ROBUST IMPLEMENTATION: THE ROLE OF LARGE SPACES

By

Dirk Bergemann and Stephen Morris

June 2005

COWLES FOUNDATION DISCUSSION PAPER NO. 1519

COWLES FOUNDATION FOR RESEARCH IN ECONOMICS
YALE UNIVERSITY
Box 208281
New Haven, Connecticut 06520-8281

http://cowles.econ.yale.edu/

Robust Implementation: The Role of Large Type Spaces*

Dirk Bergemann† Stephen Morris‡

First Version: March 2003
This Version: June 2005

Abstract

A social choice function is robustly implemented if every equilibrium on every type space achieves outcomes consistent with a social choice function. We identify a *robust monotonicity* condition that is necessary and (with mild extra assumptions) sufficient for robust implementation.

Robust monotonicity is strictly stronger than both Maskin monotonicity (necessary and almost sufficient for complete information implementation) and ex post monotonicity (necessary and almost sufficient for ex post implementation). It is equivalent to Bayesian monotonicity on all type spaces. It requires that there not be too much interdependence of types.

We characterize robust monotonicity for some interesting economic environments. We identify conditions where, if robust implementation is possible, it is possible in a direct mechanism. We identify conditions where, if robust implementation is not possible, virtual robust implementation is not possible either.

KEYWORDS: Mechanism Design, Implementation, Robustness, Common Knowledge, Interim Equilibrium, Iterative Deletion, Dominant Strategies.

JEL CLASSIFICATION: C79, D82

*This research is supported by NSF Grant #SES-0095321. The first author gratefully acknowledges support through a DFG Mercator Research Professorship at the Center of Economic Studies at the University of Munich. Some of the results reported in this paper appeared in early versions of Bergemann and Morris (2004); earlier versions of this paper contained results on ex post implementation which now appear in Bergemann and Morris (2005a); this paper contains an interim report on an ongoing research project.

†Department of Economics, Yale University, 28 Hillhouse Avenue, New Haven, CT 06511, dirk.bergemann@yale.edu.

‡Department of Economics, Yale University, 30 Hillhouse Avenue, New Haven, CT 06511, stephen.morris@yale.edu.

1 Introduction

The mechanism design literature provides a powerful characterization of which social choice functions can be achieved when the designer has incomplete information about agents' types. If we assume a commonly known common prior over the possible types of agents, the revelation principle establishes that if the social choice function can arise as an equilibrium in some mechanism, then it will arise in a truth-telling equilibrium of the direct mechanism (where each agent truthfully reports his type and the designer chooses an outcome assuming they are telling the truth). Thus the Bayesian incentive compatibility constraints characterize whether a social choice function is implementable in this sense.

There are two important limitations of Bayesian incentive compatibility analysis. First, the analysis typically assumes a commonly known common prior over the agents' types. This assumption may be too stringent in practise. In the spirit of the "Wilson doctrine" (Wilson (1987)), we would like implementation results that are *robust* to different assumptions about what players do or do not know about other agents' types. Second, the revelation principle only establishes that the direct mechanism has *an* equilibrium that achieves the social choice function. In general, there may be other equilibria that deliver undesirable outcomes. We would like to achieve *full* implementation, i.e., show the existence of a mechanism all of whose equilibria deliver the social choice function. We studied the first "robustness" problem in an earlier work, Bergemann and Morris (2004). The second "full implementation" problem has been the subject of a large literature. In the incomplete information context, key full implementation references are Postlewaite and Schmeidler (1986), Palfrey and Srivastava (1989) and Jackson (1991). In this paper, we study "robust implementation" where we require robustness and full implementation simultaneously. Requiring both simultaneously adds extra structure to the problem and enables us to derive distinctive new economic results.

Interim implementation on all type spaces is possible if and only if it is possible to implement the social choice function using an iterative deletion procedure. We refer to the resulting notion as *iterative implementation*. We fix a mechanism and iteratively delete messages for each payoff type that are strictly dominated by another message for each payoff type profile and message profile that has survived the procedure. This observation about iterative deletion illustrates a general point well-known from the literature on epistemic foundations of game theory (e.g., Brandenburger and Dekel (1987), Battigalli and Siniscalchi (2003)): equilibrium solution concepts only have bite if we make strong assumptions about type spaces, i.e., we assume small type spaces where the common prior assumption holds.

