Risk Aware Agent for Bidding in Second-Price Sealed-Bid Auctions and variants

This technical report introduces two forms of a risk aware agent for bidding in Simultaneous Second-Price Sealed-Bid Auctions but who's heuristics are general enough to be used in other auction settings. The main idea of the risk aware agent is the computation of a utility score that given predictions of prices over bundles of goods, considers the risk associated with the distribution of prices over each good contained in a given bundle. This utility is a generalization of the classic acquisition problem described in (Boyan & Greenwald, 2001) and used by every agent in (Yoon & Wellman, 2011) .

As a baseline, all agents from (Yoon & Wellman, 2011) were implemented to test the efficacy of the proposed risk-aware agent as well as parallel implementations of both point price predicting and distribution predicting algorithms in order to maximize the number of auction simulations executed in minimal time. The implementation language is Python and all code and documentation can be found in the github repository:

git://github.com/bam593/bmProjects.git

The contents can be viewed with a web browser at the address:

https://github.com/bam593/bmProjects/tree/master/courses/fall2011/csci2951

Risk Aware Agent

Every strategy profile described in (Yoon & Wellman, 2011) assumes the bidding process can be decomposed into two independent modules:

1) Identify a set of goods that if won would maximize the agent's surplus, defined as valuation less cost (this step is known as the acquisition problem).
2) Compute bids to place on the goods in the bundle that solves that the acquisition problem.

That is we first define a bundle as a collection of goods that we can potentially procure at auction.

If there are m goods available, we define the set of all possible bundles, X, available for purchase as

$$X = \{x : x \in \mathbb{P}^m\}$$

where $\mathbb{P}^m$ represents the power set of m items. For example, lets' assume $m = 4$, we can enumerate every possible bundle using bit vectors of length 4, representing the set of all bundles as:

$$X = \{[0,0,0,0], [0,0,0,1], \dots, [1,1,1,0], [1,1,1,1]\ \}$$

If the j^{th} entry is 1 in the i^{th} bundle bit vector, this indicates the j^{th} good is purchased in bundle i, otherwise the j^{th} good is not included in the bundle.

The agents' valuation function is then a function that assigns a scalar to each bundle

$$v : \mathbb{P}^m \rightarrow \mathbb{R}.$$

This function is known to the agent prior to bidding. The function used in this report are described in (Sodomka & Greenwald).

Given a vector of prices representing the price paid for each good in a bundle, $p \in \mathbb{R}^m$ we can calculate the cost , c, associated with each bundle as the dot product of the i^{th} bundle and the price vector p and thus compute the surplus σ for each bundle.

$$c = x_i \cdot p$$

$$\begin{aligned} \sigma_i(X, p) &= v - c \\ &= v - x_i \cdot p \end{aligned}$$

Therefore the acquisition problem of step (1) is a matter of solving (Boyan & Greenwald, 2001):

$$X^* = ACQ_i(p) = \arg\max_{X \subseteq \mathcal{X}} \sigma_i(X, p)$$