

A Algorithmes de graphes

Philippe Lacomme
Christian Prins - Marc Sevaux



Une véritable boîte à outil logicielle pour la
résolution de vos problèmes de graphes !

- ▶ Sources en langage Delphi (compatible Kylix) de tous les algorithmes du livre
- ▶ Fichiers de données des exemples d'application
- ▶ Exécutable permettant d'appliquer les algorithmes à ses propres jeux de données sans programmation
 - ▶ Borland Delphi 7 Personnel (pour Windows)
 - ▶ Borland Kylix 3 Edition Open (pour Linux)

EYROLLES

2-005-622-1



2-005-622-1

Algorithmes de graphes

Philippe Lacomme
Christian Prins - Marc Sevaux

2^e édition 2003

EYROLLES

Table des matières

AVANT-PROPOS.....	1
--------------------------	----------

CHAPITRE 1 : INTRODUCTION AUX GRAPHERS.....	7
--	----------

1.1 Intérêt des graphes et applications	7
1.2 Graphes orientés.....	8
1.2.1 Les p-graphes	8
1.2.2 Les 1-graphes	9
1.3 Graphes non-orientés	10
1.4 Parties de graphes	12
1.5 Parcours et connexité	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
1.6 Quelques graphes particuliers.....	16
1.7 Références	19

CHAPITRE 2 : COMPLEXITE DES ALGORITHMES ET PROBLEMES DIFFICILES.....	21
---	-----------

2.1 Introduction	21
2.2 Notions sur la complexité des algorithmes	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
2.3 Problèmes d'optimisation combinatoire.....	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
2.4 Notions de la théorie de la complexité	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
2.5 Références	41

CHAPITRE 3 : RESOLUTION DES PROBLEMES DIFFICILES43

3.1 Introduction.....	43
3.2 Heuristiques.....	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
3.3 Méthodes arborescentes.....	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
3.4 Références.....	66

CHAPITRE 4 : IMPLEMENTATION OBJET DES GRAPHES69

4.1 Introduction.....	69
4.2 Les concepts objet de base.....	70
4.2.1 Notion de classe.....	70
4.2.2 La généricité en Delphi.....	77
4.3 Les concepts objet avancés.....	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
4.4 Proposition d'une implémentation objet des graphes.....	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
4.5 Les manipulations de base sur les graphes.....	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
4.6 Un exemple d'utilisation.....	132
4.7 Remarques et références.....	133

CHAPITRE 5 : EXPLORATIONS DE GRAPHES, COMPOSANTES CONNEXES ET BIPARTISME135

5.1 Introduction.....	135
------------------------------	------------

5.2 Construction des listes de prédécesseurs	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
5.3 Décomposition d'un graphe en niveaux.....	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
5.4 Exploration de graphes	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
5.5 Composantes connexes.....	155
5.5.1 Le problème et ses applications.....	155
5.5.2 La méthode AllCC	156
5.5.3 Exemple d'utilisation	157
5.6 Composantes fortement connexes	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
5.7 Test de bipartisme	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
5.8 Références	173

CHAPITRE 6 : PROBLEMES DE CHEMINS OPTIMAUX

175

6.1 Introduction	175
6.2 Les problèmes de chemins optimaux	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
6.3 Algorithmes à fixation d'étiquettes	182
6.3.1 Algorithme de Dijkstra.....	182
6.3.2 Algorithme de Sedgewick et Vitter	188
6.3.3 Algorithmes à buckets.....	189
6.4 Algorithmes à correction d'étiquettes.....	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
6.5 Application en ordonnancement	206
6.5.1 Algorithme de Bellman et graphes sans circuits	206
6.5.2 Ordonnancements et plus longs chemins	208
6.5.3 La méthode Schedule	209
6.5.4 Exemple	210
6.6 Evaluation des algorithmes	212
6.7 Références	214

CHAPITRE 7 : PROBLEMES DE FLOTS ET COUPLAGES

7.1 Introduction	215
7.2 Problème du flot maximal	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
7.3 Problèmes de flot de coût minimal	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
7.4 Problèmes de couplages	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
7.5 Références	275

CHAPITRE 8 : ARBRES ET ARBORESCENCES

8.1 Introduction	277
8.2 Définitions – Énoncés des problèmes	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
8.3 Exemples d'applications	279
8.3.1 Construction de lignes électriques	279
8.3.2 Réseau d'irrigation par gravité	280
8.4 Les problèmes de l'ARPM	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
8.5 Arborescence de poids minimal	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
8.6 Références	307

CHAPITRE 9 : PARCOURS EULERIENS ET HAMILTONIENS309

9.1 Introduction	309
9.2 Parcours eulériens et chinois	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
9.3 Le problème du voyageur de commerce.....	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
9.4 Références	350

CHAPITRE 10 : PROBLEMES DE COLORATION351

10.1 Introduction	351
10.2 Généralités sur la coloration de graphes	352
10.2.1 Définition des problèmes et complexité	352
10.2.2 Quelques propriétés simples.....	352
10.3 Deux exemples d'applications	354
10.3.1 Problèmes d'emploi du temps	354
10.3.2 Carrés latins.....	354
10.4 Heuristiques séquentielles.....	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
10.5 Méthode exacte	362
10.5.1 Principes.....	362
10.5.2 La méthode Backtrack.....	365
10.6 Méthode de recuit simulé.....	367
10.7 Recherche tabou	369
10.7.1 Principes.....	370
10.7.2 La méthode TabuCol.....	370
10.8 Evaluation des méthodes de coloration	373
10.9 Références	378

ANNEXE 1 : CD-ROM D'ACCOMPAGNEMENT	379
Structure du CD-Rom	379
Installation des environnements de développement.....	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
Installation du code source.....	391
Description de la partie Windows.....	391
Description de la partie Linux.....	391
Utilisation du code source.....	391
Utilisation de graph_master.exe	393
Informations supplémentaires sur Delphi/Kylix	394
Copyright (C) 2003 (Lacomme, Prins, Sevaux).....	395
Site Web des auteurs.....	395
ANNEXE 2 : BIBLIOGRAPHIE.....	397
INDEX.....	405