

Computing the Communication Costs of Item Allocation

Timothy W. Rauenbusch
Artificial Intelligence Center
SRI International
Menlo Park, CA 94025
rauenbusch@ai.sri.com

Stuart M. Shieber and Barbara J. Grosz
Division of Engineering and Applied Sciences
Harvard University
Cambridge, MA 02138
shieber,grosz@eecs.harvard.edu

Abstract

Multiagent systems require techniques for effectively allocating resources or tasks to among agents in a group. Auctions are one method for structuring communication of agents' private values for the resource or task to a central decision maker. Different auction methods vary in their communication requirements. This paper makes three contributions to the understanding the types of group decision making for which auctions are appropriate methods. First, it shows that entropy is the best measure of communication bandwidth used by an auction in messages bidders send *and* receive. Second, it presents a method for measuring bandwidth usage; the dialogue trees used for this computation are a new and compact representation of the probability distribution of every possible dialogue between two agents. Third, it presents new guidelines for choosing the best auction, guidelines which differ significantly from recommendations in prior work. The new guidelines are based on detailed analysis of the communication requirements of Sealed-bid, Dutch, Staged, Japanese, and Bisection auctions. In contradistinction to previous work, the guidelines show that the auction that minimizes bandwidth depends on both the number of bidders and the sample space from which bidders' valuations are drawn.

1 Introduction

Multiagent system designers can achieve significant cost savings by making the correct choice of algorithm for team decision making. The results in this paper show that no single auction type minimizes bandwidth usage for all team sizes or for all possible valuations for the resource. For instance, Sealed-bid auctions require the least communication for small problems. The Dutch, Staged, and Bisection auctions each require least communication in some situations.

A Sealed-bid auction requires each bidder and the auctioneer to exchange 5 bits of information in a system with 60 agents where each agent's valuation is drawn independently and uniformly from the range \$1 to \$32. A Dutch auction requires an exchange of approximately one bit on average under

the same assumptions. A difference of four bits of information may seem insignificant by today's standards but modern systems may make millions or billions of related team decisions every second. While sacrificing no team decision quality, a system designer could save over 80 percent of its communication bandwidth just by implementing a different set of auction rules.

Previous work has made recommendations for the best choice of auction for making group decisions. However, the assumptions that led to those recommendations are incompatible with real systems in which communication bandwidth is costly, such as those using Internet-like networks.

This paper makes three main contributions to the understanding of communication for decision making in multiagent systems. First, we argue for entropy as the metric of communication bandwidth used by all messages exchanged. Communication in any multiagent system is made up of a series of messages that one agent sends to another. System designers need to choose an encoding for messages. For example, the number nine is commonly given the binary encoding "1001" but in ASCII code it is assigned the binary encoding "0011 1001". Measuring communication in decision-making algorithms using a particular message encoding could lead to results that are applicable only for that encoding. This paper uses principles of Information Theory to measure information in a coding-independent way. The receiver of a message can generate a probability distribution over the set of possible messages it can receive. The entropy of that distribution is a lower bound on the average size of the encoding for each message.

Second, we provide details of a three-step method for measuring bandwidth used by an algorithm. In the first step, the analyst builds a dialogue tree that represents all possible sequences of messages exchanged between the auctioneer and each bidder. In the second step, the edges of the dialogue tree are labeled with the probability associated with each message. Finally, in the third step, the expected information in the dialogue is calculated using the tree representation.

Third, we apply the analysis to Sealed-bid, Dutch (descending), Japanese (ascending), Staged (ascending), and Bisection auctions and provides system designers with the knowledge necessary to choose the auction that minimizes communication bandwidth. Auctions are particularly attractive for multiagent decision making because they provide a

way to structure the allocation of a resource or task to the member of a multiagent system that values it most, when the resource's value is private to each group member. Equivalently, auctions are used to assign a task to the member of a group that is best suited to perform it when the suitability of each group member to the task is private [Hunsberger and Grosz, 2000; Rauenbusch, 2004].

