COMMUNICATION COMPLEXITY AND MECHANISM DESIGN

Timothy Van Zandt
INSEAD

19 October 2006

Abstract
The curse of the Revelation Principle is that it leads us to focus on unrealistic mechanisms in which agents report all private information to the principal, who then makes all decisions centrally. This is not true when communication costs are introduced. Then partial communication of information, sequential back-and-forth conversations, and decentralization of decisions become part of optimal mechanisms. This paper looks at the interplay between the incentive constraints and the communication constraints. When can they be separated, so that e.g. one can separately study the incentive compatibility of a social choice rule and the minimum-cost communication protocol that realizes the social choice rule? (JEL: D82, C72)

1. Introduction

The Revelation Principle in mechanism design is both a blessing and a curse. It is a blessing because it so greatly simplifies the design of optimal mechanisms and allows one to abstract from the details of the mechanism and focus on the social choice rule to be implemented. It is a curse because direct mechanisms provide such an unrealistic picture of decision-making in organizations. Only in the simplest of cases, such as private-values auctions of a single good, does it ring true that the privately-informed agents reveal all their information to a central authority, which then determines all decisions according to a pre-specified contract.

The obvious explanation of why organizations are run by intricate and indirect flows of partial information is that it would be too complex for everyone to communicate all information to one person and for this one person to process and use this information. The integration of mechanism design and complexity considerations using formal models of complexity has been labeled *algorithmic mechanism design* in the field of theoretical computer science, where this area of research has recently received considerable attention.

There are two aspects to this complexity: communication complexity and computational complexity. Both capture certain limits on human brainpower. In the case of communication, the main bottlenecks are not the networks, mail systems, or other infrastructures that transmit information from one human to another, but rather the processing

Acknowledgements: The author is a member of the CEPR.

Email address: timothy.van-zandt@insead.edu

ability of the humans, particularly on the receiving end.

This note looks at communication complexity in mechanism design. One motive for studying communication (rather than computation) complexity is convenience. Communication is modeled by a message game—just like our mechanisms—and so we do not have add new machinery to the description of a mechanism; we only add a measure of communication complexity. In contrast, to model computation, we need to bring in models of information processing that are not currently part of mechanism design and that tend to be unfamiliar to economists. The two parts do not fit neatly together.

The main theme of this note is to consider when communication complexity and incentive constraints interact in an interesting (and complicated) way. We show that, for ex post Nash implementation, there is a nice separation between the checking of incentive compatibility constraints of a social choice function and the design of a communication-efficient protocol to compute the social choice function. This separation does not exist for the other notions of incentive compatibility that we consider. One implication of the separation is that (when it holds) incentives constraints do not increase the communication complexity of computing a social choice function. However, this implication is not as strong as it sounds. For example, in a model with quasilinear preferences, if the principal does not care about the transfers, then the cost of computing a decision rule for the non-transfer part of an outcome is increased when there are incentive constraints—due to the need to compute also transfers that make the social choice function incentive compatible (Fadel and Segal (2006)).

Besides the point outlined in that paragraph, this note is also intended to be a brief introduction to the modeling of communication complexity in mechanism design. The focus is on formal models of such complexity, and we do not attempt to review the papers that have already studied mechanism design with communication constraints. There are quite a few, though still a small part of the mechanism design literature. Seminal papers include Green and Laffont (1986, 1987), Melumad, Mookherjee, and Reichelstein (1992, 1997), and Laffont and Martimort (1998). See Poitevin (2000) and Mookherjee (2006) for surveys. Nor will we attempt to survey the literature on communication complexity that does not take into account incentives (which includes both work in computer science and early work on message space size in economics). A recent contribution in this area that is also a good introduction and guide to the literature is Nisan and Segal (2006).

2. Elements of a model

Let's fix notation for some standard components of static mechanism design. ("Static" refers to the fact that agents receive private information once and decisions are made at just one point in time, as opposed to a flow of information and decisions. The mechanisms may still involve dynamic message games.) Exogenous parts of the model including the following.

