phi'_i)^2 dx = \frac{1}{h^2} \int_{x_{i-1}}^{x_i} a(x) dx + \frac{1}{h^2} \int_{x_i}^{x_{i+1}} a(x) dx \\ &= \frac{1}{6h} \left[a(x_{i-1}) + 4a\left(\frac{x_{i-1} + x_i}{2}\right) + 2a(x_i) + 4a\left(\frac{x_i + x_{i+1}}{2}\right) + a(x_{i+1}) \right]. \end{aligned}$$

et

$$\begin{aligned} d_{N+1} &= K_{N+1,N+1} = \int_{x_N}^{x_{N+1}} a(x) (\phi'_{N+1}(x))^2 dx = \frac{1}{h^2} \int_{x_N}^{x_{N+1}} a(x) dx \\ &= \frac{1}{6h} \left[a(x_N) + 4a\left(\frac{x_N + x_{N+1}}{2}\right) + a(x_{N+1}) \right]. \end{aligned}$$

Pour tout $i = 1, \dots, N$, nous avons :

$$\begin{aligned} c_i &= K_{i+1,i} = K_{i,i+1} = \int_{x_i}^{x_{i+1}} a(x) \phi'_{i+1}(x) \phi'_i(x) dx = -\frac{1}{h^2} \int_{x_i}^{x_{i+1}} a(x) dx \\ &= -\frac{1}{6h} \left[a(x_i) + 4a\left(\frac{x_i + x_{i+1}}{2}\right) + a(x_{i+1}) \right]. \end{aligned}$$

Enfin, la matrice K est tridiagonale.

2.3.3 Matrice de masse

Si $|i - j| > 1$, $\text{supp}(\phi_i) \cap \text{supp}(\phi_j) = \emptyset$, donc $A_{ij} = 0$ alors la matrice A est creuse.

Pour les éléments de diagonale principale, nous avons pour $i = 1, \dots, N$:

$$e_i = A_{i,i} = \int_{x_{i-1}}^{x_{i+1}} \phi_i^2(x) dx = 2h/3.$$

et

$$e_{N+1} = A_{N+1,N+1} = \int_{x_N}^{x_{N+1}} \phi_{N+1}^2(x) dx = h/3.$$

Pour tout $i = 1, \dots, N$, nous avons :

$$s_i = A_{i+1,i} = A_{i,i+1} = \int_{x_i}^{x_{i+1}} \phi_i(x) \phi_{i+1}(x) dx = h/6.$$

2.3.4 Méthode d'Euler implicite

En utilisant la méthode d'Euler implicite sur le système différentiel (2.16), on obtient le système linéaire suivant :

$$\begin{cases} U^0 = 0, \\ \left(A + \frac{\Delta t}{2} K\right) U^{k+1} = \left(A - \frac{\Delta t}{2} K\right) U^k + \frac{\Delta t}{2} (R^k + R^{k+1}). \end{cases} \quad (2.17)$$

Où $U^k = U(t_k)$ et $R^k = R(t_k)$ pour tout $k = 1, \dots, I$ tels que $\Delta t = T/I$.

2.3.5 Méthode de Cholesky

Le système linéaire (2.17) est un système linéaire à matrice symétrique, définie positive et tridiagonale. Donc, on peut utiliser l'algorithme de Cholesky pour la résolution numérique comme suit :

Initiation : Soient a , f et h trois fonctions données, $N \in \mathbb{N}^*$ et $h = 1/(N+1)$ et $x_i = ih$.

