# ECE592-084/CSC591-084: Optimizations and Algorithms

**Instructor:** Dr. Shih-Chun Lin (919-515-5128; slin23@ncsu.edu)
**Classroom:** EB2-1230
**Credit Hours:** 3
**Class Time:** Mondays and Wednesdays, 4:30 P.M. – 5:45 P.M.
**Instructor Office:** 3072 Engr Bldg II
**Instructor Office Hours:** Mondays and Wednesdays, 6 P.M. – 7 P.M. at Zoom section or by email appointment with "[ECE592]" in the subject line.

**Course Description:** This course introduces advanced optimization theory and algorithms with rapidly growing applications in machine learning, systems, and control. Methods are given to obtain a non-dynamic system's extremum (minimum or maximum) and use these methods in various engineering applications. This course aims to prepare graduate students with a solid theoretical and mathematical foundation and applied techniques at the intersection of optimization, algorithms, and machine learning to conduct advanced research in related fields. Students will gain expertise in designing algorithms based on common techniques, dealing with intractable problems, and implementing algorithms given the description. Students must undertake a semester-long project (at Google Colab) that practices the optimization theory and algorithms in their areas of interest. These projects can replicate or improve a known solving strategy for a given optimization problem to assess and compare the performance.

**Student Learning Outcomes:** Upon completion of this course, students will:

- Solid Theoretical Foundation: Develop a profound theoretical understanding of optimization, algorithms, and machine learning, providing them with a solid academic basis for advanced research.
- Algorithmic Proficiency: Gain expertise in designing algorithms based on commonly used techniques, enabling them to address complex and intractable problems effectively.
- Implementation Skills: Acquire the practical skills to implement algorithms based on descriptions, fostering their ability to translate theoretical knowledge into tangible solutions.
- Research Experience: Undertake a comprehensive semester-long project using platforms like Google Colab, applying optimization theory and algorithms to areas of personal interest.
- Problem Solving and Evaluation: Engage in projects that involve replicating or enhancing existing solving strategies for optimization problems, allowing students to assess and compare their performance critically.

By focusing on these learning outcomes, this course prepares students to navigate the intricate landscape of optimization, algorithms, and machine learning and contribute significantly to advanced research in these vital domains.

## Course Structure

**Prerequisite:** Introductory courses in Probability and Linear Algebra.
**Textbook:** E.K.P. Chong and S.H. Zak, "An Introduction to Optimization," John Wiley & Sons, 2013.
The course website will provide a list of essential and trending papers. Useful reference books/papers on optimization theory and mathematical backgrounds include:

- S. Boyd and L. Vandenberghe, "Convex Optimization," Cambridge University Press, 2004.
- Y. Nesterov, "Introductory Lectures on Convex Optimization: A Basic Course," Springer, 2004.
- M. Bazarra, H.D. Sherali, and C.M. Shetty, "Nonlinear Programming: Theory and Algorithms," John Wiley & Sons, 2006.

- P. Domingos, "A Few Useful Things to Know about Machine Learning," Communications of the ACM, vol. 55, no. 10, pp. 78-87, October 2012.

**Course Materials:** Slides and in-class notes on Moodle.

**Topics:**

- Nonlinear unconstrained optimization, linear programming, nonlinear constrained optimization, computational and search methods for optimization, convex optimization, and integer programming.
- Greedy, divide and conquer, dynamic programming; approximation algorithms.
- Stochastic optimization, sparsity, regularized optimization, interior-point methods, proximal methods, robust optimization.
- Convergence rate analysis, momentum-based acceleration, distributed and asynchronous algorithm design, saddle point escaping.

**Course Project Information:**

- The course project will be a group project, and each group will consist of 3 students.

**Semester-long project:**

1. *Research:* Let students practice the optimization theory and algorithms in their areas of interest. It is allowed to be a replication or an improvement of a known solving strategy for a given optimization problem to assess/compare performance characteristics.
2. *Survey:* Choose a contemporary topic of optimization theory and algorithms or machine learning, read some recent papers related to your selected topic, and prepare a survey covering these papers. While some suggested topics will be provided by the instructor (available on a first-come, first-served basis), you can also propose your survey topic that may be of higher interest to your group.