Our recommendations, based on a minimizing communication requirements, differ from those of economists and computer scientists. Economic analysis typically ignores communication costs entirely. Some computer scientists [Shoham and Tennenholtz, 2001] have focused on preference revelation, which concerns the willingness to disclose information. They consider only those messages sent from a bidder to an auctioneer and ignore message sent in the opposite direction. Some researchers [Grigorieva *et al.*, 2002] have used communication complexity or other metrics that assume a particular message encoding. Their results may be misleading for measuring bandwidth requirements in systems that employ more efficient encodings—our results are coding-independent.

This paper is organized as follows. In Section 2 the single item allocation problem is formally defined, and the five auctions are described. Next, Section 3 details the process for measuring communication in a dialogue using Dialogue Trees. Section 4 describes the application of dialogue trees to auctions. Guidelines for system designers choosing auction rules that minimize communication are given in Section 5. Section 6 highlights important related work and Section 7 gives conclusions and suggests areas for future work.

2 Item Allocation and Auctions

A single-item allocation problem is characterized by a group of n bidder agents and a seller agent (also called the auctioneer) that possesses a single, atomic item. Each bidder has a value for the item that is private and drawn independently and uniformly from the set of integers from 0 to $2^R - 1$ inclusive. Another way to look at a bidder's value is that it is being drawn from one of 2^R bins. The distribution from which each bidder's value is drawn is common knowledge. Bidder i 's value is denoted by x_i . The goal of the seller is to allocate the task to the bidder with the highest value. If there is a tie for the highest value, the task may be allocated to any of the bidders with the highest value. A *solution* to a single-item allocation problem is the index i , where x_i is the maximum value among all n bidders.

We analyze five auction types: Sealed-bid, Japanese, Staged, Dutch, and Bisection. This particular list of five auction types is representative of the range of auctions typically used to allocate a single item and is not intended to be exhaustive. For reference, the rest of this section provides a description of each auction type. Rauenbusch [2004] provides more detail, including pseudocode for each. In each auction, we assume bidders are honest. Prices are used to structure communication with the bidders and not as a tool for building in incentives for honesty.

Sealed-bid. All bidders send their value to the auctioneer. The winner is the bidder that sends the highest value.

Japanese (Ascending). The auctioneer maintains a *current price*, initially set to 0. The auctioneer sends each bidder in turn the current price. If a bidder's value is greater than or equal to the current price, it sends a message affirming its continued participation in the auction. Otherwise, it sends a message indicating its desire to leave the auction. The auctioneer then increments the current price, and repeats the process. If only one participating bidder remains in the auction after a round, the auction ends and that remaining bidder is the winner. If no participating bidders remain, the winner is chosen from the bidders in the previous round. Once a bidder leaves the auction, it may not rejoin.

Staged (Ascending). The auctioneer maintains a *current price*, initially set to 0. In Stage 1, the auctioneer sends bidder 1 the current price. If the bidder's value is greater than or equal to the current price, it sends its value and the current price is updated to this value. Otherwise, it sends a message indicating its desire to leave the auction. The auctioneer then moves on to Stage 2, sends the current price to bidder 2, and the process repeats. The auctioneer continues in this way with each bidder and the process ends after the n th stage. The winner is the last bidder that did not leave the auction.

Dutch (Descending). The auctioneer maintains a *current price*, initially set to $2^R - 1$. The auctioneer sends each bidder in turn the current price. The bidder sends a message indicating whether its value is equal to the current price. If no bidder's value is equal to the current price, the auctioneer decrements the price and repeats. If one or more bidder has value equal to the current price, the auctioneer chooses one as the winner.

Bisection. The auctioneer maintains an *lower bound* denoted l and *upper bound* denoted u , initially set to 0 and 2^R , respectively. The auctioneer also maintains a list of active bidders, initially the set of all bidders. The auctioneer calculates the *current price* as $u - \frac{u-l}{2}$. The auctioneer sends each bidder in turn the current price. Each bidder sends a message of either "Yes" or "No" to indicate whether its value is greater than or equal to the current price. If there are two or more bidders that sent a "Yes" message, the lower bound is updated to the current price, the set of active bidders updated to include only those that sent a "Yes" message, and the process repeats. If no bidder sent a "Yes" message, the upper bound is updated with the current price and the procedure repeats. If one bidder sent a "Yes" message, that bidder is declared the winner and the procedure ends. If the upper bound and lower bound differ by only one, one of the active bidders is chosen as the winner. After finding a winner, typically the bisection auction may proceed into a "price determination" phase that provides incentives for honesty. Because we assume honesty, the price determination phase is omitted from our analysis.

