Evaluating the Benefits of Generalized Additive Representation in a Multiattribute Auction Setting

Yagil Engel and **Michael P. Wellman**
University of Michigan, Computer Science & Engineering
2260 Hayward St, Ann Arbor, MI 48109-2121, USA
{yagil,wellman}@umich.edu

Abstract

We compare the economic efficiency achieved by multiattribute auctions that use an accurate GAI modeling of preferences, with a multiattribute auction that is limited to an additive representation. We draw random GAI-structured utility functions with various internal structures, generate additive functions that approximate the GAI utility, and compare the performance of the auctions using the two representations. We draw general preference modeling and methodological conclusions from both the process and the results.

Introduction

The *Generalized Additive Independence* (GAI) representation is gaining popularity within AI researchers as a convenient and effective modeling of multiattribute preference structure (Boutilier, Bacchus, & Brafman 2001; Gonzales & Perny 2004; Braziunas & Boutilier 2005). Its main advantage is that it covers the spectrum between the ever-popular *additive representation*, which assumes that preferences over any subset of attributes are not affected by the values of other attributes, and the full joint utility that makes no assumption of independence between attributes. It allows the modeler to pick the right tradeoff between the complexity of representation and its accuracy. Yet, this is a fairly simple representation, in contrast to other traditional representations that require more complex forms and involve additional costs (e.g. Keeney and Raiffa (1976)).

In practice, however, the additive representation is still prominent, at least in the electronic commerce related application we consider. A common explanation is that the additive representation is considered, even if not exactly right, a "good enough" approximation. We argue that the field of preference handling does not provide sufficient tools to refute this kind of a claim. In particular, there is a need to quantify the benefits of a GAI representation over the additive simplification.

In this work we set out to provide such a methodological quantification, and systematically compare the performance of an accurate GAI modeling with its additive approximation, via simulations and using randomly generated data. The problem space we use is that of multiattribute auctions, and we experiment with a type of auction we recently proposed (Engel & Wellman 2007), that is capable of taking advantage of a known GAI structure. Though the specific results are relevant to the specific auction mechanism we tested, we believe that the methodology, the simulation framework and the spirit of the results can be applied to other applications.

There are several challenges that needed to be addressed in performing such simulations. The first one is how to generate semantically sound GAI structured utility functions randomly. There are two layers to this problem: 1) the sampling must yield functions that are internally consistent. 2) The local GAI functions may be completely random, but a more interesting and a more credible experimentation would consider that these functions may have some internal structure, which may imitate true utility better. A second challenge is how to create the baseline additive representation—what should we exactly compare the performance of the GAI-based application with.

We begin by providing background on multiattribute auctions, and GAI auctions in particular. Next, we discuss our goals in the simulations, how the challenges above were addressed, and some of the most relevant results.

GAI auctions

A *multiattribute auction* is a market mechanism that extends traditional, price-only mechanisms by facilitating the negotiation over a set of predefined attributes representing various non-price aspects of the deal. For example, a procurement department of a company may use a multiattribute auction to select a supplier of hard drives. Supplier offers may be evaluated not only over the price they offer, but also over various qualitative attributes such as volume, RPM, access time, latency, transfer rate, and so on. In addition, different suppliers may offer different contract conditions such as warranty, delivery time, and service. Procurement is the common application of multiattribute auctions, hence we limit our discussion to the case of one buyer and multiple sellers.

As mentioned, most multiattribute mechanisms, in recent literature and in practice, assume the simple additive multiattribute form (Bichler 2001; Parkes & Kalagnanam 2005). Another common "shortcut" used in multiattribute auctions literature (but rarely in practice) is based on the *second score* approach introduced by Che (1993). If the buyer initially reveals a full evaluation function over the domain (also called a *scoring function*), each seller can compare it to its own cost function. The competition can then be reduced to a single dimensional one, where sellers compete over the surplus, or score, they can provide the buyer. The downside is, of course, the unrealistic information revelation requirement imposed on the buyer.