- Set $N = \{1, \ldots, n\}$ of agents.
- $\forall i \in N$: set Θ_i of "types". (Let $\Theta = \Theta_1 \times \cdots \times \Theta_n$.)
- Set X of outcomes.
- Preferences $x \succsim_\theta^i x'$ for each i that depend on the state $\theta \in \Theta$.

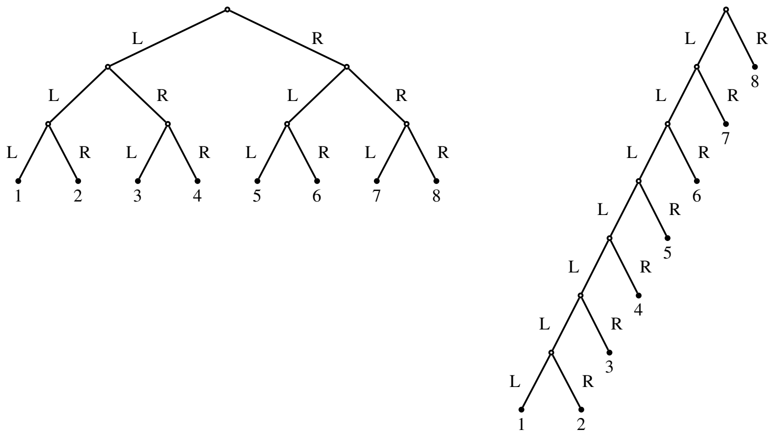

Figure 1. Two binary trees for communicating $\theta \in \{1, \dots, \Theta\}$.

- (If Bayesian) utilities $u_i : X \times \Theta \to \mathbb{R}$.
- (If Bayesian) probability measure p on Θ.

We then determine endogenously the following.

- A social choice function $f : \Theta \to X$.
- A mechanism, which is an extensive game form Γ with terminal histories assigned to outcomes in X. Actions in the mechanism are called messages.
- A plan of action for player i specifies the message that player i sends at each information set. Let S_i be the set of plans of action, with typical element s_i.
- A strategy σ_i for player i specifies each message as a function of the player's type, i.e., $\sigma_i : \Theta_i \to S_i$. Let Σ_i be the set of strategies.
- For a profile s of plans of action, let $\lambda(s)$ be the outcome at the terminal node reached when players use s.
- A pair (Γ, σ), where Γ is a mechanism and σ is a strategy profile, is called a *protocol*. A protocol (Γ, σ) "computes" a s.c.f. f, defined by $f(\theta) = \lambda(\sigma_1(\theta_1), \dots, \sigma_n(\theta_n))$.

3. One-way communication complexity: bits

This note will not get into the finer details of measuring communication complexity. We will merely outline the most common measure when communication is discrete: bits.

For this purpose, suppose that a there is a single player, so that Θ is that player's type space. Suppose that a player reports the value of $\Theta = \{1, \dots, 8\}$. Communication is represented by a binary tree with 8 leaves, as in Figure 1.

There are different trees that can be used to communicate θ. Each tree can be represented by a binary tree with leaves Θ, and so the set of possible communication protocols is the set $\mathscr{T}(\Theta)$ of such binary trees. For $T \in \mathscr{T}(\Theta)$, let $\delta(\theta, T)$ be the depth of leaf θ in T

If we measure complexity by maximum path length ("worst-case", or $\max_{\theta\in\Theta} \delta(\theta, T)$), then complexity is minimized by a balanced binary tree (left side of Figure 1). If instead we measure complexity by the expected path length, $\sum_{\theta\in\Theta} p(\theta)\delta(\theta, T)$. then for non-uniform distribution on the set of types, we may want to use an imbalanced tree (such as the one on the right side of Figure 1), such that the most likely types require shorter messages.

4. The advantage of interactive communication

In a direct mechanism, all players simultaneously report their entire types to a central authority. There is none of the back-and-forth dialogue that exists in organizations.