Pour $i = 1, \dots, N$, on a :

$$\begin{aligned} r_i &= \frac{h}{3} \left[f\left(\frac{x_{i-1} + x_i}{2}\right) + f(x_i) + f\left(\frac{x_i + x_{i+1}}{2}\right) \right] \\ r_{N+1} &= \frac{h}{6} \left[2f\left(\frac{x_N + x_{N+1}}{2}\right) + f(x_{N+1}) \right] + h(t). \end{aligned}$$

et pour $i = 1, \dots, N$, on a :

$$\begin{aligned} b_i &= \frac{1}{6h} \left[a(x_{i-1}) + 4a\left(\frac{x_{i-1} + x_i}{2}\right) + 2a(x_i) + 4a\left(\frac{x_i + x_{i+1}}{2}\right) + a(x_{i+1}) \right] \\ b_{N+1} &= \frac{1}{6h} \left[a(x_N) + 4a\left(\frac{x_N + x_{N+1}}{2}\right) + a(x_{N+1}) \right] \\ c_i &= -\frac{1}{6h} \left[a(x_i) + 4a\left(\frac{x_i + x_{i+1}}{2}\right) + a(x_{i+1}) \right]. \end{aligned}$$

Factorisation :

$$d_1 = \sqrt{b_1}$$

Pour tout $i = 2, \dots, N$

$$\begin{cases} \ell_i = c_{i-1}/d_{i-1}, \\ d_i = \sqrt{b_i - \ell_i^2}. \end{cases}$$

Résolution :

$$Ly = r \iff \begin{cases} y_1 = r_1/d_1 \text{ et} \\ y_i = (r_i - \ell_i y_{i-1})/d_i, \text{ pour tout } i = 2, \dots, N + 1. \end{cases}$$

$$L^T u = y \iff \begin{cases} u_{N+1} = y_{N+1}/d_{N+1} \text{ et} \\ u_i = (y_i - \ell_{i+1} u_{i+1})/d_i, \text{ pour tout } i = N, \dots, 1. \end{cases}$$

Méthode de Levenberg-Marquardt pour un problème inverse de conduction thermique

Dans ce chapitre, nous avons proposé un algorithme basé sur la méthode de Levenberg-Marquardt et la méthode des éléments finis pour estimer un coefficient dans un problème non linéaire gouverné par une équation de la chaleur dans un milieu non homogène en dimension un avec condition initiale et des conditions aux limites de Dirichlet-Neumann. Le coefficient est rapproché par une forme polynomiale et l'algorithme numérique est employé pour trouver les coefficients de cet polynôme.

La méthode de Levenberg-Marquardt, initialement conçue pour l'application aux problèmes d'estimation des paramètres non linéaires. La solution des problèmes inverses de conduction thermique avec la méthode de Levenberg-Marquardt peut être organisée dans les étapes de base suivantes :

- ☞ Le problème inverse,
- ☞ Le schéma itératif,
- ☞ Le critère d'arrêt,
- ☞ L'algorithme de calcul.

Nous présentons ci-dessous les détails de chacune de ces étapes appliquées sur le problème direct (2.1)-(2.3)

3.1 Problème inverse

Pour le problème inverse, le coefficient de diffusion $a(x)$ est considéré comme inconnu. Une condition supplémentaire obtenue à partir de mesures de température sur la frontière $x = \ell$ donnée par :

$$u(\ell, t_i) = g(t_i), \quad 0 \leq t_i \leq T \text{ avec } i = 1, \dots, I. \quad (3.1)$$

est utilisé pour estimer le coefficient de diffusion $a(x)$. Pour la solution de ce problème inverse, nous considérons que la fonction a doit être paramétrée sous la forme d'un polynôme suivant :

$$a(x) \approx p_0 + p_1x + p_2x^2 + \dots + p_mx^m. \quad (3.2)$$

Les coefficients inconnus p_0, p_1, \dots, p_m peuvent être déterminés en utilisant le problème des moindres carrés non linéaires suivante :

$$S(P) = \sum_{i=1}^I [u_i(P) - g_i]^2. \quad (3.3)$$

Où

- S = Somme d'erreur carré ou fonction objective
- $P^T = (p_0, p_1, \dots, p_m)$ vecteur de paramètres inconnus
- $u_i(P) \equiv u(\ell, t_i, P)$ = température estimée en t_i
- $g_i \equiv g(t_i)$ = température mesurée en t_i
- m = nombre total de paramètres inconnus
- I = Nombre total de mesures, où $I \geq m$.