Let Θ denote a space of possible outcomes, or possible *configurations* of a particular item. Let $S = \{a_1, \ldots, a_m\}$ be the set of attributes (such as mentioned above) describing Θ. We assume that each trader has a *willingness-to-pay* function (for the buyer) and *willingness-to-accept*, or cost function for a seller. We describe these functions using the framework of *measurable value function (MVF)* (Dyer & Sarin 1979), which is a utility function $u(S)$ that is shown to accommodate willingess-to-pay and cost (Engel & Wellman 2007). The GAI decomposition, defined below, was originally introduced by Fishburn (1967).[1]

Definition 1. *Let S be a set of attributes, and let $I_1, \ldots, I_g \subseteq S$ such that $\bigcup_{i=1}^{g} I_i = S$. $I_1, \ldots, I_g$ are called* generalized additive independent *(GAI) if there exist functions $f_1, \ldots, f_g$ such that*

$$u(S) = \sum_{r=1}^{g} f_r(I_r). \quad (1)$$

We call each I_r a GAI *element*, and any assignment to I_r a *sub-configuration*.

A *GAI network* is defined as the graph that contains a node for each GAI element and an edge for each pair of elements that intersect (Gonzales & Perny 2004). GAI auctions are restricted to the case that the GAI network is a tree or a forest. This requirement does not reduce generality, but may result in a higher dimensionality (larger sized GAI elements). When the subsets $I_1, \ldots, I_g$ are disjoint, the GAI representation degeneralizes to an *additive representation*. In additive auctions, each attribute is usually considered on its own, that is $u(S) = \sum_{i=1}^{m} f_i(a_i)$.

GAI auctions address both the revelation problem and the preference modeling problems mentioned above. It reflects the buyer's partial valuation through a price structure, that contains a price for any sub-configuration of the buyer's GAI structure. In each round, all active sellers are bidding on full configurations, and their bids are stored by their projections on the GAI elements. The auction selects the *buyer-optimal set*: a set of configurations which are approximately optimal for the buyer, based on a well-defined notion of approximation. Then the price is reduced by δ for any sub-configuration that is a part of some seller's bid, and is not a part of a configuration in the buyer-optimal set. This guarantees that at some point each seller converges to bid on an approximately efficient configuration, that is a configuration that maximizes the surplus between her cost and the buyer's willingness-to-pay, at which point the auction becomes a single dimensional competition on the surplus the sellers are willing to transfer to the buyer, through a discount. This achieves the same effect as achieved by revealing the buyer's scoring function, but here it is only a fraction of the buyer's information that is revealed.

[1] We use a somewhat non standard definition in order to simplify the presentation.

GAI auctions provide theoretical guarantees for the efficiency it achieves. We define the parameter e to be the *number of edges in the GAI network*, that is a measurement of the connectivity of the GAI elements. Further, let ϵ be the maximum price change per configuration at each round, that is, $\epsilon = \delta g$. The following result is limited to the case in which traders bid truthfully, meaning they use the strategy known as *straightforward bidding (SB)*. This is however a reasonable assumption, because it is also shown that straightforward bidding is an (approximately) ex-post Nash Equilibrium for sellers in GAI auctions.

Theorem 1. *(Engel & Wellman 2007) Given a truthful buyer and SB sellers, GAI auction is $(e+2)\epsilon$-efficient: the surplus of the final allocation is within $(e+2)\epsilon$ of the maximum.*

Simulations Design

The main idea behind the GAI auctions is to improve efficiency over the auctions that assume the additive representation, when the preferences are in fact not additive. However, the theoretical efficiency guarantee of GAI auctions depends on e, the connectivity of the GAI network. This suggests a tradeoff: more accurate modeling improves efficiency with respect to the true utility, but may cause loss of efficiency due to higher connectivity. Therefore, the obvious goal is to test whether GAI auctions are more efficient than additive auctions, given that the preferences are not additive. We assume that the buyer's preferences have some GAI structure, and compare the performance of the GAI auctions that model this structure, with the performance of an auction that is restricted to an additive representation. As the *additive approximating* auction (AP), we use an instance of GAI auction in which the GAI structure is additive. This auction is in fact very similar to auction AD, a previously suggested auction design that relies on an additive representation (Parkes & Kalagnanam 2005). To the best of our knowledge, no other welfare maximizing multiattribute auctions has been suggested for additive preferences, beside those that require full revelation of the buyer's utility.