- Two project deliverables are expected (no partial credit will be given for late project submissions):
  1. A 1-page extended abstract (25%): **Due September 16, 2024**
  2. A final report and slides (75%): **Due December 2, 2024**

**Extended Abstract and Final Report/Slides Format:**

- An extended abstract and final report should be written based on the IEEE conference paper template, which can be found in the link below. The final report should be **10 to 11 pages (double-column, single-spaced).** Also, **the corresponding presentation slides in PPT** to convey the know-how should be submitted following the class note format.
  - https://www.ieee.org/conferences/publishing/templates.html
    (use of LaTeX over MS Word is strongly encouraged for writing your reports)
  - Introduction, later sections, conclusion, references.
  - There should be multiple figures (e.g., 6-8 figures) and a table summarizing and classifying the key references (may have multiple columns studying different tradeoffs).
  - Some technical depth would be useful if relevant.
- Your 1-page extended abstract should include a summary of the proposed work and a plan (in table format) on how each group member will contribute to the project. Similarly, your final report should include an appendix section that briefly describes who worked on which part of the project.
- All used references should be explicitly cited within your report and slides. **Plagiarism (e.g., copying/pasting from Wikipedia or other papers without providing citations and changing wording) will be heavily penalized during grading. Everything in your reports should be in your own words.**

## Grade Determination

**Grade Distribution:**

40% Four homework assignments; each contributes 10%
35% Semester-long project: 25% abstract and 75% report/slides
25% Midterm exam

Weighted averages of 90, 80, and 70 will guarantee minimal letter grades of A-, B-, and C-, respectively.

## Course Schedule

**Course Outline**

*Introduction* (~4 Lectures)

- Motivating Examples
- Mathematical Preliminaries

*Unconstrained Optimization* (~8 Lectures)

- First- and Second-Order Conditions
- Algorithms for Unconstrained Optimization
  - One-Dimensional Search Methods
  - Gradient Methods
  - Newton Methods
  - Conjugate Direction Methods
  - Quasi-Newton Methods

*Least Squares Analysis* (~2 Lectures)

- Examples and Basic Properties
- Least Squares Algorithm
- Recursive Least Squares Algorithm

*Random Search Algorithms* (~2 Lectures)

- Simulated Annealing
- Particle Swarm Optimization
- Genetic Algorithms

*Linear Programming* (~3 Lectures)

- Examples and Basic Properties
- Simplex Method
- Duality

*Nonlinear Constrained Optimization* (~4 Lectures)

- Lagrange and Second Order Conditions
- Karush-Kuhn-Tucker and Second Order Conditions
- Algorithms for Constrained Optimization

*Convex Optimization* (~3 Lectures)

- Convexity
- Optimality Conditions
- Linear Matrix Inequalities (LMIs)

