



UvA-DARE (Digital Academic Repository)

Multiagent resource allocation in k-additive domains: Preference representation and complexity

Chevaleyre, Y.; Endriss, U.; Estivie, S.; Maudet, N.

DOI

[10.1007/s10479-008-0335-0](https://doi.org/10.1007/s10479-008-0335-0)

Publication date

2008

Published in

Annals of Operations Research

[Link to publication](#)

Citation for published version (APA):

Chevaleyre, Y., Endriss, U., Estivie, S., & Maudet, N. (2008). Multiagent resource allocation in k-additive domains: Preference representation and complexity. *Annals of Operations Research*, 163(1), 49-62. <https://doi.org/10.1007/s10479-008-0335-0>

General rights

It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations

If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: <https://uba.uva.nl/en/contact>, or a letter to: Library of the University of Amsterdam, Secretariat, Singel 425, 1012 WP Amsterdam, The Netherlands. You will be contacted as soon as possible.

Multiagent resource allocation in k -additive domains: preference representation and complexity

Yann Chevaleyre · Ulle Endriss · Sylvia Estivie · Nicolas Maudet

Published online: 22 March 2008
© The Author(s) 2008

Abstract We study a framework for multiagent resource allocation where autonomous software agents negotiate over the allocation of bundles of indivisible resources. Connections to well-known combinatorial optimisation problems, including the winner determination problem in combinatorial auctions, shed light on the computational complexity of the framework. We give particular consideration to scenarios where the preferences of agents are modelled in terms of k -additive utility functions, i.e. scenarios where synergies between different resources are restricted to bundles of at most k items.

Keywords Resource allocation · Negotiation · Multiagent systems · Preference representation · Computational complexity

1 Introduction

Multiagent resource allocation (Chevaleyre et al. 2006) is an interdisciplinary field, located at the intersection of computer science and artificial intelligence research on the one hand,

A preliminary version of this paper has been presented at the 1st DIMACS-LAMSADE Workshop on Computer Science and Decision Theory (Chevaleyre et al. 2004). We would like to thank the anonymous reviewers for their helpful comments on the manuscript.

Y. Chevaleyre · N. Maudet
LAMSADE, Université Paris-Dauphine, Paris, France

Y. Chevaleyre
e-mail: yann.chevaleyre@lamsade.dauphine.fr

N. Maudet
e-mail: maudet@lamsade.dauphine.fr

U. Endriss (✉)
ILLC, University of Amsterdam, Amsterdam, The Netherlands
e-mail: ulle@ilic.uva.nl

S. Estivie
LAMIH, Université de Valenciennes, Valenciennes, France
e-mail: sylvia.estivie@univ-valenciennes.fr

and the socio-economic sciences on the other. It is concerned with the study of resource allocation processes within systems of autonomous agents, that not only have preferences over alternative allocations of resources but also actively participate in computing allocations. Special emphasis is put on the computational aspects of these processes (this is the distinguishing feature that differentiates multiagent resource allocation from classical questions in microeconomics). The multiagent systems paradigm (Wooldridge 2002), developed in artificial intelligence, provides a useful framework in which to study such resource allocation problems. Furthermore, the way agents represent their individual preferences over alternative bundles of resources is an important factor; this is where decision theory comes into play.

Distributed systems in which autonomous software agents interact with each other, in either cooperative or competitive ways, can often be usefully interpreted as *societies of agents*; and we can employ formal tools from microeconomics to analyse such systems. If we model the interests of individual agents in terms of a notion of *individual welfare*, then the overall performance of the system provides us with a measure of *social welfare*. Individual welfare may, for instance, be measured by defining a utility function mapping “states of affairs” (outcomes of an election, allocations of resources, agreements on a joint plan of action, etc.) to numeric values. The concept of social welfare, as studied in welfare economics, is an attempt to characterise the well-being of a society in relation to the welfare enjoyed by its individual members (Arrow et al. 2002; Moulin 1988).

In this paper, we study a framework for multiagent resource allocation where autonomous agents agree on a sequence of multilateral deals to exchange sets of indivisible resources in order to improve their respective levels of individual welfare. Recent results pertaining to this framework concern the feasibility of ensuring convergence to an allocation of resources that is optimal from a social point of view (Sandholm 1998; Endriss et al. 2006), as well as (certain aspects of) the complexity of doing so, in terms of both computational costs (Dunne et al. 2005) and the amount of communication required to enable negotiation (Endriss and Maudet 2005).

A central parameter in a multiagent resource allocation problem is the language used to represent the preferences of agents over alternative bundles of resources. In the framework discussed here, we assume that agents use utility functions to model their preferences and we are going to discuss two alternative languages for defining such functions. As we shall argue, one of these, the so-called k -additive form, is particularly suitable in domains where agents may be uncertain about their true preferences, as it provides a natural way of encoding a hierarchy of progressively more and more accurate approximations to the agent’s complete utility function.

In Sect. 2, after introducing the negotiation framework used in this paper, we briefly recall two of the aforementioned convergence results. As we shall see, in cases where the utility functions used by agents to model their preferences over alternative bundles of resources are *additive*, it is sufficient to use very simple negotiation protocols that only cater for deals involving a single resource at a time. This result suggests to investigate generalisations of the notion of additivity, and hence we consider the case of k -additive functions, as studied, for instance, in the context of fuzzy measure theory (Grabisch 1997). The notion of k -additivity suggests an alternative representation of utility functions, which we introduce in Sect. 3. We show that this representation is as expressive as the “standard” representation (which involves listing the utility values for all possible bundles) and that it often allows for a more succinct representation of preferences.