Message	Encoding		Probability	
	Enc1	Enc2	AlgA	AlgB
a	0000	0	0.0625	0.99
b	0001	10001	0.0625	0.000333
c	0010	10010	0.0625	0.000333
d	0011	10011	0.0625	0.000333
...
p	0011	10011	0.0625	0.000333

Table 1: Two encodings for sixteen messages used by Algorithms AlgA and AlgB

3 Communication Properties of a Dialogue

This section serves three main purposes. First, it presents an argument for the use of entropy and information theory to measure communication for team decision making. Second, it highlights the need to consider all communication. In auctions, this means that complete analysis requires evaluating communication in two directions: both from the bidders to the auctioneer and from the auctioneer to the bidders. Third, it presents dialogue trees—a tool for using entropy to measure the expected information transmitted in successive messages between agents. It details the use of dialogue trees in measuring communication for team decision making.

3.1 Entropy: Metric for Measuring Communication

A metric for measuring communication is required to compare auction rules by their communication cost. In each auction, information is exchanged between the auctioneer and each bidder by sending and receiving messages. In any implementation of an auction, the center and the bidders must agree to an encoding of messages.

Measuring information required by a multiagent algorithm using a particular encoding for messages may lead to misleading results. To illustrate why, we refer to the example given in Table 1. The columns labeled Enc1 and Enc2 shows two possible encodings for each of sixteen messages labeled a through p . Two algorithms, labeled AlgA and AlgB, each require one of sixteen messages to be sent from one agent to another but they differ in the frequency with which each message is sent. The probability associated with each message for each algorithm is shown in the two rightmost columns of the table.

With encoding Enc1, both AlgA and AlgB require four bits to transmit the message. But with encoding Enc2, AlgA requires 4.75 bits and AlgB requires 1.04 bits in expectation. Therefore, the algorithm that requires the least communication depends on the encoding chosen. Just as in this toy example, conclusions about the communication properties of auctions using a particular encoding are misleading because it is not clear whether those conclusions hold for other possible encodings. Work in Information Theory [Shannon, 1948; Cover and Thomas, 1991] has shown that the entropy of a random variable describing a message is a lower bound on the average size of the encoding for that message. Rather than evaluate an algorithm using a particular encoding, we

therefore use entropy to measure expected information communicated.

3.2 Direction of Communication

It is convenient to distinguish between *coordination messages*, which are those sent by the auctioneer to a bidder, from *revelation messages*, which are those sent by the bidder to the auctioneer. In this paper, the communication costs associated with coordination and with revelation are considered when calculating the expected information transmitted in an auction. In particular, the results provided are for the sum of coordination and revelation costs. This assumption is supported by Internet-like computer networks in which increased bandwidth requires costs associated with increased infrastructure for both directions of communication.

In a Sealed-bid auction, each bidder always reveals its value. Therefore, Sealed-bid auctions have the highest bandwidth requirements for revelation messages. As the results in Section 5 indicate, it would be misleading to rely on revelation messages alone when choosing an auction. Even though Sealed-bid auctions require more information transmitted in revelation than any other auction, they require no coordination. For that reason, they have low communication requirements in settings with small teams and coarse distributions from which bidders' values are drawn.

3.3 Dialogue Trees

A *dialogue* is a sequence of messages sent from one agent to another agent, in which the agent that sends the odd-numbered messages receives the even-numbered messages. Dialogue trees simplify the construction of a probabilistic model of the messages. In this section, we describe dialogue trees and provide a detailed method for calculating the expected information in a dialogue. We use dialogue trees to measure expected information in an auctions by analyzing the dialogue between the auctioneer and each bidder. Dialogue trees apply equally to other dialogues and are not limited to analysis of auctions.

