COURSE OFFERED IN THE DOCTORAL SCHOOL

Code of the course	4606-ES-000BCHJ-0369	Name of the course	Polish	**Optymalizacja kombinatoryczna i całkowitoliczbowa**
			English	**Combinatorial and integer optimization**
Type of the course	specialized			
Course coordinator	prof. dr hab. inż. Radosław Pytlak	Course teacher	prof dr hab. inz. R. Pytlak	
Implementing unit	Faculty of Mathematics and Information Science	Scientific discipline / disciplines*	information and communication technology; automation, electronics, electrical engineering and space technologies mechanical engineering; mathematics	
Level of education	doctoral studies	Semester	winter	
Language of the course	English			

Type of assessment	grade	Number of hours in a semester	52	ECTS credits	4
Minimum number of participants	10	Maximum number of participants	45	Available for students (MSc)	Yes

Type of classes		Lecture	Auditory classes	Project classes	Laboratory	Seminar
Number of hours	in a week	2			2	
	in a semester	26			26	

* does not apply to the Researcher's Workshop

1. Prerequisites

basic knowledge of mathematical analysis and linear algebra

2. Course objectives

The main goal of the course is to familiarize students with modern combinatorial and integer optimization methods. Recent years have seen a significant acceleration in the use of integer optimization techniques in solving practical decision-making problems. This is due to the widespread availability of high-performance hardware, the intensive development of new algorithms, and the emergence of advanced modeling environments enabling the effective implementation of these mechanisms on an industrial scale.

The course also aims to present a rigorous, geometric approach to these problems, based on the foundations of linear programming. The course demystifies classical discrete mathematics problems, presenting them in a broad, unified context. This allows students to discover mathematical connections between seemingly isolated problems and learn to construct a general analytical framework for solving them.

This course will present classical methods for integer optimization problems, such as the branch-and-bound method, column generation method, and decomposition-based methods. Particular attention will be paid to advanced integration schemes, including the branch-cut-price technique and Benders' decomposition method, which are largely responsible for the recent growing interest in integer optimization. These mechanisms will be presented in the context of utilizing duality to construct optimality certificates, which allow for analytical and efficient search space reduction.

The course will also discuss selected issues related to solving convex (not necessarily linear) integer optimization problems. Particular emphasis will be placed on the use of innovative hybrid methods, in which a first-order method for a differentiable convex optimization problem provides a starting point for the actual integer method.

The course's laboratory sessions will utilize modern integer optimization programming environments (e.g., IBM ILOG CPLEX, Gurobi), which incorporate advanced problem modeling languages. These will enable students to effectively implement and verify modern integer optimization techniques and decomposition schemes using complex engineering problems as examples.

3. Course content (separate for each type of classes)
Lecture
1. Integer and combinatorial optimization problems. Dimension explosion in combinatorial problems. Alternative formulations of combinatorial and integer optimization problems. Optimality in integer and combinatorial optimization problems. Relaxation via linear programming problems as the main tool for generating optimality certificates. Combinatorial relaxation. Lagrangian relaxation. 2. Linear programming problems, primal and dual problems. Primal and dual simplex methods. Interior point methods for solving linear program-ming problems. Subgradient methods for solving dual problems. 3. Linear inequalities and polyhedra. Fourier elimination method. Farkas' lemma. Affine, convex, and conic combinations. Polyhedra and the Minkowski-Weyl Theorem. Minimal representations and facets. The decomposition theorem for polyhedra. 4. Networks and graphs. The shortest path problem. The transportation problem. Demystifying classical graph algorithms through the primal-dual algorithm and analyzing the properties of dual polyhedra. The simplex algo-rithm for flow problems in networks. 5. Perfect systems of linear inequalities for sets. Properties of integral polyhedra. Total unimodularity and its consequences for error-free continuous relaxation of discrete problems. The fundamental theorem of integer programming. Total dual integrity (TDI). Optimization in nonbipartite structures. The associative polyhedron as an example of a system guaranteeing discrete solutions despite the lack of total unimodularity. 6. Submodular polyhedra and the geometry of greedy algorithms. Construction of a matroid polyhedron. Submodular minimization and the matroid intersection theorem as a natural limit of polynomial-solvable problems. 7. Branch and bound method. Implicit enumeration in NP-hardness. The branch and bound method based on linear programming and dual certifi-cates. Application of systems for branch-bound/cut methods. The im-portance of modules for preprocessing and solution. 8. The method of cutting planes. Valid inequalities. Gomory's fractional cuts and rounding. Superadditive functions and valid inequalities. Mixed in-teger Gomory cuts. 9. Hybridization of methods for searching and separating solution spaces (Branch and Cut). Dynamic generation of constraints for exponential row problems. Application of separation oracles in practice. 10. Lagrange duality. Lagrange relaxation. Methods for solving the Lagrange dual problem. Heuristics in algorithms based on Lagrange relaxation. Choosing the Lagrange relaxation for the primal problem. 11. Column (row) generation methods. Dantzig-Wolfe formulations of integer programming problems. Solving a master linear programming problem via the column generation method. Solving a master problem via the branch and price method. The branch-cut-price method. 12. Benders' method. Benders' decomposition of an integer programming problem. The branch-cut method for the Benders formulation of an integer programming problem. 13. Application of hybrid methods to convex optimization problems in which the first-order method provides a starting point for the integer method. Summary of advanced mechanisms and architectures of modern solvers.
Laboratory
An introduction to modern optimization modeling environments and languages for integer optimization problems. An exercise in using the environment to solve linear programming problems. Using the environment to solve the problem of determining the shortest path in a graph by rigorously solving a linear programming problem. Using the environment to implement Gomory's method for the cutting stock problem. Applying integer optimization to building linear regression models with l_0 regularization, including integer problem formulations, first-order methods, and hybrid methods for building regression models. Applying integer optimization to solving complex Weapon Target Assignment (WTA) problems by implementing the branch-price-cut method for a convex integer optimization problem.