Les températures estimées $u_i(P)$ sont obtenues à partir de la solution du problème direct sur la frontière $x = \ell$. L'équation (3.3) peut être écrite sous forme vectorielle suivante :

$$S(P) = [U(P) - G]^T [U(P) - G]. \quad (3.4)$$

Où l'exposant T désigne la transposition, et le vecteur $[U(P) - G]^T$ est défini par :

$$[U(P) - G]^T = [u_1(P) - g_1, u_2 - g_2, \dots, u_I - g_I]. \quad (3.5)$$

Remarque 3.1. La fonction S définie par (3.4) est continue sur un domaine borné et fermé de \mathbb{R}^{m+1} . Donc, d'après théorème de Weierstrass, la fonction S est atteinte son minimum global.

3.2 Schéma itératif

Pour minimiser la fonction des moindres carrés non linéaires donnée par (3.4) et d'après Remarque 3.1, en utilisant la condition nécessaire d'optimalité de premier ordre, nous avons :

$$\frac{\partial S(P)}{\partial p_0} = \frac{\partial S(P)}{\partial p_1} = \dots = \frac{\partial S(P)}{\partial p_m} = 0. \quad (3.6)$$

Cette condition nécessaire pour la minimisation de $S(P)$ peut être représentée sous forme matricielle suivante :

$$\nabla S(P) = 2 \frac{\partial U^T(P)}{\partial P} (U(P) - G) = 0. \quad (3.7)$$

Où

$$\frac{\partial U^T(P)}{\partial P} = \begin{bmatrix} \frac{\partial}{\partial p_0} \\ \frac{\partial}{\partial p_1} \\ \vdots \\ \frac{\partial}{\partial p_m} \end{bmatrix} \begin{bmatrix} u_1 & u_2 & \dots & u_I \end{bmatrix} = \begin{bmatrix} \frac{\partial u_1}{\partial p_0} & \frac{\partial u_2}{\partial p_0} & \dots & \frac{\partial u_I}{\partial p_0} \\ \frac{\partial u_1}{\partial p_1} & \frac{\partial u_2}{\partial p_1} & \dots & \frac{\partial u_I}{\partial p_1} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial u_1}{\partial p_m} & \frac{\partial u_2}{\partial p_m} & \dots & \frac{\partial u_I}{\partial p_m} \end{bmatrix} \quad (3.8)$$

La matrice jacobienne $J(P)$ est définie comme la transposition de l'équation (3.8), c'est-à-dire

$$J(P) = \left[\frac{\partial U^T(P)}{\partial P} \right]^T \quad (3.9)$$

Sous forme explicite, la matrice jacobienne est écrite comme

$$J(P) = \left[\frac{\partial U^T(P)}{\partial P} \right]^T = \begin{bmatrix} \frac{\partial u_1}{\partial p_0} & \frac{\partial u_1}{\partial p_1} & \dots & \frac{\partial u_1}{\partial p_m} \\ \frac{\partial u_2}{\partial p_0} & \frac{\partial u_2}{\partial p_1} & \dots & \frac{\partial u_2}{\partial p_m} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial u_I}{\partial p_0} & \frac{\partial u_I}{\partial p_1} & \dots & \frac{\partial u_I}{\partial p_m} \end{bmatrix} \quad (3.10)$$

où m est le nombre total de paramètres inconnus et I est le nombre total de mesures.

