# A Decision-Theoretic Algorithm for Bundle Purchasing in Multiple Open Ascending-Price Auction

Buffett, Scott; Grant, A.



**Publisher's version / Version de l'éditeur:**

*The 17th Conference of the Canadian Society for Computational Studies of Intelligence (AI 2004) [Proceedings], 2004*

National Research Council Canada
Institute for Information Technology

Conseil national de recherches Canada
Institut de technologie de l'information

# A Decision-Theoretic Algorithm for Bundle Purchasing in Multiple Open Ascending-Price Auction *

Buffett, S., and Grant, A.
May 2004

* published in the 17th Conference of the Canadian Society for Computational Studies of Intelligence (AI 2004). London, Ontario, Canada. pp: 429-433. May 17-19, 2004. NRC 47144.

# A Decision-Theoretic Algorithm for Bundle Purchasing in Multiple Open Ascending-Price Auctions

Scott Buffett and Alastair Grant

Institute for Information Technology – e-Business
National Research Council Canada
46 Dineen Drive, Fredericton, New Brunswick, Canada E3B 9W4
{Scott.Buffett, Alastair.Grant}@nrc.gc.ca

**Abstract.** This paper presents an algorithm for decision-making where the buyer needs to procure one of possibly many bundles of complementary products, and items are sold individually in multiple open ascending-price (English) auctions. Auctions have fixed start and end times, and some may run concurrently. Expected utility of a bidding choice is computed by considering expected utility of choices at future decisions. The problem is modeled as a Markov decision process, and dynamic programming is used to determine the value of bidding/not bidding in each state. Three techniques for reducing the state space are given. Results show that a bidding agent using this algorithm achieves significantly higher utility on average when compared with one that does not consider the value of future choices.

## 1 Introduction

As the volume of e-commerce completed through online auctions rises, so does the need for efficient and effective decision-making systems that can analyze several options and help a potential buyer make rational bidding decisions. Online auction sites have grown considerably over the past few years by mainly targeting the consumer, but recent research has shown that more and more businesses are integrating online auctions into their supply chains [6]. While research in e-commerce focuses on the problems associated with finding and monitoring auctions that may be of interest to a potential buyer (or bidding agent), artificial intelligence techniques are needed to analyze the data and help the agent make effective decisions on which auctions to pursue as well as how much to bid.

In this paper, we present a decision-making algorithm for an agent that needs to purchase one of possibly many *bundles* of products, where items are sold individually in auctions. In our context, we consider a bundle to be a user-defined set of complementary products. There also may be alternatives for certain products, and consequently several acceptable bundles. We consider open ascending-price (English) auctions where the start time, finish time and opening bid are fixed and known in advance. There is no restriction on auction times (i.e. time periods

A.Y. Tawfik and S.D. Goodwin (Eds.): Canadian AI 2004, LNAI 3060, pp. 429–433, 2004.

for some auctions may overlap). The goal is to analyze the incomplete information on current and future auctions and make bidding decisions that give the agent the best chance of ultimately procuring the best bundle in terms of bundle preference and cost.

Previous work [2,5] has analyzed the problem of determining the expected utility over sets of auctions, but this work bases the decision on whether or not to participate in an auction on whether or not the auction is part of the set deemed the best in terms of expected utility at that time. This "greedy" approach works best when a buyer immediately must commit to a particular set. In our setting, we compute the expected utility of a choice by considering the expected utility of future consequential decision points. This is the main contribution of the paper. To accomplish this, we use a purchase procedure tree [1] to model the system of decisions and purchases, and model the decision problem as a Markov decision process. Dynamic programming is then used to determine the optimal choice at each decision. To reduce the consequentially unmanageable size of the resulting state space, we employ three state-reducing techniques. Results show that our method performs significantly better than the greedy approach in a specific example.

## 2 Problem Formalization

Let $\mathcal{A}$ be a set of English auctions where each $a \in \mathcal{A}$ is an auction for product $p_a$, and all auctions in $\mathcal{A}$ are currently open or will open in the future (i.e. none are finished). Let $P = \{p_a \mid a \in \mathcal{A}\}$ be the set of products to be auctioned in $\mathcal{A}$, and let $\mathcal{B} \subseteq 2^P$ be a set of bundles. Each bundle is specified by the buyer as being a satisfactory and complete set of products. To specify the buyer's preferences assume that a bundle purchase utility function $u : \mathcal{B} \times C \rightarrow \Re$ is given, where $C$ denotes the set of possible bundle costs. The problem is to decide whether or not to bid in each auction, with the goal of ultimately obtaining all products in some bundle $b \in \mathcal{B}$ at cost $c$ with the highest utility possible.