A *dialogue tree* is a tree data structure with labeled edges. Each node represents a message, and is labeled with the message it represents. *Query messages* are those sent by the auctioneer to request a message from the bidder; *reply messages* are those sent by the bidder. *Status messages* are those sent by the auctioneer to which no reply is expected. Figures that represent dialogue trees (such as Figure 1) show query nodes, reply nodes, and status nodes enclosed by circles, boxes, and diamonds, respectively. $\text{Nodes}(d)$ denotes the set of all nodes in dialogue tree d .

The children of a node in a dialogue tree represent the sample space from which the next message is drawn, given that the message represented by the parent node has been sent. $\text{Children}(m)$ denotes the set of child nodes of node m . $\text{Parent}(m)$ denotes the parent node of node m .

A label on an edge between a parent and child node indicates the receiver's belief, prior to receiving the message, that the message represented by the child node is the one that the sender will send. $\text{In}(m)$ denotes the edge label that is incident on node m in a dialogue tree.

The edge labels define a probability distribution over the sample space represented by the children. The probability distribution and sample space together define a probabilistic model for messages in a dialogue.

In the auctions described in this paper, a bidder always sends a reply after receiving a query; therefore, a query node is never a leaf in a dialogue tree. A reply node may be either a leaf or a non-leaf node, depending on whether the center may follow the corresponding reply message with a message. A status node is always a leaf in a dialogue tree.

The remainder of this section details how a dialogue tree is used to calculate the expected information in a dialogue. The procedure uses edge labels for two purposes: to calculate the information content of a node and to calculate the probability of visiting a node.

The *information content* (IC) of node m is the entropy of the random variable represented by the labels of all edges originating at the node. Formally,

$$\text{IC}(m) = - \sum_{c \in \text{Children}(m)} \text{In}(c) \log \text{In}(c) \quad (1)$$

A leaf node therefore has information content of 0.

A path from the root node to each leaf node represents every possible dialogue between the two agents. The amount of information in a dialogue is the sum of the information content in each node on the path. Each of the possible dialogues represented by a tree has a different probability of occurring. This probability is the product of the edge labels along the path of the dialogue from the root of the tree to a leaf. The *probability of visiting* (PV) node m (that is, the probability that a message represented by a particular node will be sent in a dialogue) is the product of the probability of the message represented by its parent node and the label on its incident edge. There is unit probability of visiting the root node. Formally,

$$\text{PV}(m) = \begin{cases} 1 & \text{if } m \text{ is root} \\ \text{PV}(\text{Parent}(m)) \cdot \text{In}(m) & \text{otherwise} \end{cases} \quad (2)$$

The *contributed information* (CI) of a node m is the product of the amount of information represented by the node and the probability the node is visited. Formally,

$$\text{CI}(m) = \text{PV}(m)\text{IC}(m) \quad (3)$$

We use expected information in a dialogue as the metric for communication. Expected information of a dialogue (EI) represented by dialogue tree d is the sum of the contributed information of each node in d . Formally,

$$\text{EI}(d) = \sum_{m \in \text{Nodes}(d)} \text{CI}(m) \quad (4)$$

Contributed information provides a straightforward way to separate the information contribution of messages sent by the center from those sent by the bidder. The child nodes of a reply node represent messages sent by the center and the child nodes of a query node represent messages sent by the bidder. The amount of information sent by the bidder is the sum of the contributed information of all query nodes and the amount

of information sent by the center is the sum of the contributed information of all reply nodes. This is counter-intuitive and arises because contributed information of each node is derived from the probabilities associated with the edges *originating* at that node, which define the information content of the messages represented by its child nodes. Section 4 describes the dialogue tree in Figure 1 and how it is used to analyze the Bisection auction.

4 Analysis of Auctions

Using dialogue trees as a tool, in each auction we first determine the structure of the tree, then calculate the appropriate edge labels. To aid in determining the structure of the tree, the messages in each of the five auctions are divided into the following two types of query/response pairs: (1) best response, and (2) value. In a best response query, the auctioneer sends the bidder a message that includes a price. The bidder then responds with the message *Yes* if its value is higher than the price and the message *No* otherwise. In a value query, the auctioneer sends a message, and the bidder responds by sending a message containing its value.