GAI Random Utility

We generate random utility functions to represent the buyer's value function and the sellers' cost functions. In order to imitate realistic, GAI structured utility functions, we first describe previous results regarding the relationship between the functions f_r of Definition 1, and local utility functions over the GAI elements. The following representation was proven by Fishburn (1967), to hold for the *von Neumann-Morgenstern* utility functions, that is utility functions that represent (in addition to ordinal preferences), a preference order over probability distributions over the outcomes. We previously adapted this representation to the framework of measurable value functions (Engel & Wellman 2007).

Let $(a_1^0, \ldots, a_m^0)$ be a predefined vector called the *reference outcome*. For any $I \subseteq A$, the function $u([I])$ stands for the projection of $u(S)$ to I where the rest of the attributes are fixed at their reference levels.

Theorem 2. *Let $\{I_1, \ldots, I_g\}$ be a GAI decomposition. Then the functional decomposition can be defined as*

$$f_1 = u([I_1]), \text{ and for } r = 2, \ldots, g \quad (2)$$

$$f_r = u([I_r]) + \sum_{k=1}^{r-1}(-1)^k \sum_{1\le i_1<\cdots<i_k<r} u([\bigcap_{s=1}^{k} I_{i_s} \cap I_r])$$

Braziunas and Boutilier (2005) establish that, similarly to the additive utility, GAI utility can be represented using locally normalized functions, weighted by scaling constants λ_r. First, let $\bar{u}_r(I_r)$ denote a utility function, over I_r, given that any attribute $a \in S \backslash I_r$ is fixed to his reference level a_0.[2] $\bar{u}_r(I_r)$ is normalized to $[0, 1]$. Next, let $\bar{f}_r(I_r)$ be defined using the formula (2), with $u([I_r])$ replaced with $\bar{u}_r(I_r)$. Then

$$\bar{u}(S) = \sum_{r=1}^{g} \lambda_r \bar{f}_r(I_r). \quad (3)$$

Note that (2) is much simplified for GAI trees, because each element I_r intersect with at most one preceding element, its parent $I_{p(r)}$:

$$\bar{f}_r = \bar{u}_r(I_r) - \bar{u}_r([I_{i_r} \cap I_{p(r)}]) \quad (4)$$

We refer to the functions $\bar{u}_r([I_r])$ as *subutility functions*. This representation lets us draw random GAI functions, for a given GAI-tree structure, using the following steps:

1. Draw random subutility functions $\bar{u}_r(I_r), r = 1, \ldots, g$ in the range [0,1].
2. Compute $\bar{f}_r(\cdot), r = 1, \ldots, g$ using (4).
3. Draw random scaling constants and compute $\bar{u}(S)$ by (3).

The scaling constants represent the importance that the decision maker gives the specific GAI element in the overall decision. This procedure results in utilities that are normalized in $[0, 1]$. To accommodate means and variances of different agents, we then scale $\bar{u}(\cdot)$ to $u(\cdot)$ in the desired range.

Structured Subutility

A subutility function in the model above may represent any valuation over the subspace. However, in practice we may often find additional structure within each GAI element. In this section we discuss structures which we consider most typical and generally applicable. We later experiment with these structured subutility functions, as well as with completely random ones. To discuss potential structures, we need the following definitions. The definitions refer to both utilities and subutilities, hence we refer to a set of attributes S'. For $Y \subset S'$, we use $\overline{Y}$ to denote $S' \setminus Y$.