Les éléments de la matrice jacobienne sont appelés coefficients de sensibilité. Le coefficient de sensibilité J_{ij} est donc défini comme la dérivée première de la température estimée à l'instant t_i par rapport au paramètre inconnu p_j , c'est-à-dire,

$$J_{ij} = \frac{\partial u_i}{\partial p_j}, \quad i = 1, \dots, I \quad \text{et} \quad j = 0, 1, \dots, m. \quad (3.11)$$

En utilisant la définition de la matrice jacobienne donnée par l'équation (3.9), l'équation (3.7) devienne :

$$2J^T(P) [U(P) - G] = 0 \quad (3.12)$$

Le problème inverse (3.4) est non linéaire, donc la matrice jacobienne est dépendre du vecteur des paramètres inconnus P . La solution de l'équation (3.12) nécessite alors une procédure itérative, qui est obtenue en linéarisant le vecteur des températures estimées, $U(P)$, avec un développement de Taylor autour de la solution à l'itération k . Une telle linéarisation est donnée par :

$$U(P) = U(P^k) + J^k (P - P^k). \quad (3.13)$$

Où $U(P^k)$ et J^k sont les températures et la matrice jacobienne à l'itération k , respectivement. En substituant l'équation (3.13) dans l'équation (3.12), nous obtenons le schéma itératif du vecteur des paramètres inconnus P , voir [3] :

$$P^{k+1} = P^k + \left[(J^k)^T J^k \right]^{-1} (J^k)^T [G - U(P^k)] \quad (3.14)$$

Le schéma itératif donné par l'équation (3.14) s'appelle la méthode de Gauss. Cette méthode est une approximation de la méthode Newton, voir [2]. Nous notons que le schéma itératif (3.14), exigent que la matrice $J^T J$ n'est pas singulier, c'est à dire :

$$\det(J^T J) \neq 0 \quad (3.15)$$

La condition (3.15) s'appelle condition d'identification, c'est-à-dire si le déterminant de $J^T J$ est nul ou très petit, les paramètres p_j , pour $j = 0, \dots, m$, ne peuvent être déterminés par le schéma itératif (3.14).

Les problèmes satisfaisant $\det(J^T J) \approx 0$ sont notés mal conditionnés. Les problèmes inverses de conduction thermique sont généralement très mal conditionnés. La méthode Levenberg-Marquardt [3, 2, 5, 10, 6, 7, 9, 8] allégé cette difficulté en utilisant un schéma itératif sous la forme :

$$P^{k+1} = P^k + \left[(J^k)^T J^k + \mu^k \Omega^k \right]^{-1} (J^k)^T [G - U(P^k)] \quad (3.16)$$

où $\mu^k > 0$ est un paramètre d'amortissement et Ω^k est une matrice diagonale.

3.3 Critère d'arrêt

Les critères suivants ont été proposés par Dennis et Schnabel [5] pour arrêter le schéma itératif de la méthode de Levenberg-Marquardt donnée par l'équation (3.14) :

$$S(P^{k+1}) < \varepsilon_1, \quad (3.17a)$$

$$\left\| (J^k)^T [G - U(P^k)] \right\| < \varepsilon_2, \quad (3.17b)$$

$$\left\| P^{k+1} - P^k \right\| < \varepsilon_3. \quad (3.17c)$$

Où ε_1 , ε_2 et ε_3 sont des tolérances donnés, et $\|\cdot\|$ est la norme euclidienne d'un vecteur.

3.4 Algorithme de calcul

Différentes versions de la méthode de Levenberg-Marquardt peuvent être trouvées dans la littérature, selon le choix de la matrice diagonale Ω^k et sur la forme choisie pour du paramètre d'amortissement μ^k , voir [3, 2, 5, 10, 6, 7, 9, 8]. Avec la matrice Ω^k donnée par :

$$\Omega^k = \text{diag} \left[\left((J^k)^T J^k \right) \right] \quad (3.18)$$

Nous présentons un algorithme de calcul basé sur la méthode de Levenberg-Marquardt pour résoudre le problème inverse de conduction thermique (2.1)-(2.3) et(3.1) , voir [10].

Algorithme 3 (Algorithme de calcul basé sur la méthode de Levenberg-Marquardt).

Entrée(s) $G = (g_1, g_2, \dots, g_I)^T$, P^0 donnés, $\mu^0 = 0.001$ et $k = 0$.