It is easy to illustrate the advantage of dialogue when we want to reduce communication complexity. For example, suppose that there are two players, 1 and 2. The players must choose which of two paths to go down to find a hidden treasure, *Left* or *Right*. Player 2 has two clues, a blue card and a red card, each of which has as the name of a path. Player 1 knows which of the two cards has the name of the path with the treasure. Thus, $\Theta_1 = \{Blue, Red\}$ and $\Theta_2 = \{Left, Right\}^2$. If the two players simultaneously announce their messages, then to reveal enough information to determine the correct path each must reveal its type, which amounts to one bit of information from player 1 and two bits from player 2. One bit can be economized by having player 1 first announce his information—which of player 2's clues contains the name of the path—and then player 2 announces only the name of the path written on that clue.

5. Incentives and complexity

The main purpose of this note is to consider the interaction between communication complexity and incentive compatibility.

What would it mean for there to be no interaction? This would be true if

1. an economist can examine a social choice function and determine whether it is incentive compatible (in some sense) without considering the details of a communication protocol, and then
2. a computer scientist can design a minimum-complexity communication protocol (with some restrictions) that realizes the social choice function, without worrying about incentive constraints.

The separation we are looking for is a converse of the Revelation Principle, as illustrated in Figure 2 and now explained. Fix a s.c.f. f.

- Let (Γ, σ) be a protocol that computes f.
- Let (Γ^D, σ^D) be the direct mechanism with truthful revelation that computes f.

Given some notion of incentive compatibility, the Revelation Principle says that (Γ^D, σ^D) is IC if (Γ, σ) is IC. Suppose that a converse holds: If (Γ^D, σ^D) is IC then (Γ, σ) is IC compatible. Then we get the separation alluded to above. The economist checks the IC constraints of f by looking at a direct mechanism with truthful revelation. The computer

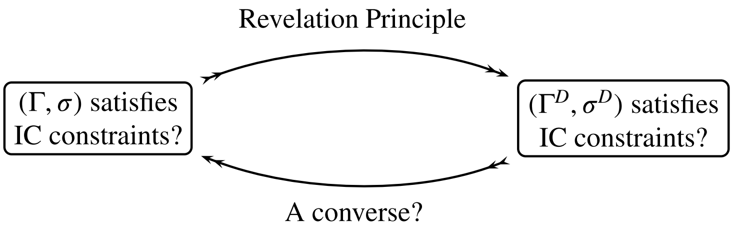

FIGURE 2. "Separation" is a converse of the Revelation Principle.

scientists then designs the communication-efficient protocol. According to the converse, whichever protocol the computer scientists comes up with will satisfy the IC constraints.

A further consequence of the separation is that incentive constraints do not increase the communication complexity of computing a given s.c.f. f. The IC constraints merely limit the set of feasible social choice functions.

6. Counter example: Dominant strategy implementation

We start with a counterexample, which helps illustrate the above.

Consider a private-values environment and suppose that we are interested in dominant-strategy implementation. That is, the incentive constraint that the economist imposes on a protocol is that each player's plan of action for each of the player's types be a dominant strategy for the player in the extensive-form game.

Suppose there are two players, 1 and 2. Suppose that $\Theta_1 = \Theta_2 = \{H, T\}$ and $X = \{X_{HH}, X_{HT}, X_{TH}, X_{TT}\}$. Let f be the s.c.f. such that $f(\theta_1, \theta_2) = X_{\theta_1\theta_2}$. Suppose that player 1's preferences are as follows:

- When $\theta_1 = H$: $X_{HT} \succ_{1H} X_{TT} \succ_{1H} X_{HH} \succ_{1H} X_{TH}$.
- When $\theta_1 = T$: $X_{TH} \succ_{1T} X_{HH} \succ_{1T} X_{TT} \succ_{1T} X_{TH}$.

Then f is DS incentive compatible. The DS-IC constraints are

- $X_{HT} \succ_{1H} X_{TT}$ and $X_{HH} \succ_{1H} X_{TH}$;
- $X_{TH} \succ_{1T} X_{HH}$ and $X_{TT} \succ_{1T} X_{TH}$.