Definition 2. *Outcome Y' is conditionally preferred to outcome Y'' given $\overline{Y}'$, if $Y'\overline{Y}' \succeq Y''\overline{Y}'$. We denote the conditional preference order over Y given $\overline{Y}'$ by $\succeq_{\overline{Y}'}$.*

Definition 3. *Y is* Preferential Independent (PI) *of $\overline{Y}$ if $\succeq_{\overline{Y}'}$ does not depend on the value chosen for $\overline{Y}'$.*

[2] Braziunas and Boutilier show that only a relevant subset of the attributes needs to be considered as fixed. This is useful for elicitation, but not necessary here.

Typical purchase and sale decisions exhibit what we call *first-order preferential independence (FOPI)*, under which most or all single attributes have a natural ordering of quality, rendering each attribute by itself PI of the rest. For example, in a hard drive procurement setting, the buyer always prefers more memory, higher RPM, longer warranty, and so on. Let $D(x)$ denote the (discrete) domain of an attribute x. To implement FOPI, we let the integer values of each attribute represent its quality. For example, if a belongs to some GAI element $I_r = \{a, b\}$, we make sure that $\bar{u}_r(a_i, b') \ge \bar{u}_r([a_j, b'])$ for any $a_i > a_j,\ a_i, a_j \in D(a)$, and any $b' \in D(b)$. This must of course hold in any GAI element that includes a, and for any attribute a that is (first-order) preferential independent. We enforce the condition after all the values for that GAI element have been drawn, by a special purpose sorting procedure, that is applied between steps 1 and 2 above.

The FOPI condition makes the random utility function more realistic, and in particular more appropriate to our target application. Once attributes exhibit FOPI, the dependencies between different attributes are likely to be framed as *complements* or *substitutes*. Intuitively, two attributes a and b are complements if the utility of improving both is higher than the sum of utilities of improving a and b separately.

Definition 4. *Let $u(\cdot)$ be a measurable value function over S'. Let $a, b \in S'$, and $Z = S' \setminus \{a, b\}$, and assume that a and b are each FOPI of the rest of the attributes. a and b are called* complements *if for any $a_i > a_{\hat{i}}$ ($a_i, a_{\hat{i}} \in D(a)$) and any $b_j > b_{\hat{j}}$ ($b_j, b_{\hat{j}} \in D(b)$), and any $Z' \in D(Z)$ ($D(Z)$ stands for the joint domain of Z),*

$$u(a^i, b^j, Z') - u(a^{\hat{i}}, b^{\hat{j}}, Z') >$$
$$u(a^i, b^{\hat{j}}, Z') - u(a^{\hat{i}}, b^{\hat{j}}, Z') + u(a^{\hat{i}}, b^j, Z') - u(a^{\hat{i}}, b^{\hat{j}}, Z').$$

a and b are substitutes if the inequality sign is (always) the other way round.

This relationship between attributes is ruled out under an additive utility function. However, it may be admitted by the weaker *utility independence* (UI) condition. We omit the formal definition for UI, which is the extension of PI to *preferences over lotteries* (Keeney & Raiffa 1976), or *preferences differences* (i.e. differences between utility values), in the MVF framework (Dyer & Sarin 1979). Intuitively, in the MVF framework, UI($Y, \overline{Y}$) is implied by the following condition: Y is PI, and the order over the preferences differences over the values of Y is also invariant to the value of $\overline{Y}$. Y and $\overline{Y}$ can still be (for example) complements: the preference differences over Y may all be magnified for a better value of $\overline{Y}$, as long as the order over them does not change.

