



**Algorithmes de graphes**  
 Philippe Lacomme, Christian Prins, Marc Sevaux  
 Lacomme, Philippe  
 Prins, Christian (1956-....)  
 Sevaux, Marc  
 Eyrolles, Paris  
**Algorithmes**  
 ISBN: 2-212-11385-4

## Table des Matières

Algorithmes de graphes

2e édition 2003

Philippe Lacomme

Christian Prins - Marc Sevaux

Eyrolles

<b>Avant-propos</b>	1
<b>Chapitre 1: Introduction aux graphes</b>	7
<b>1.1 Intérêt des graphes et applications</b>	7
<b>1.2 Graphes orientés</b>	8
1.2.1 Les p-graphes	8
1.2.2 Les l-graphes	9
<b>1.3 Graphes non-orientés</b>	10
<b>1.4 Parties de graphes</b>	12
<b>1.5 Parcours et connexité</b>	13
1.5.1 Définitions sur les parcours	13
1.5.2 Connexité et forte connexité	14
1.5.3 Parcours eulériens et hamiltoniens	15
<b>1.6 Quelques graphes particuliers</b>	16
<b>1.7 Références</b>	19
<b>Chapitre 2: Complexité des algorithmes et problèmes difficiles</b>	21
<b>2.1 Introduction</b>	21
<b>2.2 Notions sur la complexité des algorithmes</b>	22
2.2.1 Comment mesurer l'efficacité?	22
2.2.2 Taille d'une donnée	22
2.2.3 Ordre d'une fonction	23
2.2.4 Bons et mauvais algorithmes	25
<b>2.3 Problèmes d'optimisation combinatoire</b>	26
2.3.1 Définitions	26
2.3.2 Problèmes combinatoires faciles et difficiles	28
2.3.3 Exemples de problèmes faciles	29
2.3.4 Problèmes difficiles	32
<b>2.4 Notions de la théorie de la complexité</b>	36

2.4.1 La théorie de la complexité	36
2.4.2 Les classes P et NP	36
2.4.3 Les problèmes NP-complets	38
2.4.4 Impact sur l'approche pratique d'un problème réel	40
<b>2.5 Références</b>	41
<b>Chapitre 3: Résolution des problèmes difficiles</b>	43
<b>3.1 Introduction</b>	43
<b>3.2 Heuristiques</b>	43
3.2.1 Présentation générale	43
3.2.2 Evaluation des heuristiques	44
3.2.3 Heuristiques constructives	49
3.2.4 Recherches locales	51
3.2.5 Métaheuristiques	53
<b>3.3 Méthodes arborescentes</b>	59
3.3.1 Principes généraux	59
3.3.2 Séparation	59
3.3.3 Evaluation	60
3.3.4 Stratégie d'exploration	61
3.3.5 Arrêt et performances	63
3.3.6 Implémentation des méthodes arborescentes	63
3.3.7 Un exemple: le sac à dos en 0-1	65
<b>3.4 Références</b>	66
<b>Chapitre 4: Implémentation objet des graphes</b>	69
<b>4.1 Introduction</b>	69
<b>4.2 Les concepts objet de base</b>	70
4.2.1 Notion de classe	70
4.2.2 La généricité en Delphi	77
<b>4.3 Les concepts objet avancés</b>	81
4.3.1 L'héritage	81
4.3.2 Détails sur la création d'une instance	84
4.3.3 Les méthodes virtuelles et abstraites	85
4.3.4 La notion de "property"	88
4.3.5 Les méthodes de classe	89
<b>4.4 Proposition d'une implémentation objet des graphes</b>	90
4.4.1 Remarques sur le pseudo-langage utilisé	90
4.4.2 Organisation en unités	91
4.4.3 Les différentes classes	91
4.4.4 Les structures de données utilisées	92
4.4.5 L'unité U_Type_Graphe	95
4.4.6 L'unité U_Base_Graphe	98
4.4.7 L'unité U_Graphe	109
<b>4.5 Les manipulations de base sur les graphes</b>	121