- 1: Résoudre le problème direct (2.1)-(2.3) avec l'estimation disponible P^k pour obtenir le vecteur de température $U(P^k) = (u_1, u_2, \dots, u_I)^T$.
- 2: Calculer $S(P^k)$ à partir de l'équation (3.4).
- 3: Calculez la matrice jacobienne J^k définie par l'équation (3.9), puis la matrice Ω^k donnée par l'équation (3.18), en utilisant les valeurs courantes de P^k .
- 4: Résoudre le système linéaire suivant :

$$\left[(J^k)^T J^k + \mu^k \Omega^k \right] \Delta P^k = (J^k)^T [G - U(P^k)]$$

afin de calculer $\Delta P^k = P^{k+1} - P^k$.

- 5: Calculer la nouvelle estimation de P^{k+1} ,

$$P^{k+1} = P^k + \Delta P^k$$

- 6: Résoudre le problème direct (2.1)-(2.3) avec la nouvelle estimation P^{k+1} pour trouver $U(P^{k+1})$. Ensuite, calculer $S(P^{k+1})$, tel que défini par l'équation (3.4).
 - 7: Si $S(P^{k+1}) \geq S(P^k)$, remplace μ^k par $10\mu^k$ et retourne à l'étape 4.
 - 8: Si $S(P^{k+1}) < S(P^k)$, accepte la nouvelle estimation P^{k+1} et remplace μ^k par $0.1\mu^k$.
 - 9: Vérifier les critères d'arrêt donnés par les équations (3.17a)-(3.17c). Arrêter la procédure itérative si l'un d'ils est satisfait ; sinon, remplacer k par $k + 1$ et retourne à l'étape 3.
-

3.5 Calcul des coefficients de la matrice jacobienne

On note $u_k = \frac{\partial u}{\partial p_k}$ pour tout $k = 0, 1, \dots, m$. En dérivant les équations (2.1)-(2.3) par rapport à p_k . Donc, nous avons :

$$\begin{cases} \frac{\partial u_k}{\partial t} = \frac{\partial}{\partial x} \left[(p_0 + p_1 x + \dots + p_m x^m) \frac{\partial u_k}{\partial x} + x^k \frac{\partial u}{\partial x} \right], \\ u_k(x, 0) = 0, \\ u_k(0, t) = 0, \\ a(\ell) \frac{\partial u_k}{\partial x}(\ell, t) = 0. \end{cases} \quad (3.19)$$

Avec le système (3.19) et les équations (2.1)-(2.3), nous construisons le système vectoriel suivant :

$$\begin{cases} \frac{\partial \vec{U}}{\partial t} + \frac{\partial \vec{\Gamma}}{\partial x} = \vec{F} \\ \vec{U}(x, 0) = 0 \\ \vec{U}(0, t) = 0 \\ \vec{a}(\ell) U(\ell, t) = \vec{H}(t) \end{cases} \quad (3.20)$$

ou

$$\vec{U} = \begin{pmatrix} u \\ u_0 \\ \cdot \\ \cdot \\ \cdot \\ u_m \end{pmatrix} \quad \text{et} \quad \vec{F} = \begin{pmatrix} f(x, t) \\ 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \end{pmatrix}$$

$$\vec{\Gamma} = \begin{pmatrix} -(p_0 + p_1 x + \dots + p_m x^m) \frac{\partial u}{\partial x} \\ -(p_0 + p_1 x + \dots + p_m x^m) \frac{\partial u_0}{\partial x} - x \left(\frac{\partial u}{\partial x} \right) \\ -(p_0 + p_1 x + \dots + p_m x^m) \frac{\partial u_1}{\partial x} - x \left(\frac{\partial u}{\partial x} \right) \\ \cdot \\ \cdot \\ -(p_0 + p_1 x + \dots + p_m x^m) \frac{\partial u_m}{\partial x} - x^m \left(\frac{\partial u}{\partial x} \right) \end{pmatrix}$$

$$\vec{H}(t) = [h(t), 0, \dots, 0]^T$$

3.5.1 Formulation variationnelle

Soit V un espace de Hilbert défini par :

$$V = \{v \in H^1(0, \ell) : v(0) = 0\}.$$

Nous multiplions (3.20) par une fonction de test $v \in V$ et intégrant l'équation obtenue sur $[0, \ell]$, nous obtenons l'équation suivante :

$$\int_0^\ell \frac{\partial U(x, t)}{\partial t} \cdot v(x) dx - \int_0^\ell \frac{\partial \Gamma}{\partial x} \cdot v(x) dx = \int_0^\ell F(x, t) \cdot v(x) dx.$$

