Indian Institute of Management Calcutta

Working Paper Series

WPS No. 749
July 2014

Determining the Deadness Levels of Packages in Online Multi-unit Combinatorial Auctions

Anup K Sen
Professor, Indian Institute of Management Calcutta
D. H. Road, Joka P.O.Kolkata 700 104
sen@iimcal.ac.in

Amitava Bagchi
Professor, Heritage Institute of Technology
East Kolkata Township, Anandapur, Kolkata 700107
amitava.bagchi@heritageit.edu

Determining the Deadness Levels of Packages in Online Multi-unit Combinatorial Auctions

Anup K Sen
Indian Institute of Management Calcutta

Amitava Bagchi
Heritage Institute of Technology Kolkata

Abstract

In online iterative combinatorial auctions, the deadness level (DL) of a package serves as a tight lower bound on a fresh bid that can be meaningfully placed on the package. Computational methods exist for determining the DL values of packages in the single-unit case. But when there are multiple identical units of items, these levels are hard to determine, and no closed form expression or computational method has been proposed as yet. This note examines the properties of package DLs in the multi-unit case; it provides theoretical results with supporting illustrative examples and presents for the first time an exact method for computing DL values. This could help to promote more widespread use in industry of such online auctions.

Keywords: Multi-unit combinatorial auctions, online combinatorial auctions, deadness levels (DLs)**,** intelligent agent-based systems

1. INTRODUCTION

When complementary items are put on auction, bidders must be permitted to bid on packages (*i.e.*, bundles) of items. In such combinatorial auctions (CAs), bidders need guidance from the seller to estimate package valuations correctly. When only one unit of each item is on auction, we can make use of the notion of 'quotes' (Sandholm 2002) to define a lower bound (*Deadness Level DL*) and an upper bound (*Winning Level WL*) for each package. These bounds can be readily computed, enabling bidders to place bids in the intermediate region between them. But when multiple identical units of items are on auc ( )Tj EMC /LBo(L)]Tj 6.7 [Td [(w)2 of od (a)Tj 4 possible for a bidder to restrict

2.1 BASIC PROPERTIES

We now list some basic theoretical properties of online multi-unit CAs. In the single-unit case, the following results are known to hold (Adomavicius and Gupta, 2005):

i) $Maxfit(p,t) + Maxfit(S\backslash p,t) \leq Maxfit(S,t)$;
ii) $WL(p,t) = Maxfit(S,t) - Maxfit(S\backslash p,t)$;
iii) $DL(p,t) = Maxfit(p,t)$;
iv) *DL*(*p*,*t*) ” *WL*(*p*,*t*);
v) *DL*(*p*,*t*) is non-decreasing in *t*.

These results do not all generalize to the multi-unit case. For example, (i) and (iii) are no longer valid. In Table 1, at all five instants, $MDL(p,t) < Maxfit(p,t)$ and therefore $MDL(p,t)$ cannot be directly determined by computing $Maxfit(p,t)$. In addition, $Maxfit(p,t) + Maxfit(S\backslash p,t) > Maxfit(S,t)$, because some of thmcannot ber(c)4(a)4(nnot)-2(b)-10(e7/8 0 Td [(0(e7/8 0 noo2not)-2(b)-10(er(c

- While Compute_MDL() runs fast on the examples we have tried, in the worst case the running time is exponential in the number of bids. One way to speed it up would be to try and develop an incremental version that would determine, given a package *p*, the value of *MDL*(*p*,*t*+1) from the value of *MDL*(*p*,*t*) with a minimum of computation.

$S = X^9Y^9$ $p = X^3Y^3$ $S\backslash p = X^6Y^6$									
Instant	*Package*	*Bid*	*Maxfit*		*MWL*(*p*)	*r and* B_1	*s* and B_2	*newM*	*MDL*(*p*)
			S	*S\p*					
4	Y^3	85	255	205	50	X^5Y^5 = (1) + (3) + (4) = 205	XY = 70	305	50
5	X^4Y^3	125	330	210	120	Y^5 = (2) + (4) = 135	X^6Y = 120	405	75
6	X^2Y^4	130	350	225	125	Y^5 = (2) + (4) = 135	X^6Y = 135	430	80
7	XY	55	360	260	100	X^4Y =(1) + (7) =100	X^2Y^5 = 175	445	85
8	X^2	30	370	265	105	X^6Y =(1) + (7) + (8) =130	Y^5 = 145	465	95
9	X^2Y	85	400	275	125	X^3Y^5 =(4) + (7) + (9)=225	X^3Y = 70	505	105
10	XY^2	100	450	320	130	X^6Y^2 =(1) + (7) + (9) =185	Y^4 = 135	580	130

Table 2: Values of MDL(p) in Example 2

Instant t	*Package q*	*Bid*	*MDL(q)*

the bid values on the packages that constitute r_1. We choose the value of B_2 in such a way that $B + B_1 + B_2 = Maxfit(S,t)$; this cannot occur for a smaller value since $MDL(r,t) = B$. Some of the online