



SCIENCES SUP

Cours et exercices corrigés

Master • Écoles d'ingénieurs

SIMULATION NUMÉRIQUE EN C++

*Ionut Danaila
Frédéric Hecht
Olivier Pironneau*

DUNOD

2-005-593-1

SIMULATION NUMÉRIQUE EN C++

Cours et exercices corrigés

Ionut Danaila

Maître de conférences

Frédéric Hecht

Professeur

Olivier Pironneau

Professeur, membre de l'Institut

DUNOD

Table des matières

AVANT-PROPOS

IX

CHAPITRE 1 • INTRODUCTION À LA SIMULATION NUMÉRIQUE

1

1.1 Simulation numérique : définition

1

1.2 Un exemple 1D : l'équation stationnaire de la chaleur

3

1.2.1 Mise en équation

3

1.2.2 Analyse mathématique

5

1.2.3 Discrétisation du problème

6

1.2.4 Algorithme de résolution

8

1.2.5 Programmation en C++

11

1.2.6 Compilation, exécution du programme et visualisation des résultats

12

1.2.7 Commentaires et exercices

14

1.3 Analyse numérique de l'équation instationnaire de la chaleur

16

1.3.1 Équation adimensionnée

16

1.3.2 Discrétisation explicite en temps

17

1.3.3 Solution numérique de l'équation instationnaire

20

1.4 Compléments (1)

24

1.4.1 Consistance, stabilité et convergence d'un schéma aux différences finies

24

1.4.2 Stabilité au sens de Von Neumann

28

1.5 Compléments (2)

31

1.5.1 L'équation de convection

31

1.5.2	L'équation des ondes	37
1.5.3	Nature mathématique des EDP	41
CHAPITRE 2 • POUR COMMENCER À PROGRAMMER EN C++		46
2.1	Notions de base	46
2.1.1	Fonctions et passage d'arguments aux fonctions	46
2.1.2	Pointeurs et tableaux	52
2.1.3	Synthèse des déclarations C++	55
2.1.4	Structures de contrôle	56
2.1.5	Exercices	57
2.2	Classes et objets	57
2.2.1	Définition des classes	58
2.2.2	Un premier exemple : la classe vide	59
2.2.3	Création d'objets	61
2.3	Programmation générique (templates)	64
2.3.1	Fonctions templates	64
2.3.2	Classes templates	66
2.4	Héritage et polymorphisme	69
2.4.1	Classes dérivées	69
2.4.2	Fonctions virtuelles	72
2.4.3	Conversion de pointeurs	75
2.4.4	Protection des membres et classes amies	76
2.5	Astuces de programmation	77
2.5.1	Quelques règles à respecter	77
2.5.2	Utilisation du débogueur gdb	80
2.5.3	Vérificateur d'allocation	81
2.5.4	Utilisation d'un fichier Makefile	83
2.5.5	Passage de paramètres sous Unix	84
CHAPITRE 3 • ALGORITHMES ET TECHNIQUES AVANCÉES DE PROGRAMMATION		86
3.1	Modélisation des vecteurs, matrices, tenseurs	86
3.1.1	Version simple	87
3.1.2	Les classes RNM	93
3.1.3	Utilisation des classes RNM	94
3.1.4	Application : la méthode du gradient conjugué	97

3.2	Chaînes et listes chaînées	106
3.2.1	Construction de l'image réciproque d'une fonction	106
3.2.2	Stockage Morse d'une matrice creuse	110
3.3	Un algorithme de tri : heapsort	114
CHAPITRE 4 • LA MÉTHODE DES ÉLÉMENTS FINIS P^1		116
4.1	L'équation de la chaleur en 2D	116
4.1.1	Limite de validité du modèle 1D	116
4.1.2	Équation 2D et ses conditions aux limites	117
4.1.3	Modèle mathématique	118
4.2	Résolution numérique	119
4.2.1	Discrétisation en temps	119
4.2.2	Formulation variationnelle	119
4.2.3	Discrétisation en espace par éléments finis P^1	121
4.2.4	Formulation variationnelle discrète	125
4.3	Algorithme de résolution	126
4.3.1	Calcul des intégrales élémentaires	126
4.3.2	Résolution par gradient conjugué	128
4.4	Construction du maillage - le logiciel FreeFem++	131
4.4.1	Utilisation de FreeFem++	131
4.4.2	Fichier triangulation	132
4.5	Les classes de base pour les éléments finis	134
4.5.1	La classe R2 (modélisation de \mathbb{R}^2)	135
4.5.2	La classe Label (numéros logiques)	137
4.5.3	La classe Vertex (modélisation des sommets)	138
4.5.4	La classe Triangle (modélisation des triangles)	139
4.5.5	La classe BoundaryEdge (modélisation des arêtes frontières)	140
4.5.6	La classe Mesh (modélisation du maillage)	142
4.6	Les programmes	144
4.6.1	Le script femGC.edp	145
4.6.2	Le programme sfemGC.cpp	145
4.7	Les résultats	152
4.7.1	Visualisation avec gnuplot	152
4.7.2	Calcul des lignes de niveau	153
4.7.3	Comparaison avec les résultats 1D	159

