1 Class Projects

The goal of the final project is to develop a deep understanding of an important research area and, to the extent possible, work on an open problem. You may also review an existing area of the literature, providing a careful and critical comparison of different approaches to computational mechanism design. [It is harder to write an exceptionally good literature survey than to do an exceptionally good original piece of work, though!]

You are strongly encouraged to build from the discussions in class, and to also remember that there are a few papers still to be covered. Projects may be theoretical or experimental; theoretical projects must be individual, while experimental projects can be completed in pairs. A list of suggested topics for projects are provided here, but you are encouraged to propose a topic of your own for approval. Additional ideas can be found in student comments on papers and also will be posted as we continue to read papers and leading up to April 20.

Project proposals are due by noon, Friday, April 20, to Arthur Cram. We will also be discussing projects in class on Monday, April 16th.

Your proposal should be no more than a couple of pages in length. Your goal should be to convince me that your ideas are well enough focused, and that you know what the first few steps will be. You must explicitly include the following sections in your proposal:

- Provide a high-level description of your project.
- Provide motivation for your proposed work (what will we know afterwards that we don’t know now, and why do we care?)
- Place your work in the context of the papers that we have read in the course.
- Let me know what things that Jacomo and I can help with to get you started.
- Anticipate what you think you will find in your results.
2 Current Project Suggestions

Many of these projects could be either theoretical or experimental. In addition, in some of these problems there might be a place where you need computational solutions to combinatorial optimization problems (such as winner-determination problems). Expect to be able to use off-the-shelf integer programming software (e.g. CPLEX) for this purpose, that you can call from either Java or C++. Also, please let me know if you need an eecs account to perform your work. We also have access to a general statistical inference package (Hugin) and also software for decision trees (C 5.0) should you want to run experiments with real economic data (e.g. scraped from the Internet; we also have scripts to grab eBay data for example.)

2.1 Theory and Practice of Mechanism Design

Optimal Auctions. Study the design of profit-maximizing auctions assuming independent prior distributions on the valuations of the bidders. See ‘Optimal Auction Design’ paper by Myerson. [E.g., gain an understanding of virtual valuations, ironing, etc.]

Truthful Restricted CAs. Study programming language techniques in order to generate automatic proofs that functions are monotonic, and that the composition of functions are monotonic.

Competitive Auctions. Complete an analytical investigation of the performance of random-sampling optimal auctions (RSOP).

Competitive Auctions. How does a limited supply assumption modify the revenue properties of the RSOP auction?

Competitive Auctions. Recall that digital goods auctions have no costs of productions. Adding costs seem to significantly complicate the problem.

Revenue-maximizing Combinatorial Auctions. Explore alternate gradient ascent methods, alternate parameterizations for the design of heuristically-optimal, revenue-maximizing CAs (i.e. searching in the space of linear-affine VCG mechanism.)

Take-it-or-leave It offers: Extend these methods to work in an adaptive setting, where it is necessary to learn the distribution of agent valuations over time.

Incremental Mechanism Design. In the IJCAI-07 paper by Sandholm and Conitzer: analyze the relevance of the specific method used for step 3 in the algorithm framework regarding the prevention of loops. Is using method a) most likely to avoid looping?

Incremental Mechanism Design. In the IJCAI-07 paper by Sandholm and Conitzer: loops seem to be a severe problem of this approach. Which techniques can we used to prevent loops in general (e.g. tabu-search?)?

Incremental Mechanism Design. In the IJCAI-07 paper by Sandholm and Conitzer: what mechanisms can be designed without loops? For a given neighborhood-size, can we formally derive constraints on the type space so that
when the incremental mechanism design approach is applied to such a given problem we are guaranteed not to have loops?

**Incremental Mechanism Design.** In the IJCAI-07 paper by Sandholm and Conitzer: can we use machine learning techniques (e.g. SVMs) to find the new (improved, i.e. more strategyproof) mechanism?

**Incremental Mechanism Design.** In the IJCAI-07 paper by Sandholm and Conitzer: how do we get from a mechanism for $n$ agents to one for $n + 1$ agents?

### 2.2 Sponsored Search and Bidding Agent Design

**Keyword Auctions.** In the AER ’06 paper by Edelman et al.: An analysis of the relevance of the ratio between the $\alpha_i$ values. What is the impact on manipulability and truthfulness? Will the probability that the auction is strategyproof converge to 1 for certain $\alpha_i$ values?

**Keyword Auctions.** Study the various equilibria of the Google and Yahoo! ad auctions. Propose alternate refinements (e.g., with similar motivations to the "envy-free" solution.)

**Keyword Auctions.** Examine empirical evidence of mispricing in Keyword Auctions.

**Squashing.** Design a scheme that adjusts the squashing parameter while learning the underlying relevance of different adverts. [E.g., a tradeoff between exploration and exploitation.]

**Equilibrium Analysis.** As more bidding is done by simple agents like Atlas’ BidManager, we can imagine the equilibrium analysis to change as most of the population is restricted to heuristically simple strategies. How does that affect auction design or the properties of current sponsored search auctions?

**Greedy Bidding Strategies.** Study the dynamics of agents bidding in repeated VCG mechanisms.

**Greedy Bidding Strategies.** Study the convergence of alternate dynamic bidding strategies in the context of an auction protocol for sponsored search.

### 2.3 Dynamic Mechanisms

**Online Auctions.** Extend one of the models of online auctions to make it more realistic, e.g. to handle a grid-scheduling problem.

**Keyword Auctions.** Design a dynamic mechanism for a problem motivated by sponsored search auctions. See *Online Mechanisms* by D.C. Parkes. (We read this on 4/11).