Definition 5. *A set of attributes S' is* mutually utility independent *(MUI) if every subset $X \subset S'$ is UI of $S' \setminus X$.*

MUI is a very demanding condition, and imposes far more structure then FOPI. It is however, still more flexible than the additive form. A set S' of MUI attributes may still include complements or substitutes, but only if the same type of condition, with the same "strength", occurs for every two attributes in S'. Therefore, assuming MUI allows us to use a single variable to control the level of complementarity or substitutivity within each GAI element, as follows. If a set S' is MUI, it can be represented using single-dimensional subutility function, and $j + 1$ constants (where $|S'| = j$). The first j constants, $k_1, \ldots, k_j$, (roughly) determine the weights of the attributes. The last constant, k, is computable from $k_1, \ldots, k_j$ (Keeney & Raiffa 1976). k, in turn, can be regarded as a measurement of substitutivity or complementarity: if $-1 < k < 0$, the attributes of S' are substitutes, if $k > 0$ they are complements, and if $k = 0$ they are additively independent. The more extreme the value of k is within the range (the further away it is from zero), the stronger the condition is. As this is not the focus of the paper, we refer to Keeney and Raiffa (1976) for the technical details of the MUI representation, and we omit the formal result (and its proof) that establishes the connection between k and Definition 5.[3]

In an elicitation procedure, one would normally extract the j scaling constants from a user, and then compute k based on that (Keeney & Raiffa 1976). For our purposes, we first determine k according to the relationship we wish to impose on the attributes, and then draw MUI scaling constants that are consistent with this value. More explicitly, we draw random scaling constants, and then iteratively modify all the constants, until a set of constants is found that is consistent with k. The next step is to compute $\bar{u}_r(I_r)$ according to the MUI formula (Keeney & Raiffa 1976). $\bar{u}_r(I_r)$ (for all r) are in the range $[0, 1]$, hence at this point we can proceed with steps 2 and 3 above. Note that in this procedure we use several distinct sets of scaling constants: the g constants used in step 3 scale the different GAI elements, whereas the MUI constants, per GAI element, scale the attributes *within* the element.

Additive Approximation

Our goal is to test whether using the GAI price structure improves the efficiency of a multiattribute auction in comparison to the previously suggested additive price structure. To do that, we should test the performance of the additive price structure given that traders' preferences are not additive, that is they exhibit some GAI structure. In GAI auctions, the price structure needs only to model the buyer's preferences. If a seller bids on a single configuration in every round, he is not exposed to undesired combinations of values. The problem is therefore how to select the approximately buyer-optimal sets of configurations, given that the buyer's preferences are not additive. The approach we have taken is to come up with an additive function that approximates the buyer's true utility function, and use it throughout the auction. We do not rule out the possibility that there are better strategies, however any such strategy must be consistent: that is, whenever a point is selected as (approximately) optimal, it must remain optimal with respect to the buyer's "revealed preferences" throughout the auction.

A natural approach to generate a linear approximation to an arbitrary function is to use linear regression. We define an indicator variable x_{i_j} for any $a_{i_j} \in D(a_i)$, and consider any value of an assignment as a data point. For example, the assignment $a_{1_{j(1)}} \ldots a_{m_{j(m)}}$ creates the following data point:

$$\sum_{i=1}^{m} \sum_{a_{i_j} \in D(a_i)} c_{i_j} x_{i_j} = u(a_{1_{j(1)}} \ldots a_{m_{j(m)}}),$$

in which the value of the variable x_{i_j} is 1 if $j = j(i)$ and 0 otherwise. The coefficients c_{i_j} represent the values to be used as $f_i(a_{i_j})$.

When the problem includes more than just a few attributes, regression that uses all of the data points is not tractable. We recall that the function has a compact representation, through its GAI structure - it can be fully described using a relatively small number of data points. We tried regression using a separate set of data points for each GAI element. We found that this method does not yield optimal results, meaning that auction AP achieves lower efficiency compared to auction AP that uses regression over the full joint utility. The reason is that the regression through the compact representation minimizes the error per each sub-configuration, and that may result in larger errors for full configurations. Fortunately, we also found (at least for the sizes of problems we tested) that we can use a small random sample of data points from the joint utility (roughly the same number of points required by the compact representation), and that yields an approximation that does as well as an approximation that uses all of the data points of the joint utility.