## 3 The Purchase Procedure Tree

In order to structure the decision process, we use a purchase procedure tree (PPT) [1]. This tree graphically depicts the process of decisions and auctions that, when executed, will result in a complete bundle purchase. Starting at the root node, the buyer proceeds toward the leaf nodes, bidding in auctions at auction nodes and making decisions at decision nodes. At each decision node (lower case $d$'s) there are two choices: participating in the auction that will end next (always represented by the left child of the decision node), or allowing it to pass. When the auction ends, if the buyer is the winner then execution moves to the left, else to the right. Once a path is completed, the buyer will have procured a complete bundle. The example PPT in Figure 1 represents the problem where there are bundles AB, AC, BD and EF, and the auction for A ends first. The current decision ($d_1$) to be made is on whether or not to bid on A. The PPT shows the consequential decisions and auctions which result from each choice. Note that a new PPT is built for each decision. For example, if A is purchased then a new PPT will be built with root $d_3$. This tree may include new options, and perhaps may even include bundles that do not include the purchased items (i.e. BD or EF) if they are deemed viable options.

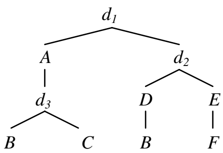


**Fig. 1.** An example purchase procedure tree

## 4 The MDP Model

To determine the expected utilities of each option, the sequence of auctions and decisions is modeled as a Markov decision process (MDP), and the optimal policy in the MDP is determined using the value iteration method of dynamic programming. Each state in the MDP is a 5-tuple $\langle P, c, \mathcal{A}_{cur}, C_{\mathcal{A}_{cur}}, t\rangle$ where $P$ is the set of purchased products, $c$ is the total amount spent on purchases, $\mathcal{A}_{cur} = (a_1, \ldots, a_m)$ contains the auctions that are currently running, $C_{\mathcal{A}_{cur}} = (c_1, \ldots, c_m)$ contains the current bid $c_i$ for each $a_i$, and $t$ is the time. The set of actions is $Q = \{bid, notbid\}$. Each terminal state has an associated reward, equal to the utility $u(b, c)$ of purchasing the bundle $b = P$ at cost $c$. The value $v(s)$ for a state $s$ is computed as the expected utility of $s$. The problem is to determine $v(s)$ for each reachable state $s$ in order to find the optimal policy $\pi : S \to Q$. For a state $s'$ at which a bidding decision must be made, $\pi(s')$ advises which course of action maximizes expected utility. Because of the stochastic nature of auctions, for many actions it is not known for certain what the next state will be. However, we assume that the buyer will have some idea of what the outcomes of an auction will be (by examining auction history, market history, estimation of competitors reserve values, etc.). We model this information in the form of a *prediction function* $F_a(c, t, t')$. For an auction $a$, $F_a(c, t, t')$ takes a bid $c$ and times $t$ and $t'$ (where $t < t'$), and returns a probability distribution $p$ on the outcomes for the current bid at time $t'$ given that the current bid at time $t$ is $c$.

The problem with modeling the decision problem in this way is that there will be far too many states in the MDP. At any given time, there may be several auctions open, each of which with several possible outcomes for the current bid. Also, there may be several different combinations of items already purchased by the buyer, and several possible costs for those purchased goods. An important contribution of this paper lies in how we deal with this computational complexity without losing too much of the important information. In particular we do three things to reduce the state space:

1. *Assume that bidding is only done at the end of the auction.* With this assumption, only states that occur when an auction ends are considered in the MDP. At these points the agent chooses either to bid (thus winning the auction), or pass. Realistically, with this strategy the bidder runs the risk of not having a bid accepted before the deadline if there are one or more bidders with the same strategy. But this is not necessarily the true strategy to be used by the buyer. It is only the assumption made about future actions in the MDP model to ease the computational burden. The buyer is free to bid earlier in an auction if the expected utility of doing so is higher than the expected utility of waiting until the end. As a result, since the utility of winning an auction with an earlier bid is always at least as good as winning it with a later bid (since bids never go down), the buyer's true expected utility is at least as high as that predicted by the algorithm (given that the prediction functions are sufficiently accurate).
2. *Use the purchase procedure tree.* The PPT shows the sequence of decisions and auctions that follow from any choice. We use this to limit the state space in two ways. First, two auctions $a_1$ and $a_2$ are considered together in $A_{cur}$ in a state only if there is a common ancestor decision node $d$ (of the nodes representing $a_1$ and $a_2$) in the PPT such that $a_1$ and $a_2$ will be open when the decision at $d$ must be made. Second, for any state $s$ at node $n$ in the PPT, the set $P$ in $s$ is always equal the union of the set of ancestor purchases in the PPT and the set of purchases already made before the PPT was built.
3. *Use the Pearson-Tukey three-point discrete approximation.* The probability measure $p$ given by $F_a(c,t,t')$ assigns positive probability to only three values, according to the Pearson-Tukey three-point approximation [3,4]. Specifically, $p(x_1) = .185$, $p(x_2) = .63$ and $p(x_3) = .185$ where $Prob(X > x_1) = .95$, $Prob(X > x_2) = .5$ and $Prob(X > x_3) = .05$.