Nevertheless, it turns out that the positive result on the structural complexity of deals obtained for additive functions *cannot* be generalised in the expected manner. Counterexamples are given in Sect. 4. In Sect. 5, we discuss connections between our framework and some well-known *combinatorial optimisation* problems (Ausiello et al. 1999). These can be used to prove *NP-hardness* results for the decision problem associated with the task of finding a socially optimal resource allocation. We prove complexity results with respect to both the standard representation of utility functions and the representation based on k -additivity. In this context, we also discuss connections of our optimisation problem to the winner determination problem in *combinatorial auctions* (Cramton et al. 2006). We are going to point out connections between different ways of representing utility functions and different *bidding languages* for such auctions along the way. Our conclusions are presented in Sect. 6.

2 Negotiating over indivisible resources

An instance of our negotiation framework consists of a finite set of (at least two) *agents* \mathcal{A} and a finite set of indivisible *resources* \mathcal{R} . A resource *allocation* A is a partitioning of the set \mathcal{R} amongst the agents in \mathcal{A} . For instance, given an allocation A with $A(i) = \{r_3, r_7\}$, agent i would own resources r_3 and r_7 . Given a particular allocation of resources, agents may agree on a (multilateral) *deal* to exchange some of the resources they currently hold. In general, a single deal may involve any number of resources and any number of agents. It transforms an allocation of resources A into a new allocation A' ; that is, we can define a deal as a pair $\delta = (A, A')$ of allocations (with $A \neq A'$).

Each agent $i \in \mathcal{A}$ is equipped with a *utility function* u_i mapping bundles of resources (subsets of \mathcal{R}) to rational numbers (note that we do not impose any restrictions on utility functions). A deal may be coupled with a number of monetary side payments to compensate some of the agents involved for an otherwise disadvantageous deal. We call a deal *rational* if and only if it results in a gain in utility (or money) that strictly outweighs a possible loss in money (or utility) for each of the agents involved in that deal (Endriss et al. 2006). This notion of rationality can be formalised using the concept of a *payment function* p mapping agents to rational numbers. Such a function has to satisfy the side constraint $\sum_{i \in \mathcal{A}} p(i) = 0$, i.e. the overall amount of money in the system remains constant. If $p(i) > 0$, then agent i *pays* the amount of $p(i)$, while $p(i) < 0$ means that it *receives* the amount of $-p(i)$.

Definition 1 (Rational deals) A deal $\delta = (A, A')$ is rational if and only if there exists a payment function p such that $u_i(A'(i)) - u_i(A(i)) > p(i)$ for all $i \in \mathcal{A}$, except possibly $p(i) = 0$ for agents i with $A(i) = A'(i)$.

While individual agents have their own interests, as a system designer, we are interested in the *social welfare* associated with a given allocation.

Definition 2 (Social welfare) The social welfare $sw(A)$ of an allocation A is defined as follows:

$$sw(A) = \sum_{i \in \mathcal{A}} u_i(A(i)).$$

This is the *utilitarian* definition of social welfare. We should stress that other notions of social welfare have been developed as well (Arrow et al. 2002; Moulin 1988).

What is the connection between the “local” notion of rationality and the “global” notion of social welfare? As shown in previous work (Endriss et al. 2006), *a deal is rational if and only if it results in an increase in social welfare*. The following result, due to Sandholm (1998), goes even further by establishing that *any* sequence of rational deals is bound to converge to a socially optimal allocation:

Proposition 1 (Convergence) *Any sequence of rational deals will eventually result in an allocation of resources with maximal social welfare.*

This means that (i) there can be no infinite sequence of deals all of which are rational, and (ii) once no more rational deals are possible the agent society must have reached an allocation that has maximal social welfare. The crucial aspect of this result is that *any* sequence of deals satisfying the rationality condition will cause the system to converge to an optimal allocation. That is, whatever deals are agreed on in the early stages of the negotiation, the system will never get stuck in a local optimum and finding an optimal allocation remains an option throughout.

A drawback of the general framework is that the above result only holds if deals involving any number of resources and agents are admissible (Sandholm 1998; Endriss et al. 2006). In some cases this problem can be alleviated by putting suitable restrictions on the utility functions agents may use to model their preferences. A particularly simple example is the class of additive functions. A utility function is called *additive* if and only if the value ascribed to a set of resources is always the sum of the values of its members. For additive utilities, the following stronger convergence result is known (Endriss et al. 2006):

Proposition 2 (Additive domains) *If all utility functions are additive, then any sequence of rational deals involving only a single resource each will eventually result in an allocation of resources with maximal social welfare.*

This result is of great practical relevance, because it shows that it is sufficient to design negotiation protocols for single resources (rather than sets) and thereby also just pairs of agents (rather than larger groups) for applications in which preferences can be modelled in terms of additive utility functions. In the next section, we are going to introduce a generalisation of this notion of additivity.

3 Preference representation

An agent’s utility function may be represented in different ways. This situation is similar, for instance, to the case of combinatorial auctions, where one can use different *bidding languages* to express the preferences of the participating agents (Nisan 2000; Sandholm 2002). Maybe the most intuitive representation of a utility function is the *bundle form* (or *explicit form*), which amounts to listing all bundles of resources to which the agent assigns a non-zero value, together with their value.