4.8	Compléments (1)	161
4.8.1	Script FreeFem++ pour l'ailette 2D	161
4.9	Compléments (2)	163
4.9.1	Construction des arêtes d'un maillage	163
4.9.2	Construction des triangles contenant un sommet donné	165
4.9.3	Calcul explicite de la matrice $\mathcal{A}^{(\alpha,\beta,\alpha_R)}$	166
4.9.4	Construction par coloriage de la structure Morse de $\mathcal{A}^{(\alpha,\beta,\alpha_R)}$	168
CHAPITRE 5 • VISUALISATION 3D		170
5.1	Graphiques 3D	170
5.1.1	Coordonnées homogènes	171
5.1.2	Perspective	172
5.1.3	Clipping	176
5.1.4	Élimination des faces cachées : algorithme du Z-buffer	177
5.2	Couleurs	177
5.3	Réalisme	178
5.3.1	Dithering	178
5.3.2	Aliasing	179
5.3.3	Algorithme de Gouraud	179
5.4	La librairie OpenGL	179
5.4.1	Première application : affichage du maillage	180
5.4.2	Deuxième application : affichage 3D de la solution	184
CHAPITRE 6 • CONSTRUCTION D'UN MAILLAGE BIDIMENSIONNEL		192
6.1	Bases théoriques	192
6.1.1	Notations	192
6.1.2	Introduction	193
6.1.3	Maillage de Delaunay-Voronoi	195
6.1.4	Forçage de la frontière	203
6.1.5	Recherche de sous-domaines	205
6.1.6	Génération de points internes	206
6.2	Algorithme de construction du maillage	207
6.3	Programme C++	208

CHAPITRE 7 • UNE MÉTHODE INTÉGRALE TRIDIMENSIONNELLE	233
7.1 Solution élémentaire du laplacien	234
7.2 Représentation intégrale des fonctions à laplacien nul	235
7.3 Implications algorithmiques	236
7.3.1 Implémentation de l'étape 1	237
7.3.2 Implémentation de l'étape 2	237
7.3.3 Implémentation de l'étape 3	239
7.3.4 Calcul des intégrales	239
7.4 Programmation C++	240
7.4.1 Lecture de la triangulation	240
CHAPITRE 8 • DIFFÉRENTIATION AUTOMATIQUE	247
8.1 Le mode direct	247
8.2 Fonctions de plusieurs variables	250
8.3 Une bibliothèque de classes pour le mode direct	251
8.4 Principe de programmation	251
8.5 Implémentation comme bibliothèque C++	252
8.6 Dérivée d'ordre supérieur	254
8.7 DA d'un programme FORTRAN	258
8.8 Compléments	260
8.8.1 DA des fonctions à une variable : la librairie complète	260
CHAPITRE 9 • DE C++ À JAVA	266
9.1 Introduction	266
9.2 Un premier programme Java	266
9.2.1 L'environnement de programmation	267
9.3 Javascript	269
9.4 Interface Utilisateur Java	271
9.5 Applets	272
9.6 Une application plus compliquée	275
9.7 Compléments	282
9.7.1 Communications java C/C++	282

CHAPITRE 10 • CRÉATION D'UN LANGAGE UTILISATEUR	285
10.1 Algèbre de fonctions	285
10.1.1 Version de base	285
10.1.2 Les fonctions C^∞	287
10.2 Grammaire	291
10.2.1 Alphabets, langages, grammaires	291
10.2.2 Grammaire LL (1)	294
10.2.3 Analyseur grammatical bison/yacc	301
10.3 Le langage utilisateur	304
10.4 Un langage utilisateur pour les éléments finis	311
CHAPITRE 11 • LES PROJETS	316
11.1 Projet 1 : ailette de refroidissement avec radiation	316
11.2 Projet 2 : simulation d'un alternateur électrique	318
11.2.1 Introduction	318
11.2.2 Formulation du problème et résolution	319
11.2.3 La géométrie	322
11.2.4 Résumé de la marche à suivre	325
11.2.5 Bibliographie du projet	327
11.3 Projet 3 : éléments finis en axisymétrie	327
11.3.1 Généralités	327
11.3.2 Restriction	327
11.3.3 Le cas cylindrique	328
11.3.4 Le Projet	328
11.4 Projet 4 : le cas tridimensionnel général	328
11.5 Préparation aux deux projets précédents	329
BIBLIOGRAPHIE	331
INDEX	333