4. Learning outcomes			
Type of learning outcomes	Learning outcomes description	Reference to the learning outcomes of the WUT DS	Learning outcomes verification methods*
Knowledge			
W01	He has knowledge about integer optimization methods.	SD_W2	report evaluation
W02	He has knowledge about numerical methods for integer optimization problems.	SD_W3	report evaluation
W03			
Skills			
U01	He can solve integer optimization problem with the help of the appropriate numerical software.	SD_U1	report evaluation
U02	He can formulate and solve integer optimization problem with the help of optimization modelling language.	SD_U2, SD_U4	report evaluation
U03	He can formulate an environment script for solving integer optimization problems in order to implement an integer optimization algorithm.	SD_U2, SD_U4	report evaluation
U04	He can evaluate effectiveness analysis of applied optimization method.	SD_U2, SD_U4	report evaluation
Social competences			
K01	He is aware of the developments in the field of optimization and associated with that the necessity of continuing education.	SD_K2	
K02			
K03			

*Allowed learning outcomes verification methods: exam; oral exam; written test; oral test; project evaluation; report evaluation; presentation evaluation; active participation during classes; homework; tests

5. Assessment criteria
The final grade is based entirely on the evaluation of laboratory reports. Passing a single laboratory requires the submission of a report with a code and an oral defense verifying the adopted solution.

6. Literature
Primary references:
[1] L. A. Wolsey, Integer Programming (Second Edition), John Wiley & Sons, 2021.
[2] M. Grötschel, L. Lovász, A. Schrijver, Geometric Algorithms and Combinatorial Optimization (Second Edition), Springer-Verlag, 1993.

[3] C. H. Papadimitriou, K. Steiglitz, Combinatorial Optimization: Algorithms and Complexity, Dover Publications, 1998.

[4] G. Nemhauser, L. Wolsey, Integer and Combinatorial Optimization, Wiley, 1988.

[5] B. Korte, J. Vygen, Combinatorial Optimization. Theory and Algorithms (Fifth Edition), Springer-Verlag, 2012.

Secondary references:

[1] D. Bertsimas, R. Weismantel, Optimization over Integers, Dynamic Ideas, 2005.

[2] M. Conforti, G. Cornuéjols, G. Zambelli, Integer Programming, Springer-Verlag, 2014

7. PhD student's workload necessary to achieve the learning outcomes**		
No.	Description	Number of hours
1	Hours of scheduled instruction given by the academic teacher in the classroom	52
2	Hours of consultations with the academic teacher, exams, tests, etc.	5
3	Amount of time devoted to the preparation for classes, preparation of presentations, reports, projects, homework	37
4	Amount of time devoted to the preparation for exams, test, assessments	10
	Total number of hours	**104**
	ECTS credits	**4**

** 1 ECTS = 25-30 hours of the PhD students work (2 ECTS = 60 hours; 4 ECTS = 110 hours, etc.)

8. Additional information	
Number of ECTS credits for classes requiring direct participation of academic teachers	2
Number of ECTS credits earned by a student in a practical course	2