An alternative representation is based on the notion of *k*-additive functions, which have been studied in the context of fuzzy measure theory (Grabisch 1997). Given a natural number *k*, a utility function is called *k*-additive if and only if the utility assigned to a bundle of resources *R* can be represented as the sum of basic utilities ascribed to subsets of *R* with cardinality $\leq k$. More formally, a utility function u_i is *k*-additive iff there exists a set of

coefficients $\{\alpha_i^T \mid T \subseteq \mathcal{R}\}$ such that $\alpha_i^T = 0$ whenever $|T| > k$, and the following holds for all $R \subseteq \mathcal{R}$:

$$u_i(R) = \sum_{T \subseteq R} \alpha_i^T. \tag{1}$$

That is, agent i enjoys an increase in utility of α_i^T when it owns all the items in T together, i.e. α_i^T represents the synergetic value of this bundle. If a utility function is defined in terms of such coefficients, we say that it is given in k -additive form.

When describing examples, we are going to use a simplified notation (resembling that of a polynomial with variables r_i taking the values 0 or 1). For instance, $u_i = 3 \cdot r_1 - 2 \cdot r_2 \cdot r_3$ represents a 2-additive utility function with two non-zero coefficients: $\alpha_i^{\{r_1\}} = 3$ and $\alpha_i^{\{r_2, r_3\}} = -2$. In the bundle form, the definition of this function would require the specification of five non-zero values (assuming $\mathcal{R} = \{r_1, r_2, r_3\}$): $u_i(\{r_1\}) = 3$, $u_i(\{r_1, r_2\}) = 3$, $u_i(\{r_1, r_3\}) = 3$, $u_i(\{r_2, r_3\}) = -2$, and $u_i(\{r_1, r_2, r_3\}) = 1$.

Utility functions that are k -additive with $k = 1$ are like the additive functions discussed in the previous section (except that they also allow for a non-zero utility value to be assigned to the empty set). Hence, the concept of k -additivity is a generalisation of the familiar concept of additivity. In fact, as we are going to show next, k -additive utility functions cover a whole range of utility functions, from the very simple additive functions to the most general utility functions without any restrictions whatsoever.

Proposition 3 (Expressive power) *Any utility function can be represented as a k -additive function with $k = |\mathcal{R}|$.*

Proof Let u_i be any utility function mapping subsets of \mathcal{R} to rational numbers. We recursively define coefficients α_i^R for $R \subseteq \mathcal{R}$ as follows:

$$\begin{aligned} \alpha_i^\emptyset &= u_i(\emptyset), \\ \alpha_i^R &= u_i(R) - \sum_{T \subset R} \alpha_i^T \text{ for all } R \subseteq \mathcal{R} \text{ with } R \neq \emptyset. \end{aligned}$$

Hence, $u_i(R) = \sum_{T \subseteq R} \alpha_i^T$. This is a k -additive utility function for $k = |\mathcal{R}|$. □

Given a utility function, we can also compute the values of the coefficients in the k -additive form directly, using the so-called *Möbius inversion* (Rota 1964; Grabisch 1997). Namely, the solution to (1) is given by the following formula:

$$\alpha_i^R = \sum_{T \subseteq R} (-1)^{|R \setminus T|} \cdot u_i(T).$$

In analogy to Proposition 3, clearly, the bundle form is also fully expressive, i.e. our two ways of representing utility functions are equivalent in the sense that they can both express any utility function over the set of resources \mathcal{R} . Besides expressive power, another important consideration concerns the *succinctness* of a representation. It turns out that neither of the two forms of representation is more succinct in all cases. In fact, as we are going to see next,

there are cases where translating a utility function given in k -additive form into the bundle form results in an exponential blow-up of the representation, and vice versa.¹

To state these results, we use the concept of polynomial simulation. We say that one representation language can *polynomially simulate* another language if and only if there is a polynomial function f such that for any given utility function u , if ℓ is the size of the representation of u in the second language then the size of the representation of u in the first language will be at most $f(\ell)$.

Proposition 4 (Efficiency of the k -additive form) *The bundle form cannot polynomially simulate the k -additive form of representing utility functions.*

Proof We prove the claim by giving an example for a utility function with a representation that is linear in the size of \mathcal{R} for the k -additive form, but exponential for the bundle form. Consider a utility function u_i that maps a bundle of resources to the number of elements in that bundle. This is a 1-additive function, which requires the specification of exactly $|\mathcal{R}|$ coefficients in the k -additive form (namely $\alpha_i^R = 1$ for all bundles R with $|R| = 1$). For the bundle form, however, the specification of a utility value for each of the $2^{|\mathcal{R}|} - 1$ non-empty bundles is required. \square

Proposition 5 (Efficiency of the bundle form) *The k -additive form cannot polynomially simulate the bundle form of representing utility functions.*

Proof We give an example for a utility function with a representation that is linear in the size of \mathcal{R} for the bundle form, but exponential for the k -additive form. Consider a utility function u_i that assigns 1 to any bundle consisting of a single resource and 0 to any other bundle. In the bundle form, u_i requires the specification of a utility value for exactly $|\mathcal{R}|$ bundles (namely those with just a single element). For the k -additive form, on the other hand, the Möbius inversion shows that we have to set $\alpha_i^R = |R| \cdot (-1)^{|R|+1}$. This is different from 0 for any of the $2^{|\mathcal{R}|} - 1$ non-empty subsets of \mathcal{R} . Hence, u_i requires the specification of an exponential number of coefficients in the k -additive form. \square

Taken together, Propositions 4 and 5 show that the preference representation languages given by the bundle form and the k -additive are *incomparable* in terms of succinctness. The examples given in our two proofs are extreme cases, where one form of representation is exponentially more succinct than the other. While the difference is not always going to be this strong, choosing the right representation for a given problem domain is still important. Broadly speaking, the k -additive form will typically be more succinct in cases where there are only limited synergies between different items. This is likely to be the case for many application domains, which makes this a useful language for expressing utilities in practice.

