Exponential Communication Inefficiency of Demand Queries

Noam Nisan   Ilya Segal

November 4, 2004

1 Introduction

Nisan and Segal (forthcoming) show that when verifying the efficiency of a combinatorial allocation, without increasing the communication burden one can restrict attention to announcing a price equilibrium. We find that a parallel conjecture for *deterministic* communication fails: We demonstrate a class of valuations for which the restriction to "demand queries," which ask agents to report their preferred allocations at given (possibly nonlinear) allocation prices, brings about an exponential blowup in the communication burden of finding an efficient allocation. Namely, for this class, an efficient mechanism exists that uses a number of bits that is proportional to the number of objects, but any demand-query mechanism that achieves efficiency, or even any improvement upon the "dictatorial" allocation of all the objects to one agent, must use an exponential number of demand queries. We also show a parallel average-case result: We construct a joint probability distribution over the agents' valuations from this class for which any improvement in the expected surplus over the dictatorial allocation requires using an exponential expected number of demand queries.

Our results bring into question the usefulness of combinatorial auction mechanisms such as "iterative auctions," and other "preference elicitation" mechanisms that use demand queries or value queries. Of course, the results still open the possibility that demand-query mechanisms work well on some classes of valuations, or on average for some "realistic" probability distributions of valuations, but such cases are yet to be characterized.

2 The Valuation Class

Consider the problem of allocating items from set M between two agents, with $|M| = m$. Let K denote the set of allocations in which the two agents receive the same number of items, and so $k \equiv |K| = \binom{m}{m/2}$ (for simplicity let m be even). Suppose we know that each agent i's valuation satisfies $v_i(S) \in \{0,1\}$ for all $S \subset M$, with $v_i(S) = 0$ for $|S| \leq m/2$ and $v_i(S) = 1$ for $|S| \geq m/2$.

Let

$$\begin{aligned} A_1 &= \{S \subset M : |S| = m/2,\, v_1(S) = 1\}, \\ A_2 &= \{S \subset M : |S| = m/2,\, v_2(N \backslash S) = 1\}. \end{aligned}$$

We suppose that it is a priori known that $|A_1|, |A_2| > k/2$, hence $A_1 \cap A_2 \neq \varnothing$ and an allocation $(S, N\backslash S)$ with surplus 2 exists. Then efficiency is equivalent to finding such an allocation. On the other hand, without finding such an allocation we achieve at most surplus 1, which could have been achieved by giving all the objects to agent 1.

3 A Fast Efficient Protocol

Proposition 1 *There exists a protocol finding an efficient allocation using no more than* $4(\log_2 k)^2$ *bits of communication.*[1]

Proof. Consider the following protocol: At each step r, we maintain a set $K_r \subset K$ with the property that

$$|A_1 \cap K_r| + |A_2 \cap K_r| > |K_r|, \tag{1}$$

and therefore $A_1 \cap A_2 \cap K_r \neq \varnothing$. Initialize with $K_0 = K$, which satisfies (*) by assumption. At each step r, partition K_r (arbitrarily) into two subsets B_1 and B_2 such that $||B_1| - |B_2|| \leq 1$, and ask each agent $i = 1, 2$ to report $a_{ij} = |A_i \cap B_j|$ for each $j = 1, 2$. Then take $K_{r+1} = B_j$ for the j that has the higher value of $a_{1j} + a_{2j} - |B_j|$, which guarantees that K_{r+1} satisfies (*).

[1] In fact, there exists an efficient protocol that uses no more than $5.3 \log_2 K$ bits of communication. This follows from the fact that this communication problem turns out to be equivalent to the monotone depth of the majority function (Karchmer and Wigderson, 1988), and Valiant's (1984) celebrated construction of such formulae.

Each step requires communicating no more than $4\lceil\log_2|K|\rceil$ bits, and in no more than $\lceil\log_2|K|\rceil$ steps K_r becomes a singleton, which by (*) must be an element of $A_1 \cap A_2$. ■

4 Demand-Query Protocols

Now we restrict to *demand-query protocols*: at each step, an agent i is offered a price vector $p : 2^M \to \mathbb{R}$ and an ordering τ over 2^M, both of which can be functions of the agents' previous messages, and reports the first element of $\arg\max_{S\subset M}(v_i(S) - p(S))$ in ordering τ. Note the importance of fixing a tie-breaking ordering in advance: If agent i's tie-breaking were allowed to depend directly on his valuation v_i, then his choice from a known tie could communicate arbitrary information about v_i, and so the restriction to demand-query protocols would not have any bite.

Another often-used type of query is a "value query," which asks an agent i to report his valuation $v_i(S)$ for a given bundle $S \subset M$. Note that in our simple model such a query is equivalent to a demand query with proces $p(S) = 0$ and $p(T) = 1/2$ for all $T \neq S$ (S will be demanded if and only if $v_i(S) = 1$). Thus, lower communication bounds for demand-query protocols will also apply to protocols that use value queries.

