

CP-IP Techniques for the Bid Evaluation in Combinatorial Auctions

Alessio Guerri and Michela Milano

DEIS, University of Bologna
Viale Risorgimento 2
40136 Bologna, Italy

{aguerrri,mmilano}@deis.unibo.it

Abstract. Combinatorial auctions are an important e-commerce application where bidders can bid on combinations of items. The problem of selecting the best bids that cover all items, i.e., the Winner Determination Problem (WDP), is NP-hard. In this paper we consider the time constrained variant of this problem, that is the Bid Evaluation Problem (BEP) where temporal windows and precedence constraints are associated to each task in the bid. We propose different algorithms based on CP, IP and a hybrid approach based on both of them. We show that even the simplest pure CP based approach outperforms the only existing approach. We selected a set of algorithms which do not dominate each other. We identified a set of instance-dependent structural features that enable to select the best class of algorithms to apply. This is the first step toward an automatic algorithm selection in algorithm portfolios.

1 Introduction

Business to business e-commerce applications are becoming more and more popular. Among them, auctions are a way of allocating items among autonomous and self-interested agents. Items are not limited to goods, but can represent also resources and services.

In this paper we consider *combinatorial auctions*, see [4]. Among M items, bidders can bid on combinations of items, and associate a price for each combination. The auctioneer should solve the Winner Determination Problem (WDP), i.e., he should choose the best bids that cover all items at a minimum cost or maximum revenue. This problem is NP-hard.

A variant of this problem is the so called Bid Evaluation Problem (BEP) for coordinated tasks. When the auctioneer should, for example, buy a set of services, he should also consider temporal constraints. Therefore, items in the bid are associated to a temporal window, a duration, and are linked by precedence constraints. In this case, beside the WDP, the auctioneer should maintain feasibility of the temporal constraints. To our knowledge, the only system that tackles this problem is MAGNET (Multi-Agent Negotiation testbed) [1] and it is based on Integer Programming and Simulated Annealing.

Surprisingly, Constraint Programming (CP) has been very rarely used to solve either the WDP or the BEP, while we think CP can be successfully used as an effective tool for modelling and solving problems related to combinatorial auctions. In particular, CP can be effective when additional constraints are introduced.

In this paper, we propose different algorithms for the BEP: two variants of a pure CP algorithm, one based on Limited Discrepancy Search (LDS) and one on Depth First Search (DFS); two approaches based on pure IP and two hybrid approaches merging CP and IP, one based on LDS and one on DFS. We show that even the simplest approach we developed, based on pure CP, outperforms the one presented in MAGNET¹. We evaluated all algorithms and discovered that those based on DFS are always dominated by the others. Among the remaining algorithms none of them dominates all the others, so we tried to select among the set of instance-dependent structural features proposed in [3] the ones that allow to select the best algorithm. An interesting result achieved is that the standard deviation of the Clustering Coefficient provides a clear indication if to use an IP or a CP based algorithm.

2 Bid Evaluation Problem: Model and Algorithms

We have different variants of combinatorial auctions. In this paper, we consider **single unit reverse auctions**, where the auctioneer wants to buy a set M of distinguishable items (services) minimizing the cost.

Each bidder j ($j = 1..n$) posts one or more bids. A bid is represented as $B_j = (S_j, Est_j, Lst_j, D_j, p_j)$ where a set $S_j \subseteq M$ of services is proposed to be sold at the price p_j . Est_j and Lst_j are lists of earliest and latest starting time of the services in S_j and D_j their duration.

The BEP can be seen as a variant of the WDP where a set of temporal constraints define the feasibility of the assignments computed by the WDP.

We implemented four algorithms, plus two variants based on DFS and LDS, to solve the BEP. Two algorithms are based on the IP model: the first (referred in tables to as IP) is a traditional complete solver implementing Branch and Bound based on linear relaxation, while the second (referred in tables to as LR+IP) is an incomplete approach that solves the linear relaxation of the problem, then ranks the variables according to their shadow price, and finally solves the IP problem considering only the first $p\%$ variables, where p is a parameter to be experimentally tuned.

