ORIE 7390: Mathematical Techniques for Optimization
D. Shmoys, S. Banerjee, M. Udell, S. Gutekunst
Organizational Meeting: Friday 8/25/17 from 2:30-3:00pm in Rhodes 253
Regular Meeting Time: Thursdays from 5:45pm-7:00pm in Rhodes 261, except on 9/12, when we have Rhodes 453

Scope
This reading course will introduce technical tools that are not generally covered in the ORIE curriculum, but that have been applied as powerful hammers to optimization problems. To do so, the course will lean on a mix of survey papers, textbook chapters, and research papers. The goal is for the class to develop an awareness of and a working familiarity with several tools, their basic mathematical formulation and how they can be applied. Tentative topics include: hierarchies in optimization, online convex optimization, Grobner bases and integer programming, and algorithms and approximation theory.

Logistics
At the beginning of the semester, we'll have a short organizational meeting to budget time to the potential topics and to fix a weekly meeting time. During the semester, we'll meet for 75 minutes each week with 1-2 students presenting on a topic. These presentations will be based on suggested sources, though presenters may supplement it with other resources or papers they find. Before each presentation, the expectation is for all students to have read survey material indicated in the course calendar (though not necessarily to have grappled with every technical detail). The target of the presentations is then to highlight the motivation and key ideas of the paper, and to explain the technical details important for applications. Finally, the course will move dynamically and flexibly; if a topic seems especially compelling, we're happy to budget more time on it!

Prerequisites
Students should be comfortable with basic proof based mathematics, linear algebra, and undergraduate optimization. Having taken or concurrently taking ORIE 6300 will provide helpful context.

Grading
This is a 1 credit S/U course. Passing the course involves the following:

- Delivering approximately 1-2 presentations (this is estimated, but depends on enrollment and on whether people prefer to pair up for presentations or give individual presentations)
- Scribing 1-2 presentations (up to enrollment)
- Missing at most 2 of the meetings (up to special circumstances) and participating during the presentations

Questions
Email Sam Gutekunst (netid SCG94)

Course Topics and Reference Material

Hierarchies in Optimization: Consider a combinatorial optimization problem with an integer program expressed in terms of n binary variables. Ideally, we'd be able to optimize over the *convex relaxation* of the integer program: the set of all convex combinations of feasible integer points. Instead, we often start with the *linear program* relaxation: the optimization problem obtained by dropping integrality constraints. Hierarchies present a systematic way to add variables and constraints (often semidefinite) such that, if you add enough of them, you exactly get the convex relaxation. While ``enough'' is generally too large, these systematic procedures have several uses. For example, applying a small number of iterations can sometimes lead to a better approximation relaxation that does not require ``too many'' new variables/constraints. They also can be used to derive certain semidefinite relaxations and to provide heuristic evidence against algorithms.

This topic involves a bit of linear algebra as well as comfort with semidefinite matrices and basic integer programming. Our goal is to cover the basics of hierarchies in general, and then to focus on applications and theory related to the strongest hierarchy: the Lasserre Hierarchy. Applications include scheduling problems, graph matching, set cover, and possibly the minimum bisection problem. Our reference material includes:

- For a general overview of hierarchies, see Convex Relaxations and Integrality Gaps by Chlamtac and Tulsiani. Application papers include Lovasz-Schrijver Reformulation by Tulsiani and On A Representation of the Matching Polytope Via Semidefinite Liftings by Stephen and Tuncel. Another cute application is in Arora's lecture notes where the famous Goemans-Williamson relaxation of Max-Cut is derived using heirarchies.
- An overview of the Lasserre Hierarchy is The Lasserre Hierarchy in Approximation Algorithms by Rothvoss. See also these slides on the same topic by Rothvoss
- We may cover an application to the minimum bisection problem, found in Lasserre Hierarchy, Higher Eigenvalues, and Approximation Schemes for Quadratic Integer Programming with PSD Objectives by Guruswami and Sinop

Online Convex Optimization

Online convex optimization describes a structured learning setting: we make sequential decisions and incur loss according to structured (convex) functions. An introductory problem (that we'll start with) is that of the hidden expert: in each of T rounds, you make a yes/no decision. You receive advice from N people, one of which is an ``expert'' (who, in general, gives the best advice). You do not know who the expert is, and her advice is not necessarily perfect, but want a decision making process that, on average, doesn't make too many more mistakes than the expert.

This topic involves some comfort with basic convex optimization and multivariable calculus (you should not be afraid of gradients), and comfort with expectation. Our goal is to cover the introductory material of a NOW monograph, including: several algorithms for the hidden expert problem, classic convex optimization, and online gradient descent. Applications include support vector machines. If there is sufficient interest, we can also discuss applications to semidefinite programming and bandit problems. Our reference material includes:

- Our main reference is Introduction to Online Convex Optimization by Hazan. We'll likely cover material from chapters 1-3, paying attention to the following applications: multiplicative weights, support vector machines, and stochastic gradient descent
- We may also cover parts of Fast Algorithms for Approximate Semidefinite Programming using the Multiplicative Weights Update Method by Hazan
- A supplemental reference is Online Learning and Online Convex Optimization by Shai Shalev-Shwartz

Grobner Bases and Integer Programming
A classic problem in mathematics is to find feasible solutions to a set of linear equations. An equivalent way to view this problem is as finding points where a set of multivariate (degree 1) polynomials vanish simultaneously. For example, solving the system

$$2x + y = 7,\ 3x - 5y = 12$$

is equivalent to finding points (x, y) where

$$P_1(x,\ y) = 2x + y - 7,\ \ P_2(x,\ y) = 3x - 5y - 12$$

vanish simultaneously (i.e., are both zero). Grobner bases are a general tool in algebraic geometry for solving these types of systems, without the requirement that polynomials be degree 1. The application we'll focus on involves using them to solve integer programs: we can encode integer programs as systems of multivariate polynomials and use theory from algebraic geometry to study those polynomials.

This material involves comfort with multivariate functions. Minimal background in abstract algebra is helpful because of the vocabulary (field, ring of polynomials, etc), but not necessary. Our goal will be to understand the theory that lets us translate integer programs to polynomial systems and the role of Grobner bases. Our reference material is based off of:

- Our main reference will be chapter 10 and (some of) chapter 11 of Algebraic and Geometric Ideas in the Theory of Discrete Optimization. This is somewhat demanding, and will be supplemented with What a Gröbner Basis? by Sturmfels, Pachters's lecture notes introducing Grobner basis, and Integer Programming with Grobner Bases by Flory and Michel. Sturmfels has relaxed, introductory lectures recorded here.
- *Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra* by Cox, Little, and Shea is a Springer UTM book that expands on some of the math, and is a supplemental reference

Algorithms and Approximation Theory
Approximation theory refers to the way we approximate functions by simpler functions. For example, we might approximate x^n in some small interval $[a, b]$ by a lower degree polynomial; Chebyshev polynomials provide one method. These approximations can be used to speed up algorithms. For example, Chebyshev polynomials can be used to approximately compute a matrix power A^n more efficiently, which in turn can be used to study properties of a random walk on a graph.

This material involves comfort with running time and numerical linear algebra (e.g., at the level of Alex Townsend's Top 10 Algorithms class). Our goal will be to understand a few basic function approximations, how they generalize to matrices, and how they can be used in simulating random walks and for numerically solving systems of linear equations.

Our main reference is Faster Algorithms via Approximation Theory by Sachadeva.