5 The Worst-Case Result

Proposition 2 *Any demand-query protocol that achieves a higher surplus than that from giving all items to one agent must ask at least $k/2-1$ queries (in the worst case).*

Proof. Take any protocol that uses less than $k/2-1$ demand queries. We describe an "adversary algorithm" for answering a sequence of queries made by the protocol and then constructing valuations consistent with all the answers, for which the total surplus for the protocol's outputted allocation is at most 1. (While choosing the valuations *after* the queries have been made seems like "cheating," the point is that they *could have* been the valuations from the outset.)

The adversary algorithm at each step maintains sets $B_1, B_2 \subset K$ (with the interpretation that B_i is the allocations in K for which agent i "could still have" value 1). The two sets are initialized with $B_1 = B_2 = K$. At each step of the protocol, if agent $i = 1, 2$ is queried, the adversary returns a bundle S^* as though the agent's valuation is described by $A_i = B_i$. Furthermore, if $S^* \in B_i$, then S^* is removed from B_{-i}. Proceed to the next step. Suppose that the protocol ultimately outputs allocation T. Then, for each $i = 1, 2$, if $T \in B_i$, T is removed from B_{-i}. Finally, the adversary sets $A_i = B_i$ for $i = 1, 2$.

Since the protocol has less than $k/2 - 1$ steps, at each of which each $|B_i|$ is reduced by at most 1, and the outputted allocation reduces $|B_i|$ by at most 1, in the end we have $|A_i| > |K|/2$ for $i = 1, 2$, so the constructed inputs are feasible. Furthermore, by construction we have $T \notin A_1 \cap A_2$. What remains to show is that the demands reported by the agents at each stage are consistent with the constructed inputs A_1, A_2. To see this, note first that the sets B_i are nonincreasing and thus at each stage, $A_i \subset B_i$. This implies that if at some stage bundle $S^* \notin B_i$ was demanded by type B_i, it will also by demanded by type A_i at the same prices and the same tie-breaking rule. On the other hand, if a bundle $S^* \in B_i$ was demanded for type B_i, then it was removed from B_{-i}, and so by construction it always remains in B_i. Then $S^* \in A_i$ and so S^* will also be demanded by type A_i at the same prices and the same tie-breaking rule. Hence, the constructed type A_i will indeed induce the same demands as those constructed by the adversary. ■

Since by Stirling's formula $k = \binom{m}{m/2} \sim \sqrt{2/(\pi m)} \cdot 2^m$ as $m \to \infty$, the result means that any demand-query protocol improving upon giving all objects to one agent requires an exponential number of queries in m, while by Proposition 1 there exists a different protocol that achieves efficiency using $O(m^2)$ bits.

Remark 3 *Sandholm and Boutilier (forthcoming) consider protocols using "rank queries," which ask an agent to report the bundle of rank r in the order of his valuations, with ties broken according to some a priori ordering τ. Such queries prove more powerful than demand queries for our valuation class. For example, such a query reveals whether the first q bundles in ordering τ contain at least r value-1 bundles. Using bisection on r between* 1 *and q, we find the number of value-1 bundles with among the first q bundles with at most* $\log_2 q$ *rank queries. Then we can use the protocol described in Proposition 1 doing this at each step, and so the total number of order queries is at most* $4(\log_2 k)^3$*. On the other hand, for other valuation classes rank queries may not achieve efficiency at all because they do not elicit the strength of the agents' preferences: For example, with only one object, an agent would* *always ranks the object higher than not having it, and we would never learn which agent has the highest valuation for the object.*

6 An Average-Case Result

We start by defining a joint probability distribution D over valuation pairs (A_1, A_2) as the uniform distribution over pairs (A_1, A_2) such that $|A_1| = |A_2| = k/2 + 1$ and $|A_1 \cap A_2| = 2$.[2]

Proposition 4 *For the joint probability distribution D, any demand-query protocol obtaining an expected surplus of at least $1 + \delta$ must use at least $T(\delta) = \frac{\max\{\delta, 1/12\}k-2}{7 \ln k}$ queries in the worst case, and at least $(\delta/2) T(\delta/2)$ queries in the average case.*

Proof. Consider first a demand-query protocol asking at most t queries in the worst case. We allow agents to reveal even more information than the demanded bundle. For a given demand query $\langle p, \tau \rangle$, define ordering π over allocations defined by increasing prices, with ties broken according to τ. When agent i is asked such a query, let him reveal his valuations for the top r allocations in ordering π out of those that have not been revealed yet, and let the other agent reveal his valuations for the same allocations (where $r \geq 1$ is a fixed integer). Furthermore, if agent i has valuation zero for all such allocations, let him say "bingo," and both agents reveal all their valuations. This response is more informative than answering the demand query: even if agent i did not say "bingo" and so his valuations are not fully known, his demand would be the highest value-1 allocation in ordering π, which has been revealed.