Consider the mechanism in Figure 3. The strategy profile "each player chooses his type" yields the s.c.f. f, and it has 2 bits of communication—the minimum. However, for player 1 when of type H, choosing H is not a best response to player's 2 plan of action "H following H and T following T". For player 1, choosing T leads to the outcome X_{TT} whereas choosing H leads to the outcome X_{HH}, yet $X_{TT} \succ_{1H} X_{HH}$.

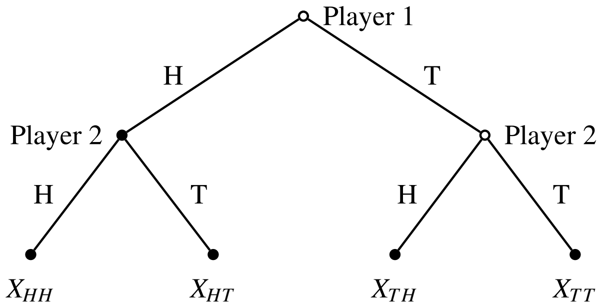

FIGURE 3. A counterexample for dominant-strategy implementation.

7. Ex-post implementation

For direct mechanisms in a private-values setting, ex-post Nash equilibrium is the same as dominant-strategy equilibrium. This is not true for indirect mechanisms.

In games of complete information, dominant-strategy equilibrium takes care of a robustness consideration: it does not require the coordination that leads to common knowledge of expectations or of equilibrium strategies. In games of incomplete information, dominant-strategy equilibrium takes care of another robustness consideration: that beliefs may not be common knowledge or, in the case of mechanism design, may not be known to the mechanism-designer. Ex-post Nash equilibrium takes care only of this second one: in contrast to Bayesian Nash equilibrium, it is belief-free.

Consider, in our previous example, the plan of action by player 2 to which player 1's strategy was not a best response. That plan of action was not chosen by any type of player 2 given the protocol, yet dominant-strategy equilibrium requires that 1's action be a best response to it. In contrast, in ex-post Nash, we only require that player 1's plan of action be a best response to those plans of action chosen by some type of player 2. This is true in the previous example.

Let's review what we are trying to show. Ex-post Nash equilibrium in a direct mechanism means that, for any type of player 2, player 1's payoff from choosing his type is as high as his payoff from reporting a different type. In ex-post Nash equilibrium in a general mechanism, the condition is that, for the plan of action of any type of player 2, player 1's payoff from choosing the plan of action of his type is as high as choosing any other plan of action. The latter follows from the former as long as "choosing any other plan of action" is the same, in terms of outcomes, as "choosing the plan of action of some other type of player 1". That is, as long as deviating in the indirect mechanism means acting as if one were of a different type.

This is true (we will show) as long as all leaves in the extensive game form of the protocol are reached for some profile of types. We call a protocol with this property *parsimonious*.

DEFINITION 1. A protocol is *parsimonious* if every leaf is reached for some state.

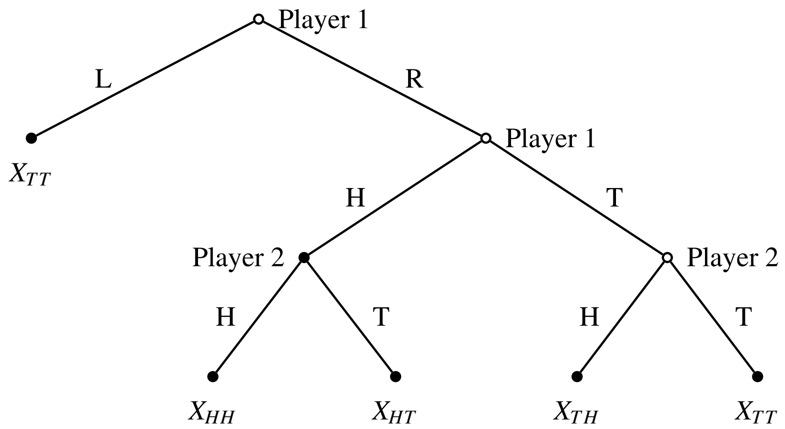

FIGURE 4. A non-parsimonious protocol.

