IFT 3512: TECHNIQUES D'OPTIMISATION II (hiver 2002)

Professeur: Patrice MARCOTTE

Objectifs généraux

La programmation mathématique regroupe un ensemble de techniques qui s'appliquent non seulement en recherche opérationnelle, mais également dans une foule de domaines de l'informatique: téléinformatique, vision par ordinateur, infographie, bioinformatique, architecture des ordinateurs, etc. Ce cours s'adresse à tous les étudiants qui s'intéressent à l'optimisation et que ne rebute pas un peu de développements mathématiques. Il traite principalement de programmation non linéaire et constitue donc une extension naturelle de la matière couverte dans le cours "Optimisation I". Il aborde également la branche de l'optimisation combinatoire qu'est la programmation en variables entières. Le site web du cours (iro.umontreal.ca/ marcotte/Ift3512/ift3512.html) comporte une liste de monographies d'intérêt divers ainsi que des notes de cours succinctes.

Plan de cours

1. Généralités (environ 1 semaine)
Notions de topologie élémentaire: ensembles ouverts, fermés, compacts, bornés, convexes, polyèdres et polytopes. Fonctions continues, différentiables, convexes. Gradient, jacobien, dérivées partielles, dérivées directionnelles, formule de Taylor. Mimima, maxima, points stationnaires. Conditions d'optimalité pour l'optimisation différentiable et sans contraintes.

2. Algorithmes (environ 1 semaine)
Définition d'un algorithme. Ordre et taux de convergence. Algorithmes fermés. Théorème de convergence globale.

3. Optimisation unidimensionnelle (environ 1 semaine)
Optimisation de fonctions unimodales: méthode de Fibonacci et de la section dorée, recherche dichotomique. Méthode de Newton et de la fausse position. Ajustement quadratique.

4. Optimisation multidimensionnelle sans contraintes (environ 2 semaines)
Méthode de la plus forte pente et inégalité de Kantorovitch. Méthode de Newton multidimensionnelle. Algorithmes de Jacobi et de Gauss-Seidel. Règles d'arrêt pour la recherche unidimensionnelle. Méthodes de gradient conjugué et quasi-Newton.

5. Programmation linéaire (environ 2 semaines)
Rappels de programmation linéaire: algorithme du simplexe, dualité, dégénérescence. Algorithmes de points intérieurs: méthodes primales et primales-duales.

6. Optimisation avec contraintes: méthodes primales (environ 3 semaine)
Conditions d'optimalité de Kuhn et Tucker. Algorithmes primaux: Zoutendijk, Frank-Wolfe, gradient réduit, simplexe convexe, gradient projeté (Rosen et Goldstein-Levitin-Polyak), méthode des contraintes actives.

7. Optimisation avec contraintes: méthodes duales (environ 2 semaines)
Méthodes de pénalités et de barrières. Méthodes primales duales. Méthode des multiplicateurs.

8. Programmation en variables entières (environ 2 semaines)
Formulations, relaxations et bornes. Techniques d'énumération implicite (Branch and Bound) et coupes de Gomory. Génération de colonnes. Programmation à deux niveaux.

Évaluation

Intra: 30% – Final: 40% – Devoirs: 30% - Seuil: 50%