We exploit this equivalence between robust and iterative implementation to obtain necessary and

sufficient conditions for robust implementation in general environments. Our necessity argument is conceptually novel, exploiting the iterative characterization. The necessary conditions for robust implementation are ex post incentive compatibility of the social choice function and a condition - *robust monotonicity* - that is equivalent to requiring interim monotonicity on every type space. Suppose that we fix a "deception" specifying, for each payoff type θ_i of each agent, a set of types that he might misreport himself to be. We require that for some agent i and a type misreport of agent i under the deception, for every misreport θ'_{-i} that the other agents might make under the deception, there exists an outcome y which is strictly preferred by agent i to the outcome he would receive under the social choice function for *every* possible payoff type profile that might misreport θ'_{-i}; where this outcome y satisfies the extra restriction that no payoff type of agent i prefers outcome y to the social choice function if the other agents were really types θ'_{-i}. This condition - while a little convoluted - is easier to interpret than the interim (Bayesian) monotonicity conditions. It is very strong and implies both Maskin monotonicity and ex post monotonicity conditions (but is strictly weaker than dominant strategies). We will present examples to illustrate the relationship between these monotonicity conditions.

The sufficiency argument requires only a modest strengthening of the necessary condition by guaranteeing the existence of a "bad" outcome. With the existence of a bad outcome, we show that the necessary conditions are also sufficient for robust implementation. The sufficient conditions guarantee robust implementation in pure, but more generally also in mixed strategies. Our robust analysis thus removes the frequent gap between pure and mixed strategy implementation in the literature.

The iterative characterization comes with the additional benefit that tight implementation results can be proved via a fixed point of a contraction mapping. In particular, we consider a general class of interdependent preferences in which the private types of the agents can be linearly aggregated. In this environment we show that the social choice function can be robustly implemented if and only if the interdependence is not too large. If γ is the weight of the type of agent j (relative to the type of agent i) for the utility of agent i, then the robust implementation condition can simply be stated as: $\gamma < 1/(I-1)$, where I is the number of agents. Surprisingly, the converse result for $\gamma > 1/(I-1)$ even extends to a robust version of virtual utility. In other words, we also show that if $\gamma > 1/(I-1)$, then not only robust implementation, but even robust virtual implementation fails. We further illustrate the strength of the contraction mapping idea in the implementation context with two examples, one of a public good and one of a private good (single unit auction) allocation problem with quasilinear utility. An important paper of Chung and Ely (2001) analyzed the single (and multi-unit) auction with interdependent valuations with dominance solvability (elimination of weakly rather than strictly dominated actions). In a linear and symmetric setting, they reported

sufficient conditions for direct implementation that coincide with the ones derived here. We show that in the environment with linear aggregation, under strict incentive compatibility, the basic insight extends from the single unit auction model to general allocations models, with elimination of strictly dominated actions only (thus Chung and Ely (2001) require deletion of weakly dominated strategies only because incentive constraints are weak). We also prove a converse result: if there is too much interdependence, then neither the direct nor any augmented mechanism can robustly implement the social choice function (this result will also hold with deletion of weakly dominated strategies).

In the implementation literature, it is a standard practice to obtain the sufficiency results with augmented mechanisms. By augmenting the direct mechanism with additional messages, the designer may elicit additional information about undesirable equilibrium play by the agents. Yet, in many environments common to applied mechanism designs, such a single crossing or supermodular preferences, the structure of the preferences may already permit direct implementation (see a companion paper, Bergemann and Morris (2005), for direct implementation results in ex post equilibrium). We thus provide necessary and sufficient conditions for robust implementation in the direct mechanism. In the direct mechanism, the agents can alert the designer only by a report of their type. In consequence, the incentive compatibility conditions for the rewards are identical to the truth-telling constraints, and the necessary and sufficient conditions for robust implementation coincide.

The results in this paper concern full implementation. An earlier paper of ours, Bergemann and Morris (2004), addresses the analogous questions of robustness to rich type spaces, but looking at the question of partial implementation, i.e., does there exist a mechanism such that *some* equilibrium implements the social choice function. We showed that ex post (partial) implementation of the social choice function is a necessary and sufficient condition for partial implementation on all type spaces.[1] This paper establishes that an analogous result does *not* hold for full implementation.

In a companion paper, Bergemann and Morris (2005), we therefore investigate the notion of ex post implementation. The necessary and sufficient conditions then straddle the Nash and Bayesian implementation conditions as an ex post equilibrium is a Nash equilibrium at every incomplete information (Bayesian) type profile. However in contrast to the iterative argument pursued here, the basic reasoning in Bergemann and Morris (2005) invokes more traditional equilibrium arguments. By comparing the conditions for ex post and robust implementation, it becomes apparent that robust implementation typically imposes additional constraints on the allocation problem. In Bergemann and Morris (2005), we showed that in single crossing environments, the same single

[1] This result does not extend to social choice correspondences.

crossing conditions which guarantee incentive compatibility also guarantee full implementation. In contrast, in the linear aggregation environment, we show that robust implementation imposes a strict bound on the interdependence of the preferences, which is not required by the truthtelling conditions. The contraction mapping behind the iterative argument directly pointed at the source of the restriction of the interaction term.