The transition probability function $Pr(s'|s,q)$ takes states $s$ and $s'$ and an action $q$ and returns the probability of occupying $s'$ directly after $q$ is performed in $s$. Dynamic programming is then used to find the value of each state:

$$v(s) = \begin{cases} u(P,c) & \text{if } P \in \mathcal{B} \\ \max\{\sum_{s' \in S} v(s')P(s'|s,bid),\ \sum_{s' \in S} v(s')P(s'|s,notbid)\} & \text{otherwise} \end{cases} \quad (1)$$

and $\pi(s) = bid$ if $\sum_{s' \in S} v(s')P(s'|s,bid) > \sum_{s' \in S} v(s')P(s'|s,notbid)$, and $\pi(s) = notbid$ otherwise.

## 5 Results

The utility achieved using our method was compared with the utility achieved using the greedy method that instructs the buyer to bid on an item if it is part of the bundle with the highest expected utility. Tests were run using the product set $P = \{A, B, C, D, E, F\}$ with auction times $A : [0, 10]$, $B : [15, 45]$, $C : [15, 35]$, $D : [20, 40]$, $E : [25, 55]$ and $F : [38, 60]$, and the bundle set $\mathcal{B} = \{AB, CD, EF\}$. Each agent had the same preferences for bundles and were risk-neutral. 2000 tests were run for each bidding method, of which 1298 instances saw the two agents purchase a different bundle. Only these cases are examined. In each test run, the bidding agent being tested competed against 8 dummy bidders in each auction. Dummy agents' reserve values were chosen at random from reasonable distributions. Before the tests were run, the bidding agents were allowed to observe the dummy agents in order to learn the prediction function for each auction. Results showed that the mean utility achieved by our agent was 0.49079 (95% confidence interval $[0.48902, 0.49257]$), compared with 0.48607 ($[0.48468, 0.48747]$) by the greedy agent. The mean of the paired differences was 0.00472 ($[0.00242, 0.00702]$). A paired t-test indicates that the difference in the means is significant at $p < 0.0001$. While this is certainly not an exhaustive test, it shows promise that, in at least one scenario, our method significantly outperforms simple greedy decision-making.

## 6 Conclusions and Future Work

This paper presents an effective algorithm for making bidding decisions in multiple English auctions where the buyer needs to procure one of possibly several bundles of products. Expected utilities of choices are estimated more accurately by considering the value of future decisions. This results in better decision-making and higher utility. For future work, we plan to carry out more extensive experiments to further support our claims. One idea is to test our method on sets of actual online auctions, such as those found on *eBay*. This will involve monitoring auctions for a period of time in order to determine a prediction function for similar future auctions, and then simulating these real auctions with our bidding agent. This will give an idea not only of how well our technique performs in real auctions, but also how accurately these prediction functions can be determined.

## References

1. S. Buffett and B. Spencer. Efficient monte carlo decision tree solution in dynamic purchasing environments. In *Proc. International Conference on Electronic Commerce (ICEC2003)*, pages 31–39, Pittsburgh, PA, USA, 2003.
2. A. Byde, C. Preist, and N. R. Jennings. Decision procedures for multiple auctions. In *Proc. 1st Int Joint Conf. on Autonomous Agents and Multi-Agent Systems (AAMAS'02)*, pages 613–620, Bologna, Italy, 2002.
3. D. L. Keefer and S. E. Bodily. Three point approximations for continuous random variables. *Management Science*, 29(5):595–609, 1983.
4. E. S. Pearson and J. W. Tukey. Approximating means and standard deviations based on distances between percentage points of frequency curves. *Biometrika*, 52(3-4):533–546, 1965.
5. C. Preist, C. Bartolini, and A. Byde. Agent-based service composition through simultaneous negotiation in forward and reverse auctions. In *Proceedings of the 4th ACM Conference on Electronic Commerce*, pages 55–63, San Diego, California, USA, 2003.
6. T. Smith. eBay and your IT supply chain. TechWeb's Internet Week, April 12, 2002, url = "http://www.internetweek.com/story/showArticle.jhtml?articleID=6406064" (accessed on December 10, 2003), 2002.