Combinatorial Auctions and Knapsack Problems*

Terence Kelly
Hewlett-Packard Laboratories
kterence@hpl.hp.com

Abstract

This note summarizes the connection between winner determination problems (WDPs) in multi-unit combinatorial exchanges and generalized knapsack problems (KPs); see [2] for an extensive treatment. Taxonomies of sealed-bid auction WDPs and of KPs align precisely, i.e., the two classes of problems are identical. *The relationship between KPs and WDPs has received remarkably little attention in E-commerce and multi-agent systems literature. However the WDP-KP connection is important because it allows us to leverage a vast body of Operations Research literature for developing and benchmarking WDP solvers. For low-dimensional multi-unit CA WDPs (i.e., few types of goods), extremely simple dynamic-programming algorithms are available whose time and memory requirements are* linear *(in the pseudo-polynomial sense) in three out of four natural measures of problem size.*

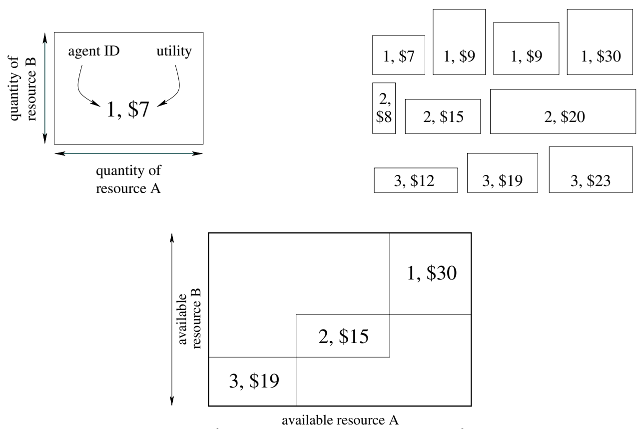

Figure 1. Illustration of 2-good-type MUCA WDP. *Top left:* a (bundle, utility) pair. *Top right:* XOR bids from several agents. *Bottom:* a feasible solution to the WDP.

1. Introduction

In a multi-dimensional knapsack problem (MDKP) we are given a container with several capacity constraints (e.g., weight and volume) and a set of items. Each item has a profit and several capacity requirements corresponding to the container's constraints. We seek to fill the container as profitably as possible without breaking it, so to speak. If items are partitioned into classes and we are required to choose exactly one item from each class, we have a multi-dimensional *multiple-choice* knapsack problem (MDMCK).

A sealed-bid combinatorial auction (CA) involves several types of goods; multi-unit CAs (MUCAs) involve multiple identical units of each good type. In the most general case, bids in such auctions may report utility for any number of bundles of goods. XOR bids with indivisible bundles allow agents to report an arbitrarily long list of (bundle, utility) pairs; each agent must receive exactly one bundle from its list. The winner-determination problem (WDP) in such an auction is to maximize aggregate reported utility subject to availability constraints for each type of good; Figure 1 illustrates the 2-D case.

This WDP is exactly the MDMCK problem; they differ only in the trivial sense that we associate different connotations with variables. Furthermore taxonomies of sealed-bid auction types and generalized KPs align precisely, as shown in Table 1. Neither generalized knapsack problems nor most of the solution methods developed for them require that the coefficients in problem instances be nonnegative; MDMCK and its special cases therefore encompass arbitrary two-sided exchanges in which bids express willingness to release some goods while simultaneously acquiring others.

Remarkably, the connection between auction WDPs and generalized KPs has largely been overlooked in recent E-commerce research. This note summarizes the auction-knapsack connection and the benefits that result from recognizing it. See [2] for an extensive treatment of the subject and a thorough literature review.

