IE 631: Integer Programming, Spring 2023

TTh(화목) 9:00-10:15 am, E2-2 (산업경영학동) #1122

Instructor: Dabeen Lee, `dabeenl@kaist.ac.kr`, E2-2 #2109.

Lectures: Tuesdays and Thursdays 9:00-10:15 am.

Office hours: To be announced.

Teaching assistant: To be announced.

Course webpage: `https://dabeenl.github.io/IE631` + KLMS course page

Assignments and lecture notes will be uploaded to the webpage as well as the KLMS page.

Course description Many real-world decision-making problems require decisions over a discrete set of available choices. For example, on/off constraints and decisions for inclusion/exclusion can be modeled with binary variables. Moreover, one would face scenarios where we choose a decision from disjunctive and disjoint sets of choice options. Integer programming provides a general and comprehensive framework to model such decision-making settings. This course will cover the fundamental subjects of integer programming that are relevant to both theory and practice. Students will learn the mathematical theory behind integer programming, integer programming formulations, cutting plane algorithms, and decomposition-based algorithms, all of which are implemented in state-of-the-art commercial software.

Key topics Theory and algorithms for integer programming. Combinatorial optimization problems.

- Theory: Integer Farkas Lemma, Polyhedral theory, Perfect formulations
- Cutting planes: Gomory-Chvátal inequalities, Cuts for mixed-integer programs, Branch-and-cut algorithm.
- Algorithms: Subgradient algorithm for Lagrangian reformulation, Danzig-Wolfe decomposition, Benders reformulation.
- Combinatorial optimization problems: Matching, Knapsack, TSP, Fixed-charge problem, Facility location.

Textbook We will use and closely follow the following textbook throughout the course.

- Integer Programming, *Conforti, Cornuéjols, and Zambelli*, `https://link.springer.com/book/10.1007/978-3-319-11008-0`

KAIST students should have access to download an electronic copy via the above link.

Prerequisites There are no formal prerequisites but you should be comfortable making mathematical arguments, writing proofs, and programming. You should also be comfortable with the background knowledge from previous courses. Such topics include

- linear programming and graph algorithms,
- basic linear algebra (vectors, matrices, inner products),
- basic complexity theory (time complexity, polynomial algorithms, P vs NP).

Assessment structure There will be bi-weekly assignments (50%), a course project (20%), and a final exam (30%). **Typesetting in Latex is required for all assignments**.

Course outline What follows is a tentative outline of this course.

1. Introduction to integer programming
    - Combinatorial optimization, Integer programs,
    - Cutting planes, Branch-and-bound algorithm.
2. Solving a system of equations with integer constraints
    - Hermite normal form,
    - Integer Farkas Lemma.
3. Integer programming formulations
    - Knapsack problem,
    - Propositional logic, Satisfiability problem, Logical inference,
    - Fixed-charge problem,
    - Union of polyhedra,
    - Set packing, Set covering,
    - Cluttery theory, Min-cut in a graph, st-path in a graph.
4. Polyhedral theory I
    - Fourier's elimination procedure, Farkas' Lemma
    - Minkowski-Weyl Theorem,
    - Recession cone, Affine hull,
    - Faces,
    - Stable set problem,
5. Polyhedral theory II

- Projection,
- Traveling salesman problem,
- Polyhedral ties between linear and integer programming.

6. Perfect formulations
   - Meyer's theorem,
   - Totally unimodularity,
   - Uncapacitated Lot Sizing Problem.
7. Chvátal-Gomory inequalities
   - Matching polytope,
   - Chvátal's integer rounding procedures,
   - Modular arithmetic and Gomory's fractional cuts,
   - Geometric view.
8. Cuts for mixed integer programs
   - Split inequalities,
   - Gomory's mixed integer cuts,
   - Lift-and-Project for mixed 0,1 programs
9. Decomposition-based iterative algorithms for integer programming.
   - Lagrangian relaxation and duality,
   - Subgradient algorithm for the Lagrangian relaxation problem,
   - Dantzig-Wolfe decomposition,
   - Benders reformulation.
10. Enumeration
    - Branch-and-cut algorithm.
11. Strong cutting planes and the separation problem
    - Ellipsoid algorithm and its consequences in combinatorial optimization,
    - Equivalence between optimization and separation.