We illustrate it with a counterexample, which is a modification of our previous example. The extensive game form is shown in Figure 4. The strategies are the following.

- Player 1. Type H: R then H. Type T: R then T.
- Player 2. Type H: Always H. Type T: Always T.

This protocol yields the same social choice function as in the previous example. It is not parsimonious because the left-most leaf is not reached for any profile of types. It does not satisfy the ex-post Nash IC constraints: for when player 1 is type H and player 2 is type H, choosing L (leading to outcome X_{TT}) is better than following the protocol (leading to outcome X_{HH}).

A non-parsimonious protocol can always be "pruned" to yield a parsimonious protocol that yields the same social choice function. First eliminate all unreached edges (and the final node of each such edge) in the extensive game form (this can be done in one step, rather than iteratively). This may leave some nodes with a single edge leading from them (each one is part of an information set at which the player has a single possible message and hence does actually not have any choice to make); each such edge can be eliminated, collapsing the two nodes connected by the edge. In a game with perfect recall (to which we restrict attention), such pruning naturally preserves the information structure of the game.

The pruned game is simpler by several measures. First, it may reduce (but never increase) the worst-case communication complexity. Second, it reduces the average communication complexity for any probability measure on the profiles of types. Third, it reduces the complexity of describing the protocol. Thus, a computer scientist designing minimum-complexity protocols would naturally restrict her attention to parsimonious protocols.

We have been using examples with private values, but our separation result for ex-post implementation holds without such an assumption.

PROPOSITION 1. *Suppose a protocol is parsimonious and suppose it realizes a s.c.f. that is ex post implementable. Then the strategy profile is an ex post Nash equilibrium of the mechanism.*

Proof. Let (Γ, σ) be the protocol and let f be the s.c.f. realized by the protocol. Let i be one of the players. The ex-post IC constraints for player i for the s.c.f. f are

$$\forall \theta_i, \theta_{-i}, \theta'_i : \qquad f(\theta_i, \theta_{-i}) \succsim^i_{(\theta_i,\theta_{-i})} f(\theta'_i, \theta_{-i})\,. \tag{1}$$

The ex-post IC constraints for player i for an indirect protocal are

$$\forall \theta_i, \theta_{-i}, s'_i : \qquad \lambda(\sigma_i(\theta_i), \sigma_{-i}(\theta_{-i})) \succsim^i_{(\theta_i,\theta_{-i})} \lambda(s'_i, \sigma_{-i}(\theta_{-i}))\,. \tag{2}$$

.

Let's examine equation (2) for a particular θ, θ_{-i}, and s'_i. Assuming that the indirect protocol computes f, the l.h.s. of equation (2) is $f(\theta_i, \theta_{-i})$. Now consider the r.h.s. Since the protocol is parsimonious, there is some state θ' such that the terminal node $\lambda(s'_i, \sigma_{-i}(\theta_{-i}))$ is reached. That is,

$$\lambda(s'_i, \sigma_{-i}(\theta_{-i})) = \lambda(\sigma_i(\theta'_i), \sigma_{-i}(\theta'_{-i})) \tag{3}$$

If two pairs (s_i, s_{-i}) and (s'_i, s'_{-i}) of plans of action reach the same terminal node, then (s_i, s'_{-i}) and (s'_i, s_{-i}) also reach that node. (Each pair reaches the terminal node if and only if, at each information set on the path to that node, each individual plan of action chooses the action that continues along that path.) Thus,

$$\lambda(\sigma_i(\theta'_i), \sigma_{-i}(\theta'_{-i})) = \lambda(\sigma_i(\theta'_i), \sigma_{-i}(\theta_{-i}))\,. \tag{4}$$

Again, since the indirect mechanism computes f,

$$\lambda(\sigma_i(\theta'_i), \sigma_{-i}(\theta_{-i})) = f(\theta'_i, \theta_{-i})\,. \tag{5}$$