It is important to note that this method of comparison is overestimating the quality of an additive approximation. The reason is that typically we will not have the accurate utility function available to us when we generate the approximation. The extraction, or elicitation of the utility function is usually the most serious bottleneck of a multiattribute mechanism. Therefore, the major reason to use an additive approximation is to perform a shorter elicitation. Hence in practice we will try to obtain the additive function directly, rather than obtain the full utility and then approximate it. The result of such process is somewhat unpredictable, because the elicitation queries may not be coherent: if the willingness to pay for a_1 depends on the value of b, then what is the willingness to pay for a_1 when we do not know b? We therefore consider this method of approximation a "best case" scenario of what we might experience in practice.

Efficiency Analysis

All of our efficiency results are measured in terms of percentage of the maximal possible surplus, that is the surplus yielded by the optimal seller-configuration pair. The efficiency of the auctions depends on many factors, such as: d, the size of the domains of the attributes (to simplify, we use the same domain size for all the attributes). N, The number of sellers participating. ϵ (or δ), the amount of price decrement, which allows to tradeoff efficiency with the number of rounds required. Efficiency also depends on the distribution from which utility functions are drawn. In particular, on the

[3] An intuition for this result, limited to the specific case of MUI between two attributes, is also given by Keeney and Raiffa (1976).

differences between the means (μ) of the different traders, and the variance (σ) of each.

However, we are mostly interested in measuring efficiency with respect to the GAI structure. As shown in the theoretical results, the structure affects efficiency through the factor e. Further, the size of the GAI elements is expected to affect to performance of AP. We found that the size of the largest GAI element (denoted by ξ) is particularly crucial for AP. To isolate these factors that we care about, we first describe how the results vary according to the choices of the "side factors": the buyer's mean, the sellers' mean, σ, d, and N, for several fixed GAI structures with fully random subutility functions. This allows us to justify the parameter values we used for the rest of the simulations.

We expect seller's costs to be generally lower than buyer's valuations, otherwise there is no potential surplus. We arbitrarily selected the buyer's mean to be 600, and varied the general sellers' mean from 700 down to 300. Normally, different sellers have different cost levels, so we selected a mean for each seller from an interval of size 2σ around the general sellers' mean. We experimented with several values (between 50 and 250) for σ, for both the buyer and the sellers. We found that the choice of these parameters does have a serious effect on the efficiency, mostly that of AP. The difference in the means has similar effect on both types (auctions are more efficient when the difference is larger), and we picked the value of 500 to the sellers to reflect costs that are reasonably lower than the buyer's valuation. We also noticed that the smaller the variance is, the better an additive approximation performs. We postulate that it is simply easier to approximate the function when "there is less to lose" by being wrong. We picked the value $\sigma = 200$, to make sure the problem is sufficiently challenging.

Next, we varied d, between two to ten, for all the attributes. As expected, the larger the domain is, the more challenging the problem is for the additive approximation. Perhaps less expected, is the fact that the size of the domain seems to have no effect on the GAI auctions. We used domain sizes of three to five for the rest of the simulations. As for number of sellers, we varied this parameter from 2 to 40, and did not find a significant effect on either auction types. We used five sellers for the rest of the simulations.

Efficiency and GAI Structure

In the next experiment we used a roughly fixed GAI structure, with six elements and $e = 5$ (that is, a tree rather than a forest), and $\delta = 4$. We increased the number of attributes, by increasing the size of each element. Figure 1 shows the efficiency obtained given ξ, the size of the largest GAI element. As expected, the size of the GAI elements has negligible, or no effect on the efficiency of GAI auctions. It has a dramatic effect on the efficiency of AP. When $\xi = 1$, the decomposition is in fact additive and hence AP performs optimally. The performance then deteriorates when ξ increases.