In general, either representation may require the specification of up to $2^{|\mathcal{R}|}$ values. However, for utility functions that are k -additive for some value of k , the k -additive form requires at most $\sum_{i=0}^k \binom{|\mathcal{R}|}{i}$ coefficients to be specified. This is equal to $2^{|\mathcal{R}|}$ for $k = |\mathcal{R}|$, but less for lower values of k .

In many application domains, it will be reasonable to assume that utility functions are k -additive with a relatively small value of k . Indeed, the larger a bundle of resources, the

¹Nisan (2000) proves a number of similar separation results for different types of bidding languages for combinatorial auctions and Coste-Marquis et al. (2004) do the same for a number of logic-based languages for expressing ordinal preferences.

more difficult would it be for an agent to estimate the additional benefit incurred by owning all the resources in that bundle *together* (i.e. beyond the benefit incurred by the relevant subsets). Our analysis shows that this *cognitive* argument in favour of the k -additive form is further supported by *computational* arguments (small values of k allow for a succinct representation in the k -additive form, but not necessarily in the bundle form). By the same argument, even when k is not very small, the utility function we obtain by disregarding the coefficients associated with the largest bundles will often serve as a good *approximation* to an agent's true preferences (the larger a bundle, the less likely that bundle will incur a significant benefit beyond the value derived from its subsets).

The bundle form corresponds to the so-called XOR-language for expressing bids in combinatorial auctions (Nisan 2000; Sandholm 2002). The potential benefits of exploiting the k -additive form in the context of combinatorial auctions, however, appears not to have been recognised until very recently (Conitzer et al. 2005). Still, there are also some interesting parallels between the k -additive form and the “Toolbox DNF” bidding language (Zinkevich et al. 2003). Indeed, the definitions of both languages seem almost identical. The crucial difference is that the ToolBox DNF is parametrised by the *number of coefficients*, while the k -additive form is parametrised by the *maximal cardinality of bundles with non-zero coefficients*. The ToolBox DNF will be useful in domains where there are a (relatively small) number of specific bundles that have cumulative value. The k -additive form will be useful in situations where no fixed set of valuable bundles can be specified *a priori*, but where cognitive or computational arguments can be given that support the assumption that no bundle exceeding a certain cardinality could possibly incur any additional synergetic value. Also note that, while the k -additive form naturally gives rise to a meaningful hierarchy of utility functions (from the 1-additive to the $|\mathcal{R}|$ -additive functions), this is not the case for the ToolBox DNF. The connections between our negotiation framework and combinatorial auctions will be further explored in Sect. 5.

4 Structural complexity of deals

Recall that Proposition 2 has shown that it is always possible to negotiate a socially optimal allocation of resources by means of rational deals involving only a single resource at a time whenever the utilities of all the agents involved are additive. Intuitively, we could have expected a similar result for k -additive utilities with $k \geq 2$ (i.e. a result that states that rational deals involving at most k resources at a time are sufficient to reach optimal allocations whenever all utility functions are k -additive). However, as we are going to show next, this turns out not to be the case. The deals required to reach allocations with maximal social welfare in the k -additive case are much more complex.

Proposition 6 (Insufficiency of restricted deals) *Even if all utility functions are k -additive for some $k \geq 2$, a deal involving the complete set of resources may be necessary to reach an allocation with maximal social welfare by means of a sequence of rational deals.*

Proof To prove the claim, we construct an example with 2-additive utility functions in which a deal involving all resources in \mathcal{R} is needed. Consider two agents sharing n resources $\mathcal{R} = \{r_1, r_2, \dots, r_n\}$, with the following 2-additive utility functions:

$$u_1 = 0,$$

$$u_2 = r_1 - r_1 \cdot r_2 - r_1 \cdot r_3 - r_1 \cdot r_4 - \dots - r_1 \cdot r_n.$$

Let A_{init} be the initial allocation describing which agent owns which resource before negotiation commences, and let A_{max} be the allocation maximising social welfare:

	A_{init}	A_{max}
Agent 1:	$\{r_1\}$	$\{r_2, r_3, \dots, r_n\}$
Agent 2:	$\{r_2, r_3, \dots, r_n\}$	$\{r_1\}$

Here, $sw(A_{init}) = 0$ and $sw(A_{max}) = 1$. In fact, the *only* allocation which has a social welfare greater than $sw(A_{init})$ is A_{max} . Recall that a deal increases social welfare if and only if it is a rational deal (Endriss et al. 2006). Thus, the only rational deal here is $\delta = (A_{init}, A_{max})$, which is a bilateral deal involving all n resources at the same time. \square

A possible objection to the example used in our proof may be that it is rather artificial. Utility functions that also have some additional properties, such as being monotonic,² besides being k -additive may be more relevant in practice. To show that the problem of requiring complex deals persists even when we make such additional assumptions, we give a further, similarly simple, example that demonstrates that also for k -additive functions that are monotonic, rational deals involving no more than k resources do not always suffice to negotiate socially optimal allocations. Consider the case of three agents and four resources with the following utility functions:

$$u_1 = 4 \cdot r_1 \cdot r_3, \quad u_2 = 3 \cdot r_1 \cdot r_2, \quad u_3 = 2 \cdot r_3 \cdot r_4.$$

Let A_{init} be the initial allocation and let A_{max} be the optimal allocation with maximal social welfare:

	A_{init}	A_{max}
Agent 1:	$\{r_1, r_3\}$	\emptyset
Agent 2:	$\{r_2, r_4\}$	$\{r_1, r_2\}$
Agent 3:	\emptyset	$\{r_3, r_4\}$

We get $sw(A_{init}) = 4$ and $sw(A_{max}) = 5$. Clearly, the only rational deal (i.e. the only deal increasing social welfare) is $\delta = (A_{init}, A_{max})$, which is a deal involving 3 (rather than just 2) resources at the same time.