4.5.1 Objectifs et principes	121
4.5.2 La classe T_Graphe	121
4.5.3 La classe T_Graphe_Matriciel	122
4.5.4 La classe T_Graphe_Liste	128
<b>4.6 Un exemple d'utilisation</b>	132
<b>4.7 Remarques et références</b>	133
<b>Chapitre 5: Explorations de graphes, composantes connexes et bipartisme</b>	135
<b>5.1 Introduction</b>	135
<b>5.2 Construction des listes de prédécesseurs</b>	136
5.2.1 Définition du problème et intérêt	136
5.2.2 Algorithme	137
5.2.3 La méthode BuildPreds	138
5.2.4 Exemple d'utilisation	139
<b>5.3 Décomposition d'un graphe en niveaux</b>	140
5.3.1 Définition du problème et applications	140
5.3.2 Algorithme	141
5.3.3 La méthode GetLayers	142
5.3.4 Exemple: calculs sur un ordinateur parallèle	144
<b>5.4 Exploration de graphes</b>	145
5.4.1 Forme générale d'un algorithme d'exploration	145
5.4.2 Exploration en largeur	147
5.4.3 Exploration en profondeur	147
5.4.4 Applications simples de l'exploration	148
5.4.5 Les méthodes BFS, DFS, GetCircuit et GetPath	150
5.4.6 Exemple d'utilisation	154
<b>5.5 Composantes connexes</b>	155
5.5.1 Le problème et ses applications	155
5.5.2 La méthode AllCC	156
5.5.3 Exemple d'utilisation	157
<b>5.6 Composantes fortement connexes</b>	159
5.6.1 Le problème et ses applications	159
5.6.2 Un algorithme simple	160
5.6.3 Algorithme de Tarjan	161
5.6.4 La méthode AllSCC	163
5.6.5 Etude de cas: évaluation des deux algorithmes	165
<b>5.7 Test de bipartisme</b>	168
5.7.1 Le problème et ses applications	168
5.7.2 Algorithme	170
5.7.3 La méthode Bipartite	171
5.7.4 Exemple d'utilisation	172
<b>5.8 Références</b>	173
<b>Chapitre 6: Problèmes de chemins optimaux</b>	175

<b>6.1 Introduction</b>	175
<b>6.2 Les problèmes de chemins optimaux</b>	176
6.2.1 Les grands types de problèmes	176
6.2.2 Les deux familles d'algorithmes	177
6.2.3 Liste des principaux algorithmes	178
6.2.4 Exemples d'applications	179
6.2.5 Graphe-exemple et notations	181
<b>6.3 Algorithmes à fixation d'étiquettes</b>	182
6.3.1 Algorithme de Dijkstra	182
6.3.2 Algorithme de Sedgewick et Vitter	188
6.3.3 Algorithmes à buckets	189
<b>6.4 Algorithmes à correction d'étiquettes</b>	194
6.4.1 Algorithme de Bellman	194
6.4.2 Algorithme FIFO	197
6.4.3 Algorithme de D'Esopo et Pape	199
6.4.4 Algorithme de Floyd	201
<b>6.5 Application en ordonnancement</b>	206
6.5.1 Algorithme de Bellman et graphes sans circuits	206
6.5.2 Ordonnements et plus longs chemins	208
6.5.3 La méthode Schedule	209
6.5.4 Exemple	210
<b>6.6 Evaluation des algorithmes</b>	212
<b>6.7 Références</b>	214
<b>Chapitre 7: Problèmes de flots et couplages</b>	215
<b>7.1 Introduction</b>	215
<b>7.2 Problème du flot maximal</b>	217
7.2.1 Concepts de base	217
7.2.2 Formulation en termes de programme linéaire	218
7.2.3 Exemples d'applications des flots	221
7.2.4 Algorithme de Ford et Fulkerson	223
7.2.5 Algorithme d'échelonnement des capacités	235
7.2.6 Algorithme des distances estimées au puits	236
7.2.7 Problèmes de flots compatibles	246
7.2.8 Comparaison des algorithmes	248
<b>7.3 Problèmes de flot de coût minimal</b>	250
7.3.1 Formulation du problème	250
7.3.2 Graphe d'écart - Condition d'optimalité	251
7.3.3 Exemples d'applications	251
7.3.4 Algorithme de Busacker et Gowen	253
<b>7.4 Problèmes de couplages</b>	261
7.4.1 Définitions et principaux problèmes	261
7.4.2 Principes généraux des algorithmes de couplage	262