We performed the same test when using utility in which all attributes are FOPI. Clearly, given FOPI, the additive approximation is much more efficient compared to random utilities. Somewhat surprisingly, the GAI auctions are slightly less efficient given this preference structure, again compared to random utility. Nevertheless, the additive approximation achieves lower efficiency compared to the accurate preference modeling, with differences that pass the statistical significance test ($P < 0.01$), for $\xi \geq 4$. Moreover, in practice FOPI may apply just to a subset of the attributes. We also note that the performance of GAI auctions can always be improved using a smaller value of ϵ and δ, whereas it hardly improves performance of AP. With $\delta = 2$, a statistically significant difference (with the same confidence level) is already detected for $\xi \geq 2$. We used $\delta = 2$ hereafter.

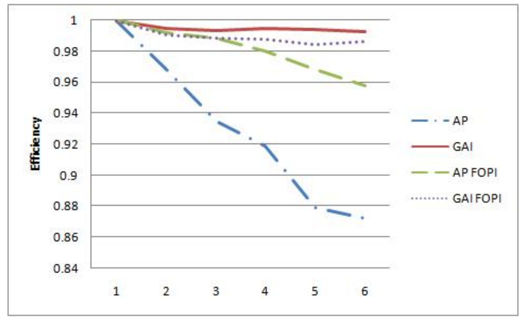

Figure 1: Efficiency vs. the size of largest GAI element.

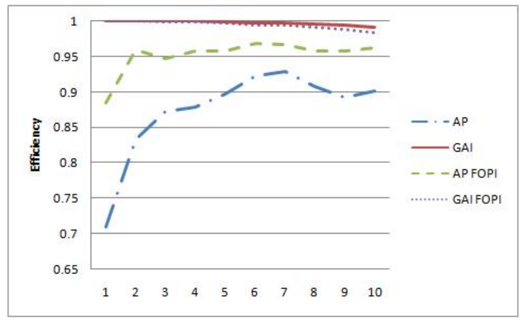

Figure 2: Efficiency vs. of the number of GAI elements.

The next experiment measures efficiency as a function of e, for a given size of the GAI elements. We did not test GAI forests, only trees, so e is equivalent to the number of GAI elements minus one. The simulation started with a structure of a single element of size five, to which we added another element in each trial, leading to a ten-elements structure.[4] On a single element, the GAI auction is similar to NLD (Parkes & Kalagnanam 2005), which is an auction that assigns a price to every point in the joint domain, being agnostic of any preference structure. Since $e = 0$, the efficiency of GSI is close to perfect. This structure is on the other extreme compared to an additive representation, therefore the performance of AP is particularly inferior (only 70% efficient).

With more GAI elements, the efficiency of GAI auctions declines at a very slow pace. The theoretical guarantee $(e+2)\epsilon$, is based on the worst case, in which a configuration chosen from the approximately buyer-optimal set consists of

[4]We did not find the particular tree structure to be influential on the results; the final structure used in the reported results has a maximum of three children per node.

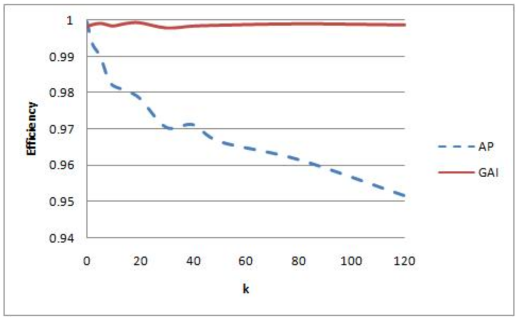

Figure 3: Efficiency as a function of $k \geq 0$.

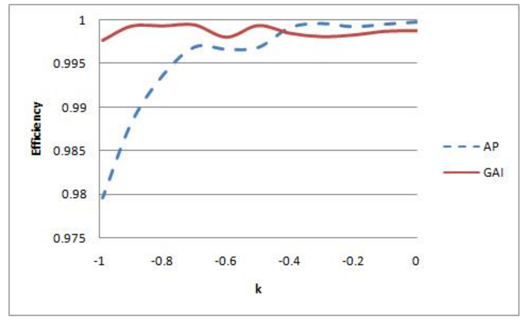

Figure 4: Efficiency as a function of $k \leq 0$.

sub-configurations which are all at the largest possible distance from optimum. In practice, the loss is closer to $e\delta$—a much smaller error. The performance of AP improves when the number of elements grows while their maximal and average sizes are fixed. The intuitive reason is that changing the structure that way takes it closer to an additive representation. Under FOPI, we see a similar phenomena as before. However, GAI FOPI outperforms AP FOPI even for ten elements, with a statistically significant difference.