In summary, our results show, differently from what one might have expected, that the restriction to utility functions that are k -additive for a given value of k does *not*, in general, reduce the complexity of deals required to reach a socially optimal allocation of resources in an agent society whose members follow a simple rational negotiation strategy.

To be able to use simple negotiation protocols efficiently, we need to make much stronger assumptions on the utility functions used by agents. In a recent paper (Chevaleyre et al. 2005), we prove a generalisation of Propositions 1 and 2 and show that rational deals involving at most k items each are sufficient for convergence to an optimal allocation whenever all utility functions are *additively separable* with respect to a common partition of \mathcal{R} (i.e. synergies across different parts of the partition are not possible and overall utility is defined

²A utility function is called monotonic if and only if the utility of a set of resources is never lower than the utility assigned to any of its subsets.

as the sum of utilities for the different sets in the partition Fishburn 1970), and each set in this partition has at most k elements.

5 Computational complexity results

In this section, we are going to analyse the *computational complexity* of the problem of finding an allocation that maximises social welfare, both with respect to the bundle form and with respect to the k -additive form of representing utility functions.³ In Sect. 3, we have already mentioned the connection between different representations of utility functions (in our case the bundle form and the k -additive form) in our distributed negotiation framework and different bidding languages in combinatorial auctions. In what follows, we explore a further connection between the two areas.

If we view the problem of finding an allocation with maximal social welfare as an algorithmic problem faced by a central authority (rather than as a problem of designing suitable negotiation mechanisms), then we can observe an immediate relation to the so-called *winner determination problem* in combinatorial auctions (Cramton et al. 2006; Rothkopf et al. 1998). In a combinatorial auction, bidders can put in bids for different *bundles* of items (rather than just single items). After all bids have been received, the auctioneer has to find an allocation for the items on auction amongst the bidders in a way that maximises his revenue. If we interpret the price offered for a particular bundle of items as the utility the agent in question assigns to that set, then maximising revenue (i.e. the sum of prices associated with winning bids) is equivalent to finding an allocation with maximal utilitarian social welfare. This equivalence holds, at least, in cases where the optimal allocation of items in an auction is such that *all* of the items on auction are in fact being sold (so-called *free disposal*).

Winner determination in combinatorial auctions is known to be NP-complete (Rothkopf et al. 1998).⁴ The quoted result applies to the case of the “standard” bidding language, which allows bidders to specify prices for particular bundles and makes the implicit assumption that they are prepared to obtain any number of disjoint bundles for which they have submitted a bid (Nisan 2000 calls this the “OR language”). Our languages for expressing utilities are more general than this (they can express a larger class of preference structures). The correspondence to combinatorial auctions suggests that the problem of finding an allocation with maximal social welfare is at least NP-hard. We can make this observation more precise by showing how our problem relates to well-known NP-complete “reference problems” (Garey and Johnson 1979; Ausiello et al. 1999). One such problem is SET PACKING. We use the schema of Ausiello et al. (1999) to define combinatorial problems:

SET PACKING

Instance: Collection \mathcal{C} of finite sets.

Solution: Collection of disjoint sets $\mathcal{C}' \subseteq \mathcal{C}$.

Measure: Cardinality of \mathcal{C}' .

³We assume familiarity with the basic concepts of complexity theory; the textbook by Papadimitriou (1994) provides an excellent introduction to the subject.

⁴More precisely, the *decision problem underlying the winner determination problem*, i.e. the problem of checking whether there exists an allocation that achieves at least a given minimal revenue K is NP-complete. The concept of NP-completeness applies to decision problems rather than optimisation problems (Ausiello et al. 1999). The winner determination problem is still NP-hard in the sense that solving it is at least as hard as solving any NP-complete decision problem.

The *optimisation problem* known as MAXIMUM SET PACKING is the problem of finding a solution C' for which the cardinality of C' is maximal. The underlying *decision problem* is the problem of answering the question whether there exists a solution C' for which the cardinality exceeds a given threshold K . This decision problem is known to be NP-complete in the size of the instance, i.e. with respect to the number of sets in \mathcal{C} (Garey and Johnson 1979).

Proposition 7 (Complexity wrt. bundle form) *The decision problem underlying the problem of finding an allocation with maximal social welfare with utilities represented in bundle form is NP-complete.*

Proof NP-membership follows from the fact that the conditions imposed on valid solutions can be checked in polynomial time. We prove NP-hardness by showing how to reduce SET PACKING to our problem. Given an instance \mathcal{C} of SET PACKING, consider the following negotiation problem: $\mathcal{R} = \bigcup_{C \in \mathcal{C}} C$; $\mathcal{A} = \mathcal{C} \cup \{0\}$; $u_C(R) = 1$ if $R = C$ and $u_C(R) = 0$ otherwise; and $u_0(R) = 0$ for all bundles R . That is, the elements of the sets in \mathcal{C} are the resources and there is one agent for every set in \mathcal{C} , as well as an additional agent called 0. Every agent values “its” bundle at 1 and every other bundle at 0. Agent 0 values all bundles at 0. Then for every allocation A there exists an allocation A' with at least the same social welfare that directly corresponds to a solution C' , i.e. each of the agents in \mathcal{C} either owns (only) its favourite bundle or no resources at all, and agent 0 owns all other resources. Hence, there exists an allocation A with $sw(A) > K$ if and only if there exists a solution C' with $|C'| > K$. \square

Dunne et al. (2005) have recently established a similar result. However, these authors prove NP-hardness with respect to the number of resources in the system rather than with respect to the combined size of the representations of utility functions (which will typically be significantly higher), i.e. our lower complexity bound is sharper than that of Dunne et al. On the other hand, the NP-membership result of Dunne et al. relies on the representation of utility functions as programs, which is more succinct than the bundle form, i.e. their upper complexity bound is sharper than the one reported here. Fargier et al. (2004) also prove a similar result. In their resource allocation framework agents can, by default, share individual resources, but if a particular resource can only be owned by one agent at a time, this can be specified by giving additional constraints. Crucially, all these results relate to the complexity of finding an optimal allocation by means of a *centralised* algorithm. For a discussion of the aspects of complexity that are relevant in a distributed negotiation setting, we refer to Endriss and Maudet (2005).