Avec L'intégration par parties, nous avons :

$$\frac{d}{dt} \int_0^\ell U(x, t) \cdot v(x) dx + \int_0^\ell \Gamma \cdot \frac{\partial v(x, t)}{\partial x} dx = \int_0^\ell F(x, t) \cdot v(x) dx.$$

Nous utilisons le produit scalaire de l'espace $L^2(0, \ell)$:

$$(f, g) = \int_0^\ell f(x) \cdot g(x) dx$$

et on définit la forme bilinéaire suivante :

$$a(U, v) = \int_0^\ell \Gamma \cdot \frac{\partial v(x, t)}{\partial x} dx = \begin{cases} \int_0^\ell - (p_0 + p_1 x + \dots p_m x^m) \frac{\partial u}{\partial x} \frac{\partial v}{\partial x} dx \\ \int_0^\ell \left(- (p_0 + p_1 x + \dots p_m x^m) \frac{\partial u_1}{\partial x} \frac{\partial v}{\partial x} - \frac{\partial u}{\partial x} \frac{\partial v}{\partial x} \right) dx \\ \int_0^\ell \left(- (p_0 + p_1 x + \dots p_m x^m) \frac{\partial u_2}{\partial x} \frac{\partial v}{\partial x} - x^2 \frac{\partial u}{\partial x} \frac{\partial v}{\partial x} \right) dx \\ \cdot \\ \cdot \\ \int_0^\ell \left(- (p_0 + p_0 x + \dots p_m x^m) \frac{\partial u_m}{\partial x} \frac{\partial v}{\partial x} - x^m \frac{\partial u}{\partial x} \frac{\partial v}{\partial x} \right) dx. \end{cases}$$

Enfin , la formulation variationnelle du problème (3.20) est donnée par :

$$\begin{cases} \frac{d}{dt} (U, v)_{L^2(0, \ell)} + a(U, v) = (F, v)_{L^2(0, \ell)} \\ U(x, 0) = 0 \\ U(0, t) = 0 \\ a(\ell) U(\ell, t) = H(t) \end{cases} \quad (3.21)$$

3.5.2 Semi discrétisation par éléments finis

Soit V_h un sous espace de V de dimension $N + 2$. On considère le problème approché suivant : trouver $u_h, u_{0,h}, u_{1,h}, \dots, u_{m,h} \in V_h$ tels que

$$\begin{cases} \frac{d}{dt} (U_h, v_h) + a(U_h, v_h) = (F, v_h), \\ U_h(x, 0) = 0, \\ U_h(0, t) = 0, \\ a(\ell) U_h(\ell, t) = H(t). \end{cases} \quad (3.22)$$

Où $v_h \in V_h$ et $U_h = [u_h, u_{0,h}, u_{1,h}, \dots, u_{m,h}]^T$.

Nous subdivisons l'intervalle $[0, \ell]$ en $N + 1$ sous-intervalle des distances égales h :

$$0 = a_0 \leq a_1 \leq \dots \leq a_N \leq a_{N+1} = \ell$$

avec

$$a_i = (i - 1) \cdot h$$

Soit

$$V_h = \left\{ v_h \in C^0[0, \ell] : v_h|_{[a_i, a_{i+1}]} \in P_1, \forall i = \overline{1, N_x} \right\}$$

et

$$V_{0,h} = \{v_h \in V_h : v_h(0) = v_h(\ell) = 0\}.$$

Notez que la dimension finie nous permet de construire une base finie pour l'espace correspondant, Dans le cas de $V_{0,h}$ nous avons $\{\varphi_i\}_{i=2}^{N_x}$ ou $i = 2, \dots, N_x$.