Figures 3 and 4 show efficiency as a function of k, for complements and substitutes, respectively. We used a fixed GAI structure with four elements, the largest of which has four attributes, and imposed the same k on all the elements. As expected, the stronger the complementarity is between the attributes, the lower the efficiency of AP, whereas this relationship does not affect the efficiency of GAI auctions. The results are different for the case of substitutes. Here, it seems the the additive approximation performs well, and the performance starts to deteriorate only for extreme values of k. Very roughly, we can say that when relationship between attributes (within each GAI element) is limited to (mild) substitutions, it could be a good idea to use an additive approximation.

Unfortunately, our interpretation of the parameter k lacks quantitative scaling: there is no clear intuition of what the actual numbers mean, beyond the qualitative classification mentioned above. This is a subject for future research.

We also tested the other advantage promised by GAI auctions, which is the limited information revelation. The information revelation aspect has been studied systematically for auction AD (Parkes & Kalagnanam 2005). We have verified that also in GAI auctions, the buyer's information revealed is limited to a fraction of his private information (omitted).

Conclusions

We performed a simulation study of multiattribute auctions, comparing a mechanism that assumes additive preferences with a recent one we proposed that employs GAI preference structure. The study validates the usefulness of GAI auctions when preferences are non-additive but GAI, and allow us to quantify the advantages for particular preference models. In particular, we found significant benefit to supporting the accurate preference structure, especially when the GAI subutilities do not exhibit FOPI. When the local functions do exhibit structure of their own, in most cases the benefit of an accurate GAI model is still significant. An additive representation may be a reasonable approximation when the GAI structure is fairly close to an additive one, or when attributes within each GAI element are known to be substitutes.

We believe this study provides several methodological lessons applicable to a broader class of preference research problems. First, we consider the problem of generating structured random utility functions. The functions remain random to the extent that they conform to a specific GAI structure, and that the preference order over each element exhibit a structure of our choice: FOPI, or MUI with some predetermined level of complementarity or substitutivity. Second, we studied the problem of finding an additive approximation to an arbitrary GAI function. We found that performing linear regression using a relatively small set of random points, achieves an approximation that does as well as one done using all the points in the domain.

References

Bichler, M. 2001. *The Future of e-Markets: Multi-Dimensional Market Mechanisms*. Cambridge Univ. Press.

Boutilier, C.; Bacchus, F.; and Brafman, R. I. 2001. UCP-networks: A directed graphical representation of conditional utilities. In *UAI'01*, 56–64.

Braziunas, D., and Boutilier, C. 2005. Local utility elicitation in GAI models. In *UAI'05*, 42–49.

Che, Y.-K. 1993. Design competition through multidimensional auctions. *RAND Journal of Econ.* 24(4):668–680.

Dyer, J. S., and Sarin, R. K. 1979. Measurable multiattribute value functions. *Operations Research* 27:810–822.

Engel, Y., and Wellman, M. P. 2007. Generalized value decomposition and structured multiattribute auctions. In *ACM-EC'07*, 227–236.

Fishburn, P. C. 1967. Interdependence and additivity in multivariate, unidimensional expected utility theory. *Intl. Economic Review* 8:335–342.

Gonzales, C., and Perny, P. 2004. GAI networks for utility elicitation. In *KR'04*, 224–234.

Keeney, R. L., and Raiffa, H. 1976. *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*. Wiley.

Parkes, D. C., and Kalagnanam, J. 2005. Models for iterative multiattribute procurement auctions. *Management Science* 51:435–451.