

RCP104 - Optimisation en informatique

🌟 Valide le 24-04-2019

Présentation

Code : RCP104

Prérequis

6 crédits

Élèves ingénieurs, étudiants de master M1.

Prérequis : avoir des connaissances de base en algorithmique, réseaux informatiques, programmation, graphes et recherche opérationnelle.

Responsabilité nationale :
EPN05 - Informatique / Cédric BENTZ

Objectifs pédagogiques

A partir de problèmes concrets en informatique (majoritairement, mais pas exclusivement, issus des réseaux de télécommunication), apprendre à traiter des problèmes difficiles de la recherche opérationnelle : savoir écrire un modèle mathématique et proposer des méthodes, optimales ou non (mais efficaces malgré tout), utilisant des outils pratiques pour résoudre ces problèmes (méthodes heuristiques, programmation linéaire et logiciels).

Contact national :
EPN05 - Informatique

2 rue Conté

75003 Paris

01 40 27 22 58

Swathi Rajaselvam

swathi.ranganadin@cnam.fr

Compétences

L'étudiant ayant suivi cet enseignement sait reconnaître et modéliser un problème de recherche opérationnelle. Il sait le résoudre avec des outils simples. Il sait en particulier aborder certains problèmes d'optimisation combinatoire dans les réseaux informatiques.

Programme

Contenu

1- Présentation de l'ensemble du cours à partir d'un problème d'optimisation concret (localisation). Le problème est-il difficile (du point de vue de la complexité) ? Si oui, comment écrire un modèle mathématique ? Ce modèle permet-il d'obtenir de façon suffisamment efficace une solution optimale à l'aide d'un logiciel ? Si oui, l'étude est terminée. Sinon, comment obtenir une solution approchée et comment valider la solution trouvée ?

2- Apprendre à écrire un programme mathématique : choisir les variables, déterminer leurs domaines, écrire l'objectif et les contraintes. Particularité des modèles en variables binaires ou entières. Travail sur des "cas d'école" : partition de graphes (clustering), coloration (planification), etc.

Application à divers problèmes réels : dimensionnement/conception de réseaux, routage multicast dans les réseaux, placement de copies de fichiers, etc.

3- Apprendre à transformer un problème d'optimisation non linéaire en un programme linéaire de façon à pouvoir utiliser les logiciels. Techniques de linéarisation, prise en compte de rapports ou de produits de variables, etc.

4- Résolution approchée de problèmes difficiles par des méthodes générales (recuit simulé, méthode tabou, algorithmes génétiques, etc.) ou par des méthodes spécifiques (heuristiques ad-hoc). Validation des résultats obtenus par les heuristiques à l'aide de bornes basées, par exemple, sur la résolution du problème (ou d'une relaxation) par un solveur (ou logiciel de résolution).

5- Utilisation d'un solveur libre d'accès (par exemple, GLPK) par le biais d'un modèleur (GMPL) ou du format de fichier LP. Mise en oeuvre sur ordinateur pendant certaines séances. Rappel des principes de la programmation linéaire, et introduction aux techniques de résolution de programmes linéaires en nombres entiers.

6- Étude d'un cas réel, sous la forme d'un projet ou d'un (ou plusieurs) TP noté.

Description des modalités de validation

Examen noté sur 16.

Projet ou TP noté sur 4.

Bibliographie

Titre	Auteur(s)
Optimisation discrète (Dunod)	Alain Billionnet
Algorithmes de graphes (Eyrolles)	Philippe Lacomme, Christian Prins, Marc Sevaux
Métaheuristiques pour l'optimisation difficile (Eyrolles)	Johann Dréo, Alain Pétrowski, Patrick Siarry, Eric Taillard
Réseaux informatiques : conception et optimisation	Malek Rahoual et Patrick Siarry