One algorithm is based on a pure CP model (referred in tables to as CP). Starting from the same model, one variant explores the search tree with DFS and one with LDS. In both cases, the heuristic used to select the variable value is the bid-price divided by bid-size, that is $p_i/|S_i|$. The last approach, referred to as LR+CP in tables, performs an indeed quite loose but effective integration. Starting from a CP model, we solve a linear relaxation at the root node and we

¹ This software has been kindly provided by the authors J. Collins and M. Gini.

order values based on shadow prices. Again we have two variants, one based on DFS and one on LDS.

3 Comparing CP and MAGNET

We first compared the pure CP algorithms we developed and MAGNET [1] on instances generated using MAGNET itself. We ran our experiments on a 2.4Ghz Intel Pentium 4 with 512Mb RAM. We considered four kinds of MAGNET instances with a number of tasks between 5 and 20 and bids between 15 and 400.

For each instance set we used our CP algorithms, MAGNET implementing Simulated Annealing (referred to as SA in figures) and, when possible, MAGNET using Integer Programming (M-IP).

In the first two sets of experiments (5 tasks and 15 bids and 10 tasks and 35 bids), the M-IP approach always finds the optimal solution, while in the third (10 tasks and 100 bids) it does not provide the optimal solution within 15 minutes. In the first set also SA provides the optimal solution, while, in the second set, it provides the optimal solution only in the 60% of the cases. In the third set of experiments it never provides the optimal solution. Our CP approaches always finds the optimal solution in all instance sets. Mean search time for the first three sets are depicted in Figure 1, where for each group values are normalized w.r.t. mean value over all algorithms.

As concern the fourth instance set (20 tasks and 400 bids) none of the approaches find the optimal solution within 15 minutes, but solutions found found by the CP based approaches are, on average, 30% better than those produced by MAGNET. Moreover, the time to produce the best solution is in general considerably lower than 15 minutes and our algorithms always outperform MAGNET. The relative quality of SA with respect to LDS and DFS is also depicted in Figure 2 where we show the trend of the solution quality for hard instances with SA, DFS and LDS. M-IP approach did not find any solution.

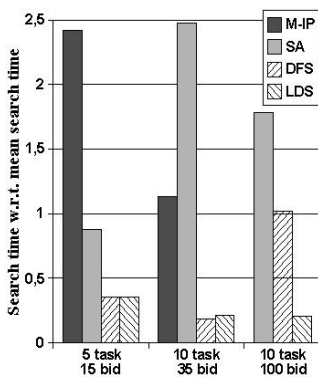


Fig. 1. Comparison between algorithms

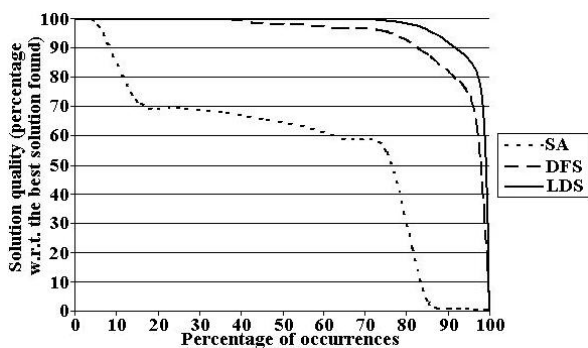


Fig. 2. Comparison between algorithms on instances with 20 tasks and 400 bids

4 Experimental Results of CP, IP and Hybrid Approaches

In this section, we provide results on instances generated using both MAGNET and CATS, a suite for generating realistic auction instances [2]. CATS instances are more realistic, enabling to set an higher bid-size and bid-size variability. We generated problems with 10, 15, 20 and 30 tasks, with a number of bids growing from 40 up to 1000 and with a variable tasks-per-bid values.

In Figure 3 we present results for all algorithms described in section 2 except for those using DFS since they are always outperformed by those using LDS. Each group in the histogram represents a different instance set, having the mean tasks-per-bid value expressed in x-axis. Y-axis values represent, for each group, mean search time normalized w.r.t. mean over all algorithms. Some instances are best solved by CP-based approaches (namely CP and LR+CP), while others by IP approaches (namely IP and LR+IP), depending on the number of tasks-per-bid (see Section 5 for a deeper analysis).