Decomposing these algorithms into two types of constituent query/response pairs is a tool used to simplify of the analysis. The measurement of the expected information in a dialogue for each auction is independent of this decomposition. For example, if a bidder in the Staged auction responds *Yes* when sent the first message, it always sends its value. It is therefore not necessary to send a query message for the bidder's value after receiving the response. But, there is zero communication cost for the value query (because the probability of sending it given a *Yes* response is 1).

Two methods are used to determine the edge labels. The first and simplest way to determine the edge labels is by simulation. An auction is run many times in simulation, and the frequency of each message is recorded and used for the edge labels. The main advantage of this approach is that it requires little labor, after coding the algorithm. One disadvantage of the simulation method is that the time required to run the many simulations needed to accurately estimate the frequency of low-probability messages usually found near the leaves of the dialogue tree may be prohibitive. In addition, this method requires a different simulation for each setting of parameters of interest. For example, the results given in Figure 2 would require 1220 sets of simulations: one for each of 122 team sizes and 10 settings for the number of bins.

The second method is to calculate the edge labels analytically. This approach uses the common knowledge from which the bidder's value is drawn, and the knowledge acquired through messages represented by higher levels of the tree. The main drawback with this approach is that it is labor intensive because an analyst must reason about the receiver's mental model for each message in each algorithm. The main advantage of this approach is that the procedure for generating edge labels in one particular setting (e.g., for a team of 20 agents and 4 bins) applies equally well to other settings (e.g., 21 agents and 8 bins) by substituting appropriate parameters. An additional advantage is that the edge labels are calculated precisely rather than estimated.

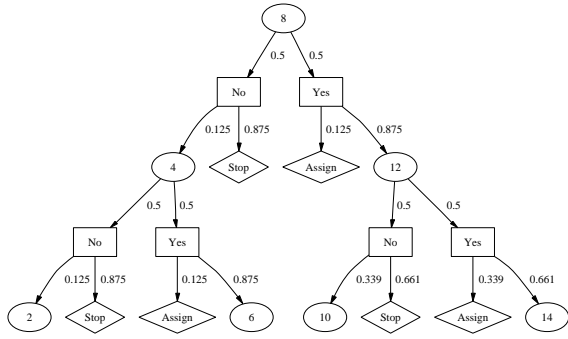


Figure 1: Highest three levels of a dialogue tree for Bisection auction with four bidders and sixteen bins

The results presented in this paper were based on generating edge labels using the second method. The first method was used to verify the results. The rest of this section provides an example of a dialogue tree for the Bisection auction to illustrate the use of dialogue trees to measure the expected amount of information transmitted in the five auctions. Details of the analysis have been omitted due to lack of space. Rauenbusch [2004] provides the details of the analysis of the dialogue trees for each auction.

The calculation of the edge labels in any dialogue tree involves reasoning about the knowledge of the receiver of each message: the distribution from which the bidder’s value is drawn and all messages represented in higher levels of the tree. Figure 1 shows the dialogue tree that represents the first five messages exchanged between the auctioneer and one bidder in a Bisection auction. In the tree, the message containing the best response query with value b is represented by a query node with label b .

To provide an example of the reasoning involved in computing edge labels, we specifically consider the edges on the path from the root node labeled 8 to the leaf node labeled 14. Calculation of edge labels in the figure assumes that there are four bidders, with values drawn from 16 bins—0 through 15 inclusive.

The root of the tree corresponds to the best response query with value 8. The bidder replies to this query with *Yes* if its value is greater than or equal to 8, and *No* otherwise. The receiver of the *Yes* or *No* message—the auctioneer—believes that the *Yes* message will be sent with probability 0.5 because it knows the distribution from which the bidder’s value is drawn. Therefore, the edge into the *Yes* node is labeled 0.5.

To compute the next edges, labeled 0.125 and 0.875, we first assume that the bidder sent a *Yes* response to the first query. The bidder will win the auction (and will be sent a message indicating that it is assigned to the item) if and only if no other bidder sent a *Yes* response to the first query. Given the common knowledge that bidders’ values are distributed uniformly between 0 and 15, the probability that all three other bidders sent a *Yes* query is $(0.5)^3 = 0.125$. Therefore, the edge incident on the *Assign* node is labeled 0.125, and the edge incident on the 12 query node is labeled with its complement 0.875.