In this paper, we follow the classic implementation literature in allowing for arbitrary mechanisms, including modulo and integer games. By allowing for these mechanisms, we are able to make tight connections with the existing implementation literature. Allowing for these badly behaved mechanisms does complicate our analysis: for example, we must allow for transfinite iterated deletion of best responses in our definition of iterative implementation. Of course, when direct implementation turns out to be possible, we do not need badly behaved mechanisms. We also report some results on what happens with "nice" mechanisms where rationalizable messages always exist and best responses are well defined. We can further strengthen our necessary conditions in this case.

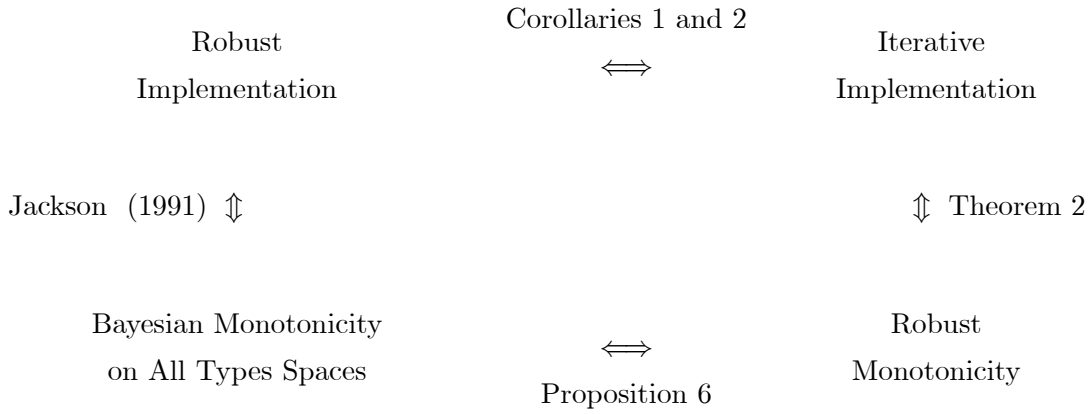

Figure 1: Relationship between Bayesian and Robust Implementation / Monotonicity

Our results extend the classic literature on Bayesian implementation due to Postlewaite and Schmeidler (1986), Palfrey and Srivastava (1989) and Jackson (1991). We focus in this paper on an indirect approach to extending these results. We first note the equivalence between robust implementation and iterative implementation. We then exploit the equivalence to report a direct argument showing that robust monotonicity is a necessary and almost sufficient condition for iterative implementation. But in the light of the classic literature, we know that a necessary and almost sufficient condition for robust implementation must be Bayesian monotonicity on all type spaces. We confirm and clarify our results by directly checking that robust monotonicity is equivalent to

Bayesian (or interim) monotonicity on all type spaces. Figure 1 gives a stylized account of the connection between these alternative approaches.

The remainder of the paper is organized as follows. Section 2 describes the formal environment and solution concepts. Section 3 considers four examples that illustrate the main results of the paper. Section 4 establishes the relation between iterative and robust implementation. Section 5 establishes necessary conditions for robust implementation in the direct mechanism. Section 6 reports our main result on the necessary and sufficient conditions for robust implementation. Section 7 explores the link between robust and virtual implementation. Section 8 considers the preference environment with a linear aggregation of the types and obtains sharp implementation results. Section 9 discusses extensions and variations of our implementation results, examining the role of lotteries, pure strategies and "nice" mechanisms. Section 10 concludes. The Appendix contains some additional examples and proofs.

2 Setup

2.1 The Payoff Environment

We consider a finite set of agents, $1, 2, ..., I$. Agent i's *payoff type* is $\theta_i \in \Theta_i$. We write $\theta \in \Theta = \Theta_1 \times \cdots \times \Theta_I$. There is a set of outcomes Z. We assume that each Θ_i and Z are countable. Each individual has von Neumann Morgenstern utility function $u_i : Z \times \Theta \to \mathbb{R}$. Thus we are in the world of interdependent types, where an agent's utility depends on other agents' payoff types. We allow for lotteries over deterministic outcomes.[2] Let $Y = \Delta(Z)$ and extend u_i to the domain $Y \times \Theta$ in the usual way:

$$u_i(y, \theta) = \sum_{z \in Z} y(z) u_i(z, \theta).$$

A social choice function is a mapping $f : \Theta \to Y$. If the true payoff type profile is θ, the planner would like the outcome to be $f(\theta)$. In this paper, we restrict our analysis to the implementation of a social choice function rather than a social choice correspondence or set.[3]

2.2 Type Spaces

We are interested in analyzing behavior in a variety of type spaces, many of them with a richer set of types than payoff types. For this purpose, we shall refer to agent i's *type* as $t_i \in T_i$, where

[2] The role of the lottery assumption and what happens when we drop it are discussed in Section 9.1.