Now use equations (3), (4), and (5) to replace the r.h.s. of equation (2) by $f(\theta'_i, \theta_{-i})$. Remember also that the l.h.s. is $f(\theta_i, \theta_{-i})$. Thus, this IC is equivalent to

$$f(\theta_i, \theta_{-i}) \succsim^i_{(\theta_i,\theta_{-i})} f(\theta'_i, \theta_{-i})\,.$$

This is just one of the IC constraints for the direct mechanism, in equation (1), which are assumed to hold. □

8. Nash and SPE implementation

Recall that the information structure in Nash and SPE implementation is that all agents have the same information. Therefore, without incentive problems, efficient communication is for one agent to announce the desired outcome. Incentive constraints make this impossible. There is no separation and incentive constraints may greatly increase the communication complexity of computing a social choice function. SPE is probably better than Nash, however, because SPE takes advantage of interactive communication.

9. Bayesian mechanism design

Nash and SPE implementation can be viewed as a special case of Bayesian mechanism design (with perfect correlation of types), so we already have examples showing that incentives can increase communication complexity. Fadel and Segal (2006) contains further examples. Fadel and Segal refer to the lack of separation as a need to "hide" information due to the incentive constraints, which thereby raises communication complexity. We leave unexplored in this short note the quantitative impact and whether there are subclasses of problems for which the separation holds.

10. Transferable utility

The separation does not mean that incentives constraints do not increase communication complexity given some objective of the principle. The principle may have economic objectives with indifference classes of social choice functions. Within an indifference class, the different social choice functions may have different communication complexities. Even with separation, the incentive constraints may make infeasible the low-complexity social choice functions, thereby raising the communication cost of achieving a target indifferent class.

This is a rather abstract introduction that frames the following leading and motivating example: when there is transferable (quasilinear) utility and the principle does not care about the transfers.

Specifically, suppose that the outcomes space can be decomposed as $X = \hat{X} \times \mathbb{R}^n$, where $\mathbb{R}^n$ are profiles of transfers. Call $g \colon \Theta \to \hat{X}$ an *allocation*. It is common in such settings that the principle cares only about the allocation. The transfers are merely a means to align incentives. The need to compute transfers at best has no effect on complexity but will typically add to the communication complexity. Hence, we can ask, for a given allocation, how much does communication complexity rise due to the need to compute transfers in order to satisfy incentive constraints? Fadel and Segal (2006) study precisely this question, for ex post Nash implementation.

References

Fadel, Ronald and Ilya Segal (2006). The Communication Cost of Selfishness. Stanford University.

Green, Jerry and Jean-Jacques Laffont (1986). Incentive Theory with Data Compression. In W. Heller, R. Starr, and D. Starett (Eds.), *Uncertainty, Information and Communication (Essays in Honor of K. J. Arrow, v. 3)*. Cambridge: Cambridge University Press.

Green, Jerry and Jean-Jacques Laffont (1987). Limited Communication and Incentive Compatibility. In T. Groves, R. Radner, and S. Reiter (Eds.), *Information, Incentives, and Economic Mechanisms*. Minneapolis: University of Minnesota Press.

Laffont, Jean-Jacques and David Martimort (1998). Collusion and Delegation. *RAND Journal of Economics*, 29, 280–305.

Melumad, Nahum, Dilip Mookherjee, and Stefan Reichelstein (1992). A Theory of Responsibility Centers. *Journal of Accounting and Economics*, 15, 445–484.

Melumad, Nahum, Dilip Mookherjee, and Stefan Reichelstein (1997). Contract Complexity, Incentives and the Value of Delegation. *Journal of Economics and Management Strategy*, 6.

Mookherjee, Dilip (2006). Decentralization, Hierarchies, and Incentives: A Mechanism Design Perspective. *Journal of Economic Literature*, 44, 367–390.

Nisan, Noam and Ilya Segal (2006). The Communication Complexity of Efficient Allocation and Supporting Prices. *Journal of Economic Theory*, 129, 192–224.

Poitevin, Michel (2000). Can the Theory of Incentives Explain Decentralization? *The Canadian Journal of Economics*, 33, 878–906.