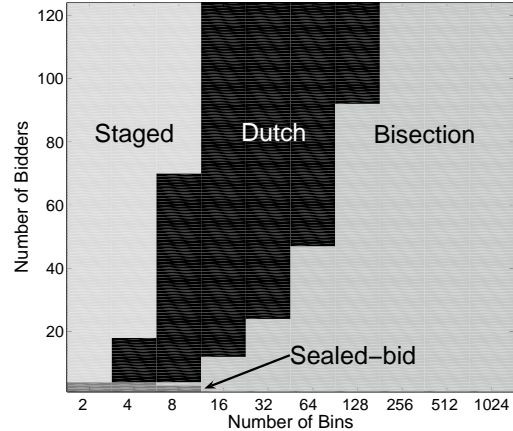


Figure 2: Algorithm with lowest expected information transmitted for varying numbers of bidders and bins

The edges incident on the next reply nodes are labeled 0.5. The auctioneer knows (1) that the bidder’s value was drawn uniformly from 0–15 by common knowledge; and (2) the bidder’s value is greater than 8 by virtue of the *Yes* response represented in a higher level of the tree. Therefore, the auctioneer’s believes that there is a probability of 0.5 that the bidder’s value is higher than 12.

The calculation of the edge labeled 0.661, incident on the node labeled 14 in the tree, is complex and full details are omitted. To get a feeling for why, the analysis begins with the knowledge that given that the bidder sent the *Yes* message represented by the top of the edge, the message represented by the node labeled 14 will be sent if and only if at least one other bidder also has value greater than 12. But the bidder knows that at least one other bidder had value greater than 8. The calculation involves the bidder assigning a belief vector representing is belief that each of one, two, and three other bidders still remain in the auction. The value 0.661 is then computed using this vector.

5 Results

Figure 2 indicates the algorithm that has lowest expected information transmitted for increasing numbers of bidders and for increasing numbers of bins. It clearly shows that choosing the algorithm that needs least expected information transmission is highly dependent on the two parameters of the environment. For large numbers of bidders and bins, Bisection requires the least communication. Sealed-bid, Dutch, and Staged auctions each require the least communication for particular parameter settings.

For a very small number of bidders and bins (fewer than five bidders with two or four bins, and fewer than three bidders with eight bins) the Sealed-bid auction performs best. A sealed-bid auction by definition requires the maximum amount of revelation and no coordination. Therefore, for very small problems, the savings in revelation from any other auction method are outweighed by the cost of coordination.

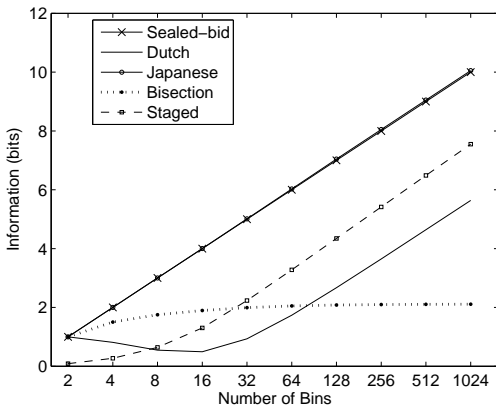


Figure 3: Expected information transmitted per bidder for varying numbers of bins with 60 bidders

When there are two bins, the Japanese auction has the same communication properties as the Sealed-bid auction because the first and only query in the Japanese auction is always sent and the bidder reveals its value (by its response that indicates whether its value is in the higher or lower bin).

For all but the smallest numbers of bidders and bins, the Bisection, Dutch, and Staged Japanese auctions perform well. The graph in Figure 3 shows the expected amount of information transmitted between the center and each bidder for a varying number of bins for a constant 60 bidders.

The first thing of note on the graph is that the communication requirements of the Sealed-bin auction increase linearly as the number of bins increases exponentially. The Sealed-bid auction has zero coordination cost and a revelation cost that is logarithmic in the number of bins.