The probability that an agent said "bingo" in response to one of the queries is bounded above by $t \left(\frac{k/2}{k - rT} \right)^r$, since the fraction bounds above the

[2]It is important that the two agents' valuations are jointly distributed. If the valuations were drawn independently from the uniform distribution over valuations described by $|A_i| = k/2 + 1$, we could achieve efficiency with a small expected number of demand queries, by announcing allocations from K in any fixed order and stopping as soon as we find an efficient allocation (which is verified with two value queries, which are equivalent to demand queries). The probability that a given allocation has value 1 to a given agent is at least $1/2$,(higher when more allocations have been checked), and so it is efficient with probability at least $1/4$. Thus, the expected number of allocations that need to be checked before finding efficiency is at most 4, and so the expected number of demand queries is at most 8 (regardless of k).

proportion of either agent's zero-valuation allocations among those that have not been revealed, and for "bingo" we need all the top r allocations in ordering π to be such allocations. Also, if "bingo" has not been said and the protocol outputs some allocation, the probability that this is an efficient allocation is bounded above by $(rt+1)\frac{2}{k}$, since this is the probability that at least one of the two efficient allocations is either one of rt allocations revealed by the agents or in some other allocation outputted by the protocol. Thus, the probability of finding an efficient allocation is bounded above by

$$t\left(\frac{k/2}{k-rt}\right)^{r}+(rt+1)\frac{2}{k}$$

We can choose any integer $r\geq 1$, and to (roughly) minimize this expression we choose $r=3\ln k$.

Suppose $T\leq\delta\frac{K}{\ln K}$, with $\delta\leq\frac{1}{12}$ Then the probability of efficiency is at most

$$\begin{aligned}&\delta\frac{k}{\ln k}\left(\frac{1}{2(1-3\delta)}\right)^{3\ln k}+6\delta+\frac{2}{k}\\ \leq\ &\delta\frac{k}{k^{3\ln(3/2)}\ln k}+6\delta+\frac{2}{k}\leq 7\delta+2/k.\end{aligned}$$

Thus, to get prob. of efficiency at least δ we need at least $T(\delta)=\frac{\max\{\delta,1/12\}k-2}{7\ln k}$ queries.

Suppose now we have a demand-query protocol with an *expected* number t of queries that finds efficiency with probability δ. Then we can terminate it after $2t/\delta$ queries (and in this case output a random allocation from k). The probability that the protocol is terminated is at most $\delta/2$, and so we still have a protocol that finds efficiency with probability $\delta/2$ at most $2t/\delta$ queries in the worst case. By the previous result, $2t/\delta\geq T(\delta/2)$, and so $t\geq(\delta/2)\,T(\delta/2)$. ■

Remark 5 *If we restricted our valuation class to have* $|A_i|\geq 2k/3$ *for* $i=1,2$*, we could always achieve efficiency with a small expected number of demand queries. Indeed, consider again the randomized protocol that picks an allocation from* K *uniformly at random and stops as soon as it finds an efficient allocation (which is verified with two value queries, which are equivalent to demand queries). Since there are at least* $k/3$ *efficient allocations, the probability of finding one in each step is at least* $1/3$*, hence the expected* *number of queries before stopping is at most* 6 *(regardless of* k*). By the Minimax Theorem, this also implies that for every probability distribution on such valuation pairs, there exists an efficient* deterministic *demand-query protocol whose expected number of queries is 6. Thus, in this case we obtain a divergence between the average-case and worst-case communication complexity of demand-query protocols. (For the latter, note that Proposition 2 is easily extended to this case to show that any demand-query protocol that achieves a surplus greater than 1 must still ask at least* $k/3$ *queries in the worst case.)*

7 Conclusion

We have shown a simple example in which a restriction to demand queries brings about an exponential blowup in the communication required to achieve or approximate efficiency. A natural direction for further research is to characterize the valuation classes for which this does not happen. Another important question is whether there exists a sufficiently restricted "universal query class" to which we can restrict attention without causing an exponential communication blowup of finding efficiency on any valuation class. The results of Nisan and Segal (forthcoming) imply that demand queries do form a universal class for nondeterministic communication, but a parallel question for deterministic communication remains open. Suggesting a universal query class would useful for designing practical deterministic mechanisms, while proving that it does not exist would suggest that the practical mechanisms should be very dependent on the valuation class at hand.

References

[1] M. Karchmer and A. Wigderson. Monotone circuits for connectivity require super-logarithmic depth. In Proceedings of the 20th ACM Symposium on Theory of Computing, pages 539–550, 1988.

[2] N. Nisan and I. Segal. The Communication Requirements of Efficient Allocations and Supporting Prices. Journal of Economic Theory, forthcoming.

[3] T. Sandholm and C. Boutilier. Preference Elicitation in Combinatorial Auctions, in P. Cramton, R. Steinberg, and Y. Shoham, Handbook on Combinatorial Auctions, Chapter 10, MIT Press, forthcoming.

[4] L. Valiant. Short Monotone Formulae for the Majority Function. J. Algorithms, 5, pages 363–366, 1984