* AAMAS'04 poster

$Id: kellyt_knapsack.tex,v 1.3 2004/04/30 19:53:47 kterence Exp $

good types	units	bundles	common name / examples	winner-determination problem
S	S	S	first price	find max
S	M	S	double auctions, single-quantity bids	classic 0-1 KP; subset-sum if #units ∝ utility
S	M	M	double auctions, XOR bids	multiple-choice KP (MCKP)
M	S	S	“combinatorial auctions”	weighted set packing (WSP)
M	S	M	single-unit CA, XOR bids	reducible to WSP via “dummy goods”
M	M	S	multi-unit CA, single-bundle bids	multi-dimensional KP (MDKP)
M	M	M	multi-unit CA, XOR bids	MDMCK

Table 1. Auction types and WDPs (S=single, M=multiple).

2. KP solvers for WDPs

Knapsack problems are simple, intuitive, and far more widely known than special cases that have received much attention in E-commerce literature, e.g., weighted set packing. A wide variety of solution methods exist for generalized KPs, offering a range of simplicity/efficiency tradeoffs. KP-based WDP solvers offer complete generality and require no compromises in problem formulation: they support arbitrary agent preference expression, arbitrary disposal costs for unallocated goods, atomic (indivisible) bid semantics, and fully general two-sided exchanges.

Most importantly, recognizing the auction-knapsack connection allows us to leverage decades of sophisticated KP research. Interactions between KP instances and solution methods have been studied intensively, and much is known about the structure of hard problem instances. This knowledge can guide both the design and the evaluation (benchmarking) of WDP solvers. OR results on *hard* KP instances can complement E-commerce research that attempts to generate *typical* WDP instances [4].

3. Dynamic Programming

The solution of MDMCK is known to require time exponential in the dimensionality of the problem instance, i.e., in the number of good types in a MUCA. However this is not the only natural measure of WDP instance size; others include the number of agents, the number of units of each type of good available, and the number of bundles in bids. The simplest solution method for MDMCK is dynamic programming (DP). A straightforward DP solver requires $O(RTBN_1N_2\cdots N_R)$ time and $O(TN_1N_2\cdots N_R)$ memory, where T is the number of agents, R is the number of good types, N_r is the number of units of good type r available, and B is the number of bundles in agent bids.

These computational requirements are exponential in the number of good types R, but they are *linear* (in the pseudo-polynomial sense) in all other natural measures of problem size. Low-dimensional problems arise in practical contexts, e.g., the problem of allocating computational resources across agents in modern data centers [2]. Straightforward DP solvers for low-dimensional MDMCK offer advantages over more elaborate solvers developed for high-dimensional problems, and they may be preferable for hard problem instances.

4. Related Work

In a few cases, e.g., [3], E-commerce research has exploited connections between specialized auction WDPs and KPs. An extensive literature review, however, revealed that this connection has not been leveraged extensively or treated in detail in E-commerce literature; see [2] for details. Somewhat ironically, the first detailed published discussion of the WDP-KP connection occurs in a very recent KP text written by Operations Researchers whose main interest is not in specific application domains such as E-commerce [1, pp. 478–482].

Much of the late-1990s E-commerce research on WDP solver design and benchmarking might have benefited from OR literature on KPs from years or decades earlier. Several early WDP benchmarks involved easy problem instances, an error that the OR community had identified and rectified in the context of KPs much earlier. Furthermore WDP solvers themselves might have built upon generalized KP algorithms developed from the 1970s onward.

References

[1] H. Kellerer, U. Pferschy, and D. Pisinger. *Knapsack Problems*. Springer, 2004.

[2] T. Kelly. Generalized knapsack solvers for multi-unit combinatorial auctions. Technical Report HPL-2004-21, HP Labs, Feb. 2004.

[3] A. Kothari, D. C. Parkes, and S. Suri. Approximately-strategyproof and tractable multi-unit auctions. In *ACM E-Commerce*, June 2003.

[4] K. Leyton-Brown, M. Pearson, and Y. Shoham. Towards a universal test suite for combinatorial auction algorithms. In *ACM E-Commerce*, Oct. 2000.