The graph shows that as the number of bins increases exponentially, the expected amount of communication required by the Bisection auction rises then levels off. For a small number of bins, the Staged auction has very low communication requirements. For small numbers of bins, the Dutch auction's communication requirements actually decrease as the number of bins increase. Therefore, as the number of bins increases, the auction with the lowest communication costs is first the Staged auction, then the Dutch auction and finally the Bisection auction.

6 Related Work

Economic analysis of auctions [Rasmussen, 1989, inter alia] focuses on the effect of auction rules and prices on the strategies of non-cooperative bidders. While this paper is concerned with systems in which strategies can be imposed by methods external to the auction itself, dialogue trees can be used to measure communication requirements of all types of auctions. In multiagent systems where the assumption of externally imposed incentives does not hold, dialogue trees can be used to compare the communication costs of auctions that impose desirable incentives on the bidders.

Researchers in computer science have used several alternatives to entropy for measuring communication in multiagent

decision making. One such approach counted the number of messages required to arrive at a team decision [Ortiz *et al.*, 2003], which is equivalent to assuming that each message has a fixed length. In systems with communication channels that carry encoded messages, the assumption that each message has a fixed length does not hold. Under a fixed length assumption, the Sealed-bid auction would always be preferred. Thus, such analyses may be misleading because an algorithm with fewer fixed-length messages will not always be the cheaper algorithm in terms of expected information transmitted.

Sunderham and Parkes [2003] measure the volume remaining in the space of feasible private information after bidders have sent the auctioneer constraints on their private information in a multi-attribute auction. They use this metric to compare the amount of revelation in auctions. For our purposes, entropy is a preferred metric because it provides a direct measure of bandwidth required by an auction and it provides the common currency of bits to measure both coordination and revelation.

Communication complexity [Kushilevitz and Nisan, 1996] provides an alternative method for analyzing communication between agents. Grigorieva *et al.* [2002] use communication complexity to analyze the bisection auction. Communication complexity evaluates the worst case amount of communication required for two agents to compute a function. The communication complexity model assumes that sending each binary message costs one bit. If any prior information is available, it is ignored for the purposes of calculating communication complexity. As long as there is *some* arbitrarily small possibility that an agent will send a '0', that communication costs one bit. Protocol trees [Yao, 1979] are used as a tool to evaluate communication complexity of an algorithm while dialogue trees are used to calculate expected information in a dialogue that represents messages sent in an algorithm.

The main benefit of this assumption is that there is no need to assume a prior distribution, and that simplifies the analysis. The main drawback is that it assumes a particular encoding of messages and therefore no savings can be attained by alternative encodings. A system designer that relies on communication complexity in choosing an auction will select an auction that performs well under a worst case assumption of the encoding cost of *each message*. In this paper, we assume that system designers prefer choosing an auction based on the expected information transmitted.

Shoham and Tennenholtz [2001] use a method related to communication complexity for the analysis of the functions computed in team decision-making mechanisms. They define f as the maximum value of n bidders' willingness to pay for an item, where each bidder i has a willingness to pay of x_i . They imply that the domain of x_i is continuous on the interval $(0, \text{maxprice})$ and assume that each bidder i can communicate x_i to the auctioneer with one bit by making use of a common clock. They claim that by using an auction similar to the Dutch auction, the function f can be computed by a single bidder communicating a single bit.

In both Yao's theory of communication complexity and Shannon's theory of information [Shannon, 1948], the cost of communicating an arbitrary value drawn from a continuous interval is infinite, not a single bit, because there is an

infinite number of messages that the bidder can send to the center. The theory of information makes assumptions that are consistent with modern wired and wireless computer networks, in which messages can be encoded. Shoham and Tennenholtz' critical assumption that a continuous value may be communicated in one bit does not hold in modern multiagent systems.

Relying on Shoham and Tennenholtz' assumptions would lead a system designer to always choose their version of the Dutch auction to minimize the amount of communication from the bidder to the center. This paper shows that the expected amount of information communicated by an algorithm is highly dependent on the number of bidders and the distribution of bidders' private values. The Dutch auction is often not the algorithm that minimizes the expected amount of communication from the bidder to the center. Therefore, a system designer that relies on Shoham and Tennenholtz' assumption may incur unnecessary costs.