$$\text{considérer : } \varphi_i(x) = \begin{cases} 0 & x \in [a_0, a_{i-1}] \\ \frac{x}{h} - (i-2) & x \in [a_{i-1}, a_i] \\ i - \frac{x}{h} & x \in [a_i, a_{i+1}] \\ 0 & x \in [a_{i+1}, a_{N_x+1}] \end{cases}$$

Alors que nous ajoutons pour V_h les deux fonctions φ et φ_{N_x+1} définies comme :

$$\varphi_1(x) = \begin{cases} 1 - \frac{x}{h} & \text{si } x \in [a_1, a_2] \\ 0 & \text{si } x \in [a_2, a_{N_x+1}] \end{cases}$$

$$\varphi_{N_x+1}(x) = \begin{cases} 0 & \text{si } x \in [a_1, a_2] \\ \frac{x}{h} - N_x + 1 & \text{si } x \in [a_{N_x}, a_{N_x+1}] \end{cases}$$

Afin que on peut écrit U_h comme une combinaison linéaire des éléments de base :

$$U_h(x, t) = \sum_{j=1}^{N_x+1} \tilde{U}_j(t) \cdot \varphi_j(x)$$

Sachant que $a(\cdot, \cdot)$ est une forme bilinéaire et que 3.22 est valable pour chaque élément de la base $\{\varphi_i\}_{i=2}^{N_x}$ on obtient

$$\sum_{j=1}^{N_x+1} \frac{d}{dt} \tilde{U}_j(t) \cdot (\varphi_j, \varphi_i) + \sum_{j=1}^{N_x+1} \tilde{U}_j(t) \cdot a(\varphi_j, \varphi_i) = (F, \varphi_i) \forall i = \overline{2, N_x} \quad (3.23)$$

Cette equation peut être écrite sous une forme vectorielle, pour cela, nous définissons les vecteurs \vec{u}, \vec{F} avec les composantes

$$F_i(t) = (F, \varphi_i)_{L^2}, u_j(t) = \tilde{u}_j(t), u_{0,j} = u_0(x_j)$$

Et les matrices M et A comme

$$m_{i,j} = (\varphi_i, \varphi_j)_{L^2}$$

$$a_{i,j} = a(\varphi_i, \varphi_j)$$

Notez que $M, A \in \mathbb{R}^{N_x-1 \times N_x+1}$, $\vec{u} \in \mathbb{R}^{N_x+1}$ et $\vec{F} \in \mathbb{R}^{N_x-1}$

Alors que (3.23) est égal au problème de Cauchy

$$M \frac{d}{dt} \vec{u}(t) + A \cdot \vec{u}(t) = \vec{F}(t) \quad \vec{u}(t_0) = \vec{u}_0, \quad (3.24)$$

3.5.3 Méthode Grank-Nicolson

La méthode Grank-Nicolson peut être appliquée à (3.24) à l'instant t_k , résultant en

$$M \left(\frac{\vec{u}_{k+1} - \vec{u}_k}{\Delta t} \right) + \frac{1}{2}A \cdot \vec{U}_{K+1} + \frac{1}{2}A \cdot \vec{U}_K = \frac{1}{2} \left(\vec{F}_k + \vec{F}_{k+1} \right) \quad (3.25)$$

Où $\vec{u}_k = \vec{u}(t_k)$, $\vec{F}_k = \vec{F}(t_k)$, $k = 0, 1, \dots$. L'équation (2.13) peut être écrite sous une forme simple comme suit :

$$\left(M + \frac{\Delta t}{2}A \right) \cdot u_{k+1} = \left(M - \frac{\Delta t}{2}A \right) \cdot \vec{u}_k + \frac{\Delta t}{2} \left(\vec{F}_k + \vec{F}_{k+1} \right) \quad (3.26)$$

Le système algébrique (3.11) est résolu par la méthode de cholesky.