7.4.3 Exemples d'applications	264
7.4.4 Algorithme pour le couplage biparti maximal	265
7.4.5 Un algorithme pour le problème d'affectation	269
7.4.6 Evaluation des algorithmes de couplage	270
<b>7.5 Références</b>	275
<b>Chapitre 8: Arbres et arborescences</b>	277
<b>8.1 Introduction</b>	277
<b>8.2 Définitions - Enoncés des problèmes</b>	278
8.2.1 Arbres et arborescences	278
8.2.2 Le problème de l'arbre de poids minimal	279
8.2.3 Le problème de l'arborescence de poids minimal	279
<b>8.3 Exemples d'applications</b>	279
8.3.1 Construction de lignes électriques	279
8.3.2 Réseau d'irrigation par gravité	280
<b>8.4 Le problèmes de l'ARPM</b>	280
8.4.1 Théorème d'optimalité des ARPM	280
8.4.2 Algorithme de Prim	281
8.4.3 Algorithme de Kruskal	285
8.4.4 Utilisation et comparaison des méthodes Prim et Kruskal	294
<b>8.5 Arborescence de poids minimal</b>	295
8.5.1 Principes de l'algorithme d'Edmonds	295
8.5.2 Un exemple	298
8.5.3 Une implémentation efficace	299
8.5.4 La méthode Edmonds	303
8.5.5 Exemple d'utilisation	306
<b>8.6 Références</b>	307
<b>Chapitre 9: Parcours eulériens et hamiltoniens</b>	309
<b>9.1 Introduction</b>	309
<b>9.2 Parcours eulériens et chinois</b>	310
9.2.1 Définitions et conditions d'existence	310
9.2.2 Parcours eulérien d'un graphe non orienté	312
9.2.3 Parcours eulérien d'un graphe orienté	317
9.2.4 Exemple d'utilisation d'EulerChain et EulerPath	320
9.2.5 Postier chinois non orienté	322
9.2.6 Postier chinois orienté	324
<b>9.3 Le problème du voyageur de commerce</b>	330
9.3.1 Le PVC comme archétype du problème difficile	330
9.3.2 Exemples d'application du PVC	331
9.3.3 Heuristiques gloutonnes	332
9.3.4 Recherches locales	337
9.3.5 Méthode de recuit simulé	340
9.3.6 Méthode tabou	342

9.3.7 Evaluation des heuristiques pour le PVC	346
<b>9.4 Références</b>	350
<b>Chapitre 10: Problèmes de coloration</b>	351
<b>10.1 Introduction</b>	351
<b>10.2 Généralités sur la coloration de graphes</b>	352
10.2.1 Définition des problèmes et complexité	352
10.2.2 Quelques propriétés simples	352
<b>10.3 Deux exemples d'applications</b>	354
10.3.1 Problèmes d'emploi du temps	354
10.3.2 Carrés latins	354
<b>10.4 Heuristiques séquentielles</b>	355
10.4.1 Méthode générique	355
10.4.2 Deux bons ordres	355
10.4.3 L'algorithme DSatur	356
10.4.4 Les méthodes Delphi	357
<b>10.5 Méthode exacte</b>	362
10.5.1 Principes	362
10.5.2 La méthode Backtrack	365
<b>10.6 Méthode de recuit simulé</b>	367
<b>10.7 Recherche tabou</b>	369
10.7.1 Principes	370
10.7.2 La méthode TabuCol	370
<b>10.8 Evaluation des méthodes de coloration</b>	373
<b>10.9 Références</b>	378
<b>Annexe 1: CD-Rom d'accompagnement</b>	379
<b>Structure du CD-Rom</b>	379
<b>Installation des environnements de développement</b>	379
Description de la partie Windows	379
Description de la partie Linux	386
Propriétés des environnements de développement et conditions d'utilisation	387
Configuration des environnements de développement	387
<b>Installation du code source</b>	391
Description de la partie Windows	391
Description de la partie Linux	391
<b>Utilisation du code source</b>	391
<b>Utilisation de graph_master.exe</b>	393
<b>Informations supplémentaires sur Delphi/Kylix</b>	394
Copyright (C) 2003 (Lacomme, Prins, Sevaux)	395
<b>Site Web des auteurs</b>	395
<b>Annexe 2: Bibliographie</b>	397
<b>Index</b>	405