Much prior work [Shoham and Tennenholtz, 2001; Sunderam and Parkes, 2003, *inter alia*] has centered around measuring how much of a bidder's preferences are revealed by an algorithm instead of how much bandwidth is used by an algorithm. Therefore, a common assumption has been that coordination messages are free while revelation messages are costly. Under that assumption, it is desirable to select an algorithm with low revelation costs, even if it has high coordination costs. The results presented in Section 5 are for the sum of revelation and coordination costs and differ from such prior work for several reasons. However, situations in which only one direction of communication is important can be handled easily by the models described in this paper by ignoring the other direction in the analysis.

7 Conclusion and Future Work

In this paper, we presented three major contributions. First, we presented an argument for measuring expected information transmitted in a dialogue to determine the bandwidth need by multiagent algorithms. Second, we provided a method for measuring expected information using dialogue trees. Third, we showed that using that method to analyze five auctions leads to recommendations for multiagent system design that differ from recommendations made in previous work. The results of the analysis indicated that the correct choice of auction depends on the number of bidders and the size of the sample space from which bidders' values for the item are drawn. The Staged, Dutch, and Bisection auctions are each appropriate for different situations, and the Sealed-bid auction is best for very small problems. The guidelines presented in this paper could lead to real savings in communication bandwidth with no loss in decision quality.

In future work, we plan to use dialogue trees to analyze algorithms for more general team decision problems than single-item assignment and for more general algorithms than auctions. Auctions are commonly suggested for item or task assignment in multiagent systems because they are a convenient method for structuring communication between agents. We plan to compare other methods for allocating a single item, such as inter-agent exchange, to auctions. We assumed

that agents were honest—small adjustments to the auction rules instead allow us to build incentives into an auction directly. We plan to evaluate the communication costs incurred by auctions with built-in incentives and analyze the impact of those incentives on the correct choice of auction method.

8 Acknowledgments

This material is based on work supported by the National Science Foundation under Grant No. IIS-9978343.

References

- [Cover and Thomas, 1991] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley-Interscience, 1991.
- [Grigorieva *et al.*, 2002] Elena Grigorieva, P. Jean-Jacques Herings, Rudolf Müller, and Dries Vermeulen. The private value single item bisection auction. In *METEOR Research Memoranda*, number RM/02/035. Maastricht University, 2002.
- [Hunsberger and Grosz, 2000] Luke Hunsberger and Barbara J. Grosz. A combinatorial auction for collaborative planning. In *Proceedings of the Fourth International Conference on Multi-Agent Systems (ICMAS-2000)*, 2000.
- [Kushilevitz and Nisan, 1996] Eyal Kushilevitz and Noam Nisan. *Communication Complexity*. Cambridge University Press, 1996.
- [Ortiz *et al.*, 2003] Charles L. Ortiz, Timothy W. Rauenbusch, and Eric Hsu. Dynamic resource-bounded negotiation in non-additive domains. In Victor Lesser, Charles L. Ortiz, Jr., and Milind Tambe, editors, *Distributed Sensor Networks: A Multiagent Perspective*. Kluwer Academic Publishers, 2003.
- [Rasmussen, 1989] Eric Rasmussen. *Games and Information*. Blackwell, 1989.
- [Rauenbusch, 2004] Timothy W. Rauenbusch. *Measuring Information Transmission for Team Decision Making*. PhD thesis, Harvard University, 2004.
- [Shannon, 1948] C. E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27, 1948.
- [Shoham and Tennenholtz, 2001] Y. Shoham and M. Tennenholtz. Rational computation and the communication complexity of auctions. *Games and Economic Behavior*, 35(1–2):197–211, 2001.
- [Sunderam and Parkes, 2003] Aditya V. Sunderam and David C. Parkes. Preference elicitation in proxied multiattribute auctions. In *Proceedings of the Fourth ACM Conference on Electronic Commerce*, 2003.
- [Yao, 1979] Andrew Chi-Chih Yao. Some complexity questions related to distributive computing (preliminary report). In *Proceedings of the eleventh annual ACM symposium on Theory of computing*, pages 209–213, Atlanta, 1979.