Conclusion générale

Dans ce mémoire, nous avons étudié un problème inverse non linéaire pour l'identification numérique d'un paramètre dans un problème de conduction thermique en dimension un avec conditions aux limites de Dirichlet-Neumann. Ce travail se déroule en deux étapes :

- ✓ **Problème direct** : nous avons utilisé la méthode des éléments finis et en terminé par la méthode d'Euler implicite et la méthode de Cholesky pour calculer la solution de ce problème.
- ✓ **Problème inverse non linéaire** : nous avons proposé un algorithme basé sur la méthode de Levenberg-Marquardt et la méthode des éléments finis pour estimer un coefficient dans un problème non linéaire gouverné par une équation de la chaleur dans un milieu non homogène en dimension un avec condition initiale et des conditions aux limites de Dirichlet-Neumann. Le coefficient est rapproché par une forme polynomiale et l'algorithme numérique est employé pour trouver les coefficients de cet polynôme.

Comme perspectives, nous avons prévu les projets de recherches suivants :

- ☞ Identification numérique le paramètre a pour un problème hyperbolique suivant :

$$\begin{cases} \frac{\partial^2 u}{\partial t^2}(x, t) - \frac{\partial}{\partial x} \left(a(x) \frac{\partial u}{\partial x}(x, t) \right) = f(x, t), & 0 < x \leq 1, t > 0. \\ u(0, t) = 0, & t > 0 \\ \frac{\partial u}{\partial x}(1, t) = 0, & t > 0 \\ u(x, 0) = u_0(x), & 0 < x \leq 1 \\ \frac{\partial u}{\partial t}(x, 0) = u_0(x), & 0 < x \leq 1. \end{cases}$$

- ☞ Identification numérique la condition au limite $g(t)$ pour un problème de diffusion avec dérivée de temps fractionnaire de Caputo :

$$\begin{cases} \mathcal{D}_0^\alpha u(t) = u_{xx} + f(x, t), & 0 < x < 1, t > 0, 0 < \alpha < 1. \\ u(x, 0) = 0, & 0 \leq x \leq 1, \\ u(0, t) = 0, & t > 0, \\ u(1, t) = g(t), & t > 0. \end{cases}$$

Où f est une fonction donnée et u une mesure donnée de la solution.

Bibliographie

- [1] H. Abdelouaheb. Résolution numérique d'un problème inverse non linéaire en dimension un. Master's thesis, Université de Laghouat, Algérie, 2015.
- [2] Y. B. Bard. *Nonlinear Parameter Estimation*. Acad. Press, New York, 1974.
- [3] J. V. Beck and K. J. Arnold. *Parameter Estimation in Engineering and Science*. Wiley, New York, 1977.
- [4] H. BENZIANE. Résolution numérique d'un problème inverse non linéaire par des méthodes de type extragradient. Master's thesis, Université Mohamed Boudiaf de M'sila, Algérie, 2016.
- [5] J. Dennis and R. Schnabel. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Prentice Hall, 1983.
- [6] K. Levenberg. A method for the solution of certain non-linear problems in least squares. *Quart. Appl. Math.*, 2 :164–168, 1944.
- [7] D. W. Marquardt. An algorithm for least squares estimation of nonlinear parameters. *J. Soc. Ind. Appl. Math.*, 11 :431–441, 1963.
- [8] T. Min, X. Chen, Y. Sun, and Q. Huang. A numerical approach to solving an inverse heat conduction problem using the levenberg-marquardt algorithm. *Mathematical Problems in Engineering*, 2014 :11 pages, 2014.
- [9] J. J. Moré. The levenberg-marquardt algorithm : implementation and theory. in *Numerical Analysis, Lecture Notes in Mathematics*, 630 :105–116, 1977.
- [10] M. N. Özisik and H. R. B. Orlande. *Inverse heat transfer, fundamentals and applications*. Taylor & Francis, 2000.