Research
Australian Mathematical Sciences Institute

ABSTRACTS: AMSI-ANZIAM LECTURER 2013 – PROFESSOR STEPHEN P. BOYD

CVXGEN: A Code Generator for Embedded Convex Optimization

Joint work with J. Mattingley

CVXGEN is a software tool that takes a high level description of a convex optimization problem family, and automatically generates custom C code that compiles into a reliable, high speed solver for the problem family. The current implementation targets problem families that can be transformed, using disciplined convex programming techniques, to convex quadratic programs of modest size. CVXGEN generates simple, flat, library-free code suitable for embedding in real-time applications. The generated code is almost branch free, and so has highly predictable run-time behavior. The combination of regularization (both static and dynamic) and iterative refinement in the search direction computation yields reliable performance, even with poor quality data. In this paper we describe how CVXGEN is implemented, and give some results on the speed and reliability of the automatically generated solvers.

Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers

Joint work with N. Parikh, E. Chu, B. Peleato, and J. Eckstein

Many problems of recent interest in statistics and machine learning can be posed in the framework of convex optimization. Due to the explosion in size and complexity of modern datasets, it is increasingly important to be able to solve problems with a very large number of features, training examples, or both. As a result, both the decentralized collection or storage of these datasets as well as accompanying distributed solution methods are either necessary or at least highly desirable. We argue that the alternating direction method of multipliers is well suited to distributed convex optimization, and in particular to large-scale problems arising in statistics, machine learning, and related areas. The method was developed in the 1970s, with roots in the 1950s, and is equivalent or closely related to many other algorithms, such as dual decomposition, the method of multipliers, Douglas-Rachford splitting, Spingarn's method of partial inverses, Dykstra's alternating projections, Bregman iterative algorithms for problems, proximal methods, and others. After briefly surveying the theory and history of the algorithm, we discuss applications to a wide variety of statistical and machine learning problems of recent interest, including the lasso, sparse logistic regression, basis pursuit, covariance selection, and support vector machines.

Dynamic Network Energy Management via Proximal Message Passing

Joint work with M. Kraning, E. Chu, and J. Lavaei

We consider a network of devices, such as generators, fixed loads, deferrable loads, and storage devices, each with its own dynamic constraints and objective, connected by AC and DC lines. The problem is to minimize the total network objective subject to the device and line constraints, over a given time horizon. This is a large optimization problem, with variables for consumption or generation for each device, power flow for each line, and voltage phase angles at AC buses, in each time period. In this paper we develop a decentralized method for solving this problem called proximal message passing. The method is iterative: At each step, each device exchanges simple messages with its neighbors in the network and then solves its own optimization problem, minimizing its own objective function, augmented by a term determined by the messages it has received. We show that this message passing method converges to a solution when the device objective and constraints are convex. The method is completely decentralized, and needs no global coordination other than synchronizing iterations; the problems to be solved by each device can typically be solved extremely efficiently and in parallel. The method is fast enough that even a serial implementation can solve substantial problems in reasonable time frames. We report results for several numerical experiments, demonstrating the method's speed and scaling, including the solution of a problem instance with over 30 million variables in 5 minutes for a serial implementation; with decentralized computing, the solve time would be less than one second.

Performance Bounds and Suboptimal Policies for Multi-Period Investment
Joint work with M. Mueller, B. O'Donoghue, and Y. Wang

We consider dynamic trading of a portfolio of assets in discrete periods over a finite time horizon, with arbitrary time-varying distribution of asset returns. The goal is to maximize the total expected revenue from the portfolio, while respecting constraints on the portfolio such as a required terminal portfolio and leverage and risk limits. The revenue takes into account the gross cash generated in trades, transaction costs, and costs associated with the positions, such as fees for holding short positions. Our model has the form of a stochastic control problem with linear dynamics and convex cost function and constraints. While this problem can be tractably solved in several special cases, such as when all costs are convex quadratic, or when there are no transaction costs, our focus is on the more general case, with nonquadratic cost terms and transaction costs.
We show how to use linear matrix inequality techniques and semidefinite programming to produce a quadratic bound on the value function, which in turn gives a bound on the optimal performance. This performance bound can be used to judge the performance obtained by any suboptimal policy. As a by-product of the performance bound computation, we obtain an approximate dynamic programming policy that requires the solution of a convex optimization problem, often a quadratic program, to determine the trades to carry out in each step. While we have no theoretical guarantee that the performance of our suboptimal policy is always near the performance bound (which would imply that it is nearly optimal) we observe that in numerical examples the two values are typically close.

The Role of Embedded Optimization in Smart Systems and Products
Stephen Boyd

Many current products and systems employ sophisticated mathematical algorithms to automatically make complex decisions, or take action, in real-time. Examples include recommendation engines, search engines, spam filters, on-line advertising systems, fraud detection systems, automated trading engines, revenue management systems, supply chain systems, automatic circuit synthesis and layout tools, electricity generator scheduling, flight management systems, and advanced engine controls. I'll cover the basic ideas behind these and other applications, emphasizing the central role of mathematical optimization and the associated areas of machine learning and automatic control. The talk will be nontechnical, but the focus will be on understanding the central issues that come up across many applications, such as the development or learning of mathematical models, the role of uncertainty, the idea of feedback or recourse, and computational complexity.