Our next aim is to establish the complexity of the same decision problem, but this time with respect to the k -additive form rather than the bundle form of representing utility functions. As the bundle form can be exponentially more succinct than the k -additive form, NP-hardness with respect to the former does not necessarily imply NP-hardness with respect to the latter. Nevertheless, as we are going to see, deciding whether there exists an allocation of resources with a utilitarian social welfare that exceeds a given threshold is also NP-complete. This time, we are going to use a reduction from another well-known combinatorial problem:

MAXIMUM 2-SATISFIABILITY

Instance: Set \mathcal{C} of clauses (disjunctions of literals) of length 2.

Solution: Satisfiable set $C' \subseteq \mathcal{C}$.

Measure: Cardinality of C' .

The decision problem of checking whether there exists a solution \mathcal{C}' for which the cardinality exceeds a given threshold K is known to be NP-complete (Garey and Johnson 1979). Note that this is the case even when the length of clauses is required to be just 2 (the closely related SAT problem, where we ask whether or not *all* clauses in \mathcal{C} are satisfiable, only becomes NP-hard for clauses with at least 3 literals).⁵

Proposition 8 (Complexity wrt. k -additive form) *For any $k \geq 2$, the decision problem underlying the problem of finding an allocation with maximal social welfare with utilities represented in k -additive form is NP-complete.*

Proof Again, NP-membership is straightforward. For the NP-hardness result it suffices to consider the case $k = 2$; the problem for $k > 2$ is at least as hard. We show NP-hardness for $k = 2$ by means of a reduction from MAXIMUM 2-SATISFIABILITY. Let \mathcal{C} be an instance of this problem. Now define \mathcal{R} as the set of propositional letters occurring in \mathcal{C} and let $\mathcal{A} = \{1, 2\}$. We now define the utility function u_1 of agent 1 using the 2-additive form. It is the sum of the following 2-additive terms, one for each clause in \mathcal{C} :

- p_i for any clause of the form $p_i \vee p_i$;
- $(1 - p_i)$ for any clause of the form $\neg p_i \vee \neg p_i$;
- $p_i + p_j - p_i \cdot p_j$ for any clause of the form $p_i \vee p_j$ with $i \neq j$;
- $p_i + (1 - p_j) - p_i \cdot (1 - p_j)$ for any clause of the form $p_i \vee \neg p_j$ with $i \neq j$;
- $(1 - p_i) + (1 - p_j) - (1 - p_i) \cdot (1 - p_j)$ for any clause of the form $\neg p_i \vee \neg p_j$ with $i \neq j$.

For agent 2, define $u_2(R) = 0$ for all bundles R . Every assignment of truth values to the propositional letters corresponds to a resource allocation in the following sense: agent 1 owns resource p iff p is set to *true*. Then the number of clauses satisfied by a given assignment is exactly the social welfare of the corresponding allocation. Hence, deciding whether there exists an allocation with a social welfare exceeding a given threshold K is at least as hard as MAXIMUM 2-SATISFIABILITY. \square

Our proof shows that we get NP-hardness even for $k = 2$ and scenarios with just two agents (with one agent using the trivial utility function mapping every bundle to 0). Other refinements of the NP-hardness result can be achieved by exploiting the special characteristics of the chosen NP-hard reference problem. Conitzer et al. (2005), for instance, show that NP-hardness persists even when each agent may only use two non-zero coefficients to represent its 2-additive utility function.

As a final complexity result, we are going to show that the problem of *verifying* that a given allocation is socially optimal is coNP-complete. This holds for both the bundle form and the k -additive form of representing utility functions and is a simple corollary to Propositions 7 and 8.

Corollary 1 (Verifying optimality) *The problem of verifying whether a given allocation has got maximal social welfare is coNP-complete (for both forms of representation of utility functions).*

⁵The reduction from MAXIMUM 2-SATISFIABILITY has been suggested to us by Jérôme Lang; an alternative proof, using a reduction from INDEPENDENT SET (Garey and Johnson 1979) together with a translation based on an idea from pseudo-boolean optimisation (Boros and Hammer 2002) for the case $k = 2$, may be found in Chevaleyre et al. (2004).

