Subject Information Guide

Complexity theory and polynomial approximation of NP-hard problems

Semester 2, 2018

Administration and contact details

Host Department	School of Science (Mathematical Sciences)
Host Institution	RMIT University
Name of lecturer	Marc Demange
Phone number	0431 331 093
Email Address	Marc.demange@rmit.edu.au
Homepage	https://www.rmit.edu.au/contact/staff-contacts/academic-staff/d/demange-associate-professor-marc
Name of Honours coordinator	Marc Demange
Phone number	03 9925 2385
Email Address	marc.demange@rmit.edu.au

Subject details

Handbook entry URL	NA
Subject homepage URL	NA
Honours student hand-out URL	NA
Start date:	16 July, 2018
End date:	11 October, 2018
Contact hours per week:	2
Lecture day and time:	To be confirmed
Description of electronic access arrangements for students (for example, WebCT)	NA

Subject content

1. Subject content description

This course will introduce students of applied or pure mathematics, operations research and/or computer science, to the topics of computational complexity and polynomial approximation of NP-hard problems. Both topics can be seen as a way to mathematically characterise the possibility to solve efficiently - in particular in polynomial time - a large-scale decision /optimisation problem. For optimisation problems that are hard in this sense (called NP-hard), polynomial approximation aims to design polynomial algorithms with absolute guarantees on the quality of the solution. It also gives tools to better analyse/understand the structure of the class of NP-hard problems, classifying them with respect to their intrinsic complexity.

This subject is at the interface between mathematics and theoretical computer science. Even though a theoretical topic, it has revealed over the last decades to have a huge impact for the practical resolution of discrete optimisation problems. As a consequence, it is important for anybody who devises algorithms, models and solves large-scale decision or combinatorial optimisation problems, to have at least a basic knowledge about this topic and to acquire skills for evaluating the complexity of the problem he/she is working with. It is essential for validating a model, but also for the design of efficient algorithms. Additionally, studying the complexity of a problem on restricted classes of instances - as close as possible to the real instances one has to solve - allows to better understand patterns that make it hard and gives keys for better solving it.

The main notions we will work with are the most standard complexity classes, approximation classes, polynomial reductions, approximation preserving reductions and completeness, as well as the famous P=NP question. We will pay great attention to various examples of proofs of NP-completeness and analyses of approximation algorithms.

2. Week-by-week topic overview

NP-complete problems

Week 1. Introduction to the topic, decision problems, deterministic Turing machines, non-deterministic Turing machines, complexity of a Turing machine, Church's thesis.

Week 2. *The main notions***:** polynomial reductions, P vs NP classes, NP-complete problems, SAT and Cook's theorem. List of classical polynomial problems.

Week 3. Examples of NP-complete problems / proof of NP-completeness: 3-SAT, Vertex cover, Graph colouring and related problems

Proving NP-completeness results

Week 4. Examples of NP-completeness proves (cont.): 3-Dimensionnal matching, Hamiltonian circuit

Week 5. *Coping with number problems*: Partition, Knapsack, Integer linear programming, 3-partition. Pseudo-polynomial algorithms and Strong NP-completeness.

Week 6. Restricted instance classes: local replacement for problem's restriction (example of graph colouring in bounded degree graph and planar graphs), some hard restricted SAT problems.

Polynomial approximation algorithms

Week 7. Introduction to polynomial approximation, the class APX, example (Vertex cover, Bin packing, Metric TSP)

Week 8. *Refining APX*: PTAS and FPTAS, examples (Knapsack, Min Makespan, Euclidean TSP)

Week 9. *Beyond APX*: LogAPX, PolyAPX, examples (Set cover, Stable Set, Graph colouring).

Reductions, hardness results and structure of NP

Week 10. Introduction to approximation preserving reductions, examples

Week 11. Approximation hardness results and completeness in approximation

Week 12. A complementary approach: differential approximation theory

3. Assumed prerequisite knowledge and capabilities

No specific a priori knowledge in the topic of complexity is assumed.
Students will be assumed to be familiar with basic notions in discrete mathematics, graphs and algorithms.