| | Date | Topic | Homework Assigned | Homework Due |
|---|---|---|---|---|
| **Week-1** | Aug 19 | Introduction | | |
| | Aug 21 | Introduction | | |
| **Week-2** | Aug 26 | Introduction | **HW1** | |
| | Aug 28 | Introduction | | |
| **Week-3** | Sept 2 | **Labor Day (No class)** | | |
| | Sept 4 | Unconstrained OPT | | |
| **Week-4** | Sept 9 | Unconstrained OPT | | |
| | Sept 11 | Unconstrained OPT | | **HW 1** |
| **Week-5** | Sept 16 | Unconstrained OPT | **1-Page Project Proposal Due (25%)** | |
| | Sept 18 | Unconstrained OPT | **HW2** | |
| **Week-6** | Sept 23 | Unconstrained OPT | | |
| | Sept 25 | Unconstrained OPT | | |
| **Week-7** | Sept 30 | Unconstrained OPT | | |
| | Oct 2 | Least Squares Analysis | | |
| **Week-8** | Oct 7 | Least Squares Analysis | | **HW2** |
| | Oct 9 | **Midterm Exam Practice** | | |
| **Week-9** | Oct 14 | **Fall Break (No class)** | | |
| | Oct 16 | **Midterm Exam at 4:30 P.M. – 5:45 P.M.** | | |
| **Week-10** | Oct 21 | Random Search Algo. | **HW3** | |
| | Oct 23 | Random Search Algo. | | |
| **Week-11** | Oct 28 | Linear Programming | | |
| | Oct 30 | Linear Programming | | |
| **Week-12** | Nov 4 | Linear Programming | | |
| | Nov 6 | Nonlinear OPT | **HW4** | **HW3** |
| **Week-13** | Nov 11 | Nonlinear OPT | | |
| | Nov 13 | Nonlinear OPT | | |
| **Week-14** | Nov 18 | Nonlinear OPT | | |
| | Nov 20 | Convex OPT | | |
| **Week-15** | Nov 25 | Convex OPT | | **HW4** |
| | Nov 27 | **Thanksgiving (No class)** | | |
| **Week-16** | Dec 2 | Convex OPT | **Final Project Report/Slides Due (75%)** | |
| **Week-17** | Dec 13 | **Fall grade due by 5 P.M. on 12/13, Friday** | | |

- The course schedule is subject to change with appropriate notification to students.
- **Audit students** need to do all the homework, take the midterm exam, and have at least a B- to get a passing grade. They do not need to work on the project.

**Late Assignments Policy**

1. There will be a total of 4 homework throughout the semester.
2. Late homework assignments will receive a 50% penalty for being up to one day late and no credit if submitted later than one day after the deadline. If you miss the deadline for the homework by less than one day, email the homework to me and the TA and take the 50% penalty.
3. Every homework assignment will have an equal weight.
4. If you have any questions regarding the homework, please see the TA first and seek answers on the Piazza (the course's online discussion forum). If you still need clarification after contacting the TA, please get in touch with me.

**Academic Integrity:** Students should refer to the University policy on academic integrity found at https://studentconduct.dasa.ncsu.edu/academic-integrity-overview/ and the Code of Student Conduct (NCSU POL11.35.01) and Pack Pledge
Authorized aid on an individual assignment includes discussing the interpretation of the problem statement, sharing ideas or approaches for solving the problem, and explaining concepts involved in the problem. Any other aid would be unauthorized and violate the academic integrity policy. All cases of academic misconduct will be submitted to the Office of Student Conduct. Violations of academic integrity will be handled in accordance with the Student Discipline Procedures (NCSU REG 11.35.02).

**Statement for Students with Disabilities:** Reasonable accommodations will be made for students with verifiable disabilities. In order to take advantage of available accommodations, students must register with the Disability Resource Office at Holmes Hall, Suite 304, 2751 Cates Avenue, Campus Box 7509, 919-515-7653. For more information on NC State's policy on working with students with disabilities, please see the Academic Accommodations for Students with Disabilities Regulation (NCSU REG 02.20.01).

**Digital Course Components:** The following digitally-hosted course components will be used: Moodle, Zoom, Panopto, Slido, and Piazza. Students may be required to disclose personally identifiable information to other students in the course, via, digital tools, such as email or web-postings, where relevant to the course. Examples include online discussions of class topics, and posting of student coursework. All students are expected to respect the privacy of each other by not sharing or using such information outside the course.

**Insurance of Course Continuity:** to ensure course continuity, changes made to the method of instructional delivery, course structure, course schedule, number of assignments, grading or other aspects of the course after the start of the term should be communicated to all students in written form (e.g., dated syllabus revision or syllabus addendum) when course changes are implemented.

**Additional NC State Rules and Regulations:** Students are responsible for reviewing the NC State University Policies, Rules, and Regulations (PRRs) which pertain to their course rights and responsibilities, including those referenced both below and above in this syllabus:

- Equal Opportunity and Non-Discrimination Policy Statement https://policies.ncsu.edu/policy/pol-04-25-05 with additional references at https://oied.ncsu.edu/divweb/policies/
- Code of Student Conduct https://policies.ncsu.edu/policy/pol-11-35-01