Proof Checking that an allocation A is *not* optimal involves computing $sw(A)$, which can be done in polynomial time, and then solving the decision problem “is there an allocation A' with $sw(A') > sw(A)$?”. The latter is NP-complete according to Proposition 7 (Proposition 8) for the bundle (k -additive) form. Hence, the complementary problem must be coNP-complete. \square

Related to this result, Dunne et al. (2005) have shown that the problem of checking whether a given allocation of resources is Pareto optimal is also coNP-complete.⁶ Related complexity results have also been obtained by Bouveret and Lang (2005), who study the computational complexity of deciding whether a given resource allocation scenario admits a solution that is both Pareto optimal and envy-free.⁷

What is the practical relevance of the connections between our negotiation framework and the combinatorial optimisation problems discussed in this section? In the proof of Proposition 8, for instance, we have reduced MAXIMUM 2-SATISFIABILITY to a very specific class of instances of the problem of finding a suitable allocation of resources. While this reduction has been useful to establish our NP-hardness result, it does not provide us with much helpful information on how to find an optimal allocation in practice. Here, the opposite direction, i.e. reductions *from* resource allocation problems *to* standard combinatorial optimisation problems may be more attractive. Obvious candidates would be the weighted variants of problems such as SET PACKING. Such a reduction would allow us to exploit existing algorithms, including highly optimised approximation algorithms (Ausiello et al. 1999), to find optimal (or near-optimal) allocations of resources. We should, however, stress that, of course, this would be a methodology for a centralised approach to finding optimal resource allocations. It is not immediately applicable to negotiation, which is a distributed process. Nevertheless, the techniques used to design optimisation and approximation algorithms may still inspire useful mechanisms for distributed resource allocation.

6 Conclusion

We have further analysed a framework for multiagent resource allocation previously studied by several authors (Sandholm 1998; Endriss et al. 2006; Dunne et al. 2005; Endriss and Maudet 2005). In particular, we have investigated scenarios where agents use k -additive utility functions to represent their preferences, which is possible whenever synergies between different resources are restricted to bundles of at most k items. We see the work presented in this paper as part of a wider research trend, which brings together ideas from different areas including microeconomics, operations research, decision theory, game theory, social choice, artificial intelligence, complexity theory, and algorithm design (Papadimitriou 2001; Lang 2005; Chevaleyre et al. 2006).

Our results presented in Sect. 4 show that, despite the positive expectations raised by the result on negotiation in additive domains (Proposition 2), the structural complexity of the negotiation protocol required to agree on a socially optimal allocation does not necessarily decrease for problems with k -additive utility functions when k gets smaller (as long as

⁶An allocation is called *Pareto optimal* if and only if there is no other allocation that would be better for at least one of the agents without being worse for any of the others. For further results on negotiating Pareto optimal allocations we refer to Endriss et al. (2006).

⁷An allocation A is called *envy-free* if and only if no agent would rather have the bundle allocated to one of the other agents (Brams and Taylor 1996); that is, if and only if $u_i(A(i)) \geq u_i(A(j))$ for all agents $i, j \in \mathcal{A}$.

$k > 1$). On the other hand, as we have seen in Sect. 3, representing utility functions in the k -additive form rather than the bundle form can be significantly more succinct, particularly in cases where a representation with a small value for k is possible.

We have also explored connections to well-known combinatorial optimisation problems, which has allowed us to establish complexity results for the problem of finding a socially optimal allocation with respect to different representations of utility functions (Sect. 5). In this context, we have also briefly discussed the relation of our negotiation framework to combinatorial auctions for different kinds of bidding languages. While our negotiation framework is clearly *not* an auction (it is, for instance, not concerned with the aspect of agreeing on the price for a set of items), the abstract “centralised” problem of finding a socially optimal allocation (which is not itself a problem faced by the agents participating in a negotiation process) directly corresponds to the winner determination problem in combinatorial auctions. Under this view, the languages used to represent utility functions correspond to bidding languages for such auctions. As regards our complexity results, it is important to stress that the high complexity of the distributed negotiation framework does not, at least not necessarily, mean that it cannot be usefully applied in practice. This view is supported by the fact that, in recent years, several algorithms for winner determination in combinatorial auctions (a problem of comparable complexity to the problems arising here) have been proposed and applied successfully (Rothkopf et al. 1998; Fujishima et al. 1999; Sandholm 2002).

One important aspect of modelling preferences that we have not discussed in this paper concerns the *elicitation* of preferences (Keeney and Raiffa 1993; Boutilier et al. 1997; Gonzales and Perny 2004). The very natural representation of utility functions in the k -additive form suggests that it would also be particularly suited to elicitation, certainly from a cognitive point of view. Note that, if elicitation is understood in purely computational terms (“how many queries of a certain type do we need to ask to be able to fully specify a preference structure?”), then elicitation complexity is closely related to succinctness (see, for instance, the discussion of elicitation complexity in the combinatorial auction literature Sandholm and Boutilier 2006). For example, the query type corresponding to the bundle form is the simple “value query” (Sandholm and Boutilier 2006; Zinkevich et al. 2003) which asks for the utility associated with a given bundle, while a query language corresponding to the k -additive form would allow us to ask for the coefficient α^R associated with a given bundle of resources R .

Open Access This article is distributed under the terms of the Creative Commons Attribution Noncommercial License which permits any noncommercial use, distribution, and reproduction in any medium, provided the original author(s) and source are credited.

References

- Arrow, K. J., Sen, A. K., & Suzumura, K. (Eds.) (2002). *Handbook of social choice and welfare* (Vol. 1). Amsterdam: North-Holland.
- Ausiello, G., Crescenzi, P., Gambosi, G., Kann, V., Marchetti-Spaccamela, A., & Protasi, M. (1999). *Complexity and approximation: combinatorial optimization problems and their approximability properties*. Berlin: Springer.
- Boros, E., & Hammer, P. L. (2002). Pseudo-boolean optimization. *Discrete Applied Mathematics*, 123(1–3), 155–225.
- Boutilier, C., Brafman, R. I., Geib, C., & Poole, D. (1997). A constraint-based approach to preference elicitation and decision making. In *Proc. AAAI spring symposium on qualitative decision theory*.
- Bouveret, S., & Lang, J. (2005). Efficiency and envy-freeness in fair division of indivisible goods: logical representation and complexity. In *Proc. 19th international joint conference on artificial intelligence (IJCAI-2005)*. San Mateo: Morgan Kaufmann.