Finally, we ran experiments on hardest problems, with 30 tasks, 1000 bids and a growing tasks-per-bid value. We found that only the IP-based approach provided results. In Table 1 we show the mean search time for both complete and incomplete IP solver (i.e., IP and LR+IP). For the incomplete approach, the percentage of variables considered is reported in the last column. Only in the first 3 instance sets it was possible to prove optimality over all instances.

5 Problem Structure Analysis

In this section, we are interested in identifying a set of instance-dependent parameters that help in determining the best algorithm to solve the instance itself. From tables in previous section, we noticed that tasks-per-bid parameter roughly influences algorithms' quality. Here we are interested in more precise parameter.

Starting from the notable classification in [3], we extracted from each instance the 25 features described in the paper. We refer to the bid graph, where each node represents a bid and each edge stands between two bids if there is one or more constraints containing that bids. An interesting result achieved is that there

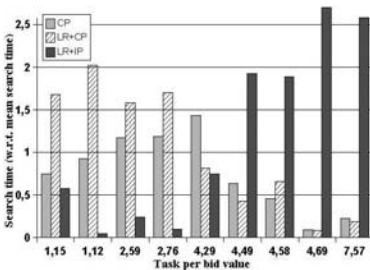


Fig. 3. Comparison between algorithms

Table 1. Comparison between IP algorithms on instances with 30 tasks and 1000 bids

Tasks for Bid	Search time (ms)	CR%	
	IP	LR+IP	
1.40	36328	1235	70
2.49	242281	3500	45
3.34	900000	6975	25
4.62	-	19681	25
6.52	-	25969	30

is a correspondence between the standard deviation of the Clustering Coefficient (SDCC) in the bid graph and the experimental results. SDCC is a measure of the *local cliqueness*. Typically, in our instances this value ranges from 0.02 e 0.2, and each time it is greater than 0.09, the IP-based approach is preferable to the CP-based one. It is worth noting the fact that, if SDCC is close to 0.09 both approaches have satisfactory behaviors. For the instances considered it is a *systematic* result, but, unfortunately, for larger instances this feature extraction takes too much time. Therefore, we looked for a similar but easier-to-compute parameter. We observed that there is a correspondence between SDCC and the Edge Density (ED) in the bid graph. ED can range from 0 to 1, and we observed that it is in inverse proportion with SDCC.

We have identified three significant ranges for the ED: if $ED < 0.5$, SDCC is always greater than 0.09 (thus the IP-based approach is preferable); if $ED > 0.75$, SDCC is always lower than 0.09 (thus the CP-based approach is preferable); if $0.5 < ED < 0.75$, we do not have a clear indication of SDCC and therefore on the preferable approach. In this case, we recompute ED allowing multiple edges between the nodes (when more than one constraint is present among them). If new ED is significantly greater than the previous one, the CP-based approach is to be preferred; otherwise, if it remains quite unchanged, the best approach is IP.

If the IP approach is the technique of choice, we can use either the complete approach or the incomplete one named LR+IP. So we have to tune the percentage of variables to be considered. We did not find a systematic correspondence between those choices and any feature we calculated, but from the L_1 and L_∞ norms of the integer slack vector we can often find out a good superior boundary to the percentage of variables to be considered using IP approach. These norms are in a way a measure of how fractional is the linear relaxation solution; we noticed that, in general, the higher the norms, the higher the percentage of variables to include to obtain the best solution.

Acknowledgement

This work was supported by the SOCS project, funded by the CEC, contract IST-2001-32530. We would like to thank Andrea Lodi and Andrea Roli for useful discussion. In addition, we warmly thank Maria Gini and John Collins for providing the MAGNET software and for their assistance in using it.

We also thank anonymous reviewers for useful comments.

References

1. J. Collins and M. Gini, An integer programming formulation of the bid evaluation problem for coordinated tasks, *Mathematics of E-Commerce*, Springer-Verlag.
2. K. Leyton-Brown, M. Pearson and Y. Shoham, Towards an Universal Test Suite for Combinatorial Auction Algorithms, *Proc. EC00*, 2000.
3. K. Leyton-Brown, E. Nudelman and Y. Shoham, Learning the Empirical Hardness of Optimization Problems: The Case of Combinatorial Auctions, *Proc CP02*, 2002.
4. T. Sandholm, Algorithm for optimal winner determination in combinatorial auction, *AI*, Vol 135, 2002.