4. Learning outcomes and objectives

Our main objective is to familiarise students with the notions in basic complexity theory and approximation theory. Students will acquire a working knowledge of the techniques used to characterise the complexity of a problem and to design and analyse approximation algorithms. At the completion of this course, students will be able to understand NP-completeness proofs and design basic ones. They will improve their skills in the design of optimisation models and their efficient solution. They will develop skills to discriminate NP-hard problems and polynomial ones and to evaluate the relevance of models and solution techniques in operations research. They will develop skills to design and analyse polynomial approximation algorithms and will become familiar with the notion of approximation preserving reductions and their possible use for the design of efficient algorithms and for analysing complexity. They will develop skills to analyse research articles and will be introduced to related research questions.

AQF specific Program Learning Outcomes and Learning Outcome Descriptors (if available):

AQF Program Learning Outcomes addressed in this subject	Associated AQF Learning Outcome Descriptors for this subject
Mathematics in Context	K1: coherent and advanced knowledge of the underlying principles and concepts in one or more disciplines K2: knowledge of research principles and methods
Problem Solving - You will have the ability to apply knowledge and skill to characterise, analyse and solve a wide range of problems.	S1: cognitive skills to review, analyse, consolidate and synthesise knowledge to identify and provide solutions to complex problem with intellectual independence S2: cognitive and technical skills to demonstrate a broad understanding of a body of knowledge and theoretical concepts with advanced understanding in some areas A2: to adapt knowledge and skills in diverse contexts
Teamwork and Project Management	S3: cognitive skills to exercise critical thinking and judgement in developing new understanding
Communication	S5: communication skills to present clear and coherent exposition of knowledge and ideas to a variety of audiences
Information Literacy	S3: cognitive skills to exercise critical thinking and judgement in developing new understanding

Learning Outcome Descriptors at AQF Level 8

Knowledge

K1: coherent and advanced knowledge of the underlying principles and concepts in one or more disciplines

K2: knowledge of research principles and methods

Skills

S1: cognitive skills to review, analyse, consolidate and synthesise knowledge to identify and provide solutions to complex problem with intellectual independence

S2: cognitive and technical skills to demonstrate a broad understanding of a body of knowledge and theoretical concepts with advanced understanding in some areas

S3: cognitive skills to exercise critical thinking and judgement in developing new understanding

S4: technical skills to design and use in a research project

S5: communication skills to present clear and coherent exposition of knowledge and ideas to a variety of audiences

Application of Knowledge and Skills

A1: with initiative and judgement in professional practice and/or scholarship

A2: to adapt knowledge and skills in diverse contexts

A3: with responsibility and accountability for own learning and practice and in collaboration with others within broad parameters

A4: to plan and execute project work and/or a piece of research and scholarship with some independence

5. Learning resources

AUSIELLO G., CRESCENZI P., GAMBOSI G., KANN V.,MARCHETTI-SPACCAMELA, A., PROTASI M., Complexity and approximation. Combinatorial optimization problems and their approximability properties, Springer, Berlin, 1999.

GAREY M. R., JOHNSON D. S., Computers and intractability. A guide to the theory of NP-completeness, W. H. Freeman, San Francisco, 1979.

PASCHOS, V. (Ed.), Paradigms of Combinatorial Optimization (Problems and New Approaches, London - Hoboken (UK - USA), ISTE – WILEY, 2014 (2nd Edition).

Complementary documents and in particular research articles will be provided, as well as exercises.

6. Assessment

Exam/assignment/classwork breakdown					
Exam	**50%**	**Assignment**	**40%**	**Class work**	**10%**
Assignment due dates		**TBC**	**TBC**		
Approximate exam date				**TBC**	

Institution Honours program details

Weight of subject in total honours assessment at host department	**12.5%**
Thesis/subject split at host department	**25% thesis/75% course work**
Honours grade ranges at host department:	
H1	**80-100 %**
H2a	**75-79 %**
H2b	**70-74 %**
H3	**65-69 %**