- Brams, S. J., & Taylor, A. D. (1996). *Fair division: from cake-cutting to dispute resolution*. Cambridge: Cambridge University Press.
- Chevalere, Y., Endriss, U., Estivé, S., & Maudet, N. (2004). Multiagent resource allocation with k -additive utility functions. In *Proc. DIMACS-LAMSADE workshop on computer science and decision theory*. Annales du LAMSADE 3.
- Chevalere, Y., Endriss, U., Lang, J., & Maudet, N. (2005). Negotiating over small bundles of resources. In *Proc. 4th international joint conference on autonomous agents and multiagent systems (AAMAS-2005)*. New York: ACM Press.
- Chevalere, Y., Dunne, P. E., Endriss, U., Lang, J., Lemaître, M., Maudet, N., Padget, J., Phelps, S., Rodríguez-Aguilar, J. A., & Sousa, P. (2006). Issues in multiagent resource allocation. *Informatica*, 30, 3–31.
- Conitzer, V., Sandholm, T. W., & Santi, P. (2005). Combinatorial auctions with k -wise dependent valuations. In *Proc. 20th national conference on artificial intelligence (AAAI-05)*. Menlo Park: AAAI Press.
- Coste-Marquis, S., Lang, J., Liberatore, P., & Marquis, P. (2004). Expressive power and succinctness of propositional languages for preference representation. In *Proc. of the 9th international conference on principles of knowledge representation and reasoning (KR-2004)*. Menlo Park: AAAI Press.
- Cramton, P., Shoham, Y., & Steinberg, R. (Eds.) (2006). *Combinatorial auctions*. Cambridge: MIT Press.
- Dunne, P. E., Wooldridge, M., & Laurence, M. (2005). The complexity of contract negotiation. *Artificial Intelligence*, 164(1–2), 23–46.
- Endriss, U., & Maudet, N. (2005). On the communication complexity of multilateral trading: extended report. *Journal of Autonomous Agents and Multiagent Systems*, 11(1), 91–107.
- Endriss, U., Maudet, N., Sadri, F., & Toni, F. (2006). Negotiating socially optimal allocations of resources. *Journal of Artificial Intelligence Research*, 25, 315–348.
- Fargier, H., Lang, J., Lemaître, M., & Verfaillie, G. (2004). Partage équitable de ressources communes: (2) Éléments de complexité et d'algorithmique. *Technique et Science Informatique*, 23(9), 1219–1238.
- Fishburn, P. C. (1970). *Utility theory for decision making*. New York: Wiley.
- Fujishima, Y., Leyton-Brown, K., & Shoham, Y. (1999). Taming the computational complexity of combinatorial auctions: optimal and approximate approaches. In *Proc. 16th international joint conference on artificial intelligence (IJCAI-1999)*. San Mateo: Morgan Kaufman.
- Garey, M. R., & Johnson, D. S. (1979). *Computers and intractability: a guide to the theory of NP-completeness*. New York: Freeman.
- Gonzales, C., & Perny, P. (2004). GAI networks for utility elicitation. In *Proc. of the 9th international conference on principles of knowledge representation and reasoning (KR-2004)*. Menlo Park: AAAI Press.
- Grabisch, M. (1997). k -order additive discrete fuzzy measures and their representation. *Fuzzy Sets and Systems*, 92, 167–189.
- Keeney, R. L., & Raiffa, H. (1993). *Decisions with multiple objectives: preferences and value trade-offs*. Cambridge: Cambridge University Press.
- Lang, J. (2005). Some representation and computational issues in social choice. In *Proc. 8th European conference on symbolic and quantitative approaches to reasoning with uncertainty (ECSQARU-2005)*. Berlin: Springer.
- Moulin, H. (1988). *Axioms of cooperative decision making*. Cambridge: Cambridge University Press.
- Nisan, N. (2000). Bidding and allocation in combinatorial auctions. In *Proc. ACM conference on electronic commerce (EC-2000)*. New York: ACM Press.
- Papadimitriou, C. H. (1994). *Computational complexity*. Reading: Addison-Wesley.
- Papadimitriou, C. H. (2001). Algorithms, games, and the Internet. In *Proc. 33rd annual ACM symposium on theory of computing (STOC-2001)*. New York: ACM Press.
- Rota, G. C. (1964). On the foundations of combinatorial theory I: Theory of Möbius functions. *Zeitschrift für Wahrscheinlichkeitstheorie und Verwandte Gebiete*, 2(4), 340–368.
- Rothkopf, M. H., Pekeč, A., & Harstad, R. M. (1998). Computationally manageable combinatorial auctions. *Management Science*, 44(8), 1131–1147.
- Sandholm, T. W. (1998). Contract types for satisficing task allocation: I. Theoretical results. In *Proc. AAAI spring symposium: satisficing models*.
- Sandholm, T. W. (2002). Algorithm for optimal winner determination in combinatorial auctions. *Artificial Intelligence*, 135(1–2), 1–54.
- Sandholm, T. W., & Boutilier, C. (2006). Preference elicitation in combinatorial auctions. In P. Cramton et al. (Eds.), *Combinatorial auctions*. New York: MIT Press.
- Wooldridge, M. (2002). *An introduction to multiagent systems*. New York: Wiley.
- Zinkevich, M. A., Blum, A., & Sandholm, T. W. (2003). On polynomial-time preference elicitation with value queries. In *Proc. ACM conference on electronic commerce (EC-2003)*. New York: ACM Press.