[3] The extension to social choice correspondences might not be straightforward. One reason is that with social choice correspondences, the incentive compatibility conditions that arise from requiring partial implementation would typically be weaker than ex post incentive compatibility, as shown by examples in Bergemann and Morris (2004).

T_i is a countable set.[4] A type of agent i must include a description of his payoff type. Thus there is a function $\widehat{\theta}_i : T_i \to \Theta_i$ with $\widehat{\theta}_i(t_i)$ being agent i's payoff type when his type is t_i. A type of agent i must also include a description of his beliefs about the types of the other agents; thus there is a function $\widehat{\pi}_i : T_i \to \Delta(T_{-i})$ with $\widehat{\pi}_i(t_i)$ being agent i's *belief type* when his type is t_i. Thus $\widehat{\pi}_i(t_{-i})[t_i]$ is the probability that type t_i of agent i assigns to other agents having types t_{-i}. A *type space* is a collection:

$$\mathcal{T} = \left(T_i, \widehat{\theta}_i, \widehat{\pi}_i\right)_{i=1}^{I}.$$

2.3 Mechanisms

A planner must choose a *game form* or *mechanism* for the agents to play in order the determine the social outcome. Let M_i be the countably infinite set of messages available to agent i.[5] Let $g(m)$ be the distribution over outcomes if action profile m is chosen. Thus a mechanism is a collection

$$\mathcal{M} = (M_1, ..., M_I, g(\cdot)),$$

where $g : M \to Y$.

2.4 Solution Concepts

Now holding fixed the payoff environment, we can combine a type space $\mathcal{T}$ with a mechanism $\mathcal{M}$ to get an incomplete information game $(\mathcal{T}, \mathcal{M})$. The payoff of agent i if message profile m is chosen and type profile t is realized is then given by

$$u_i\left(g(m), \widehat{\theta}(t)\right).$$

A pure strategy for agent i in the incomplete information game $(\mathcal{T}, \mathcal{M})$ is given by

$$s_i : T_i \to M_i.$$

A (behavioral) strategy is given by

$$\sigma_i : T_i \to \Delta(M_i).$$

The objective of this paper is to obtain implementation results for interim, or Bayesian Nash,

[4] The countable types restriction clarifies the relation to the existing literature. In Section 9.3, we discuss what happens if we allow for uncountable type spaces.

[5] This assumption clarifies the relation with the existing literature. We discuss in Section 9.2 what happens if we restrict attention to finite messages.

equilibria on all possible types spaces.[6] The notion of interim equilibrium for a given type space $\mathcal{T}$ is defined in the usual way.

Definition 1 (Interim equilibrium)
A strategy profile $\sigma = (\sigma_1, ..., \sigma_I)$ is an interim equilibrium of the game $(\mathcal{T}, \mathcal{M})$ if, for all i, t_i and m_i with $\sigma_i(m_i|t_i) > 0$,

$$\sum_{t_{-i}\in T_{-i}} \sum_{m_{-i}\in M_{-i}} \left(\prod_{j\neq i} \sigma_j(m_j|t_j)\right) u_i\left(g(m_i, m_{-i}), \widehat{\theta}(t)\right) \widehat{\pi}_i(t_{-i})[t_i]$$
$$\geq \sum_{t_{-i}\in T_{-i}} \sum_{m_{-i}\in M_{-i}} \left(\prod_{j\neq i} \sigma_j(m_j|t_j)\right) u_i\left(g(m_i', m_{-i}), \widehat{\theta}(t)\right) \widehat{\pi}_i(t_{-i})[t_i]$$

for all m_i'.

The concern for robustness, expressed by the qualifying condition, for all type spaces, pulls the interim equilibrium in the direction of rationalizability. Consequently we define a message correspondence profile $S = (S_1, ..., S_I)$, where each $S_i : \Theta_i \to 2^{M_i}$ and we write $\mathcal{S}$ for the collection of message correspondence profiles. The collection $\mathcal{S}$ is a lattice with the natural ordering: $S \leq S'$ if $S_i(\theta_i) \subseteq S_i'(\theta_i)$ for all i and θ_i. The largest element is $\overline{S} = \left(\overline{S}_1, ..., \overline{S}_I\right)$, where $\overline{S}_i(\theta_i) = M_i$ for each i and θ_i. The smallest element is $\underline{S} = (\underline{S}_1, ..., \underline{S}_I)$, where $\underline{S}_i(\theta_i) = \varnothing$ for each i and θ_i.

We define an operator b to iteratively eliminate never best responses. To this end, we denote the belief of agent i over message and payoff type profiles of the remaining agents by $\lambda_i \in \Delta(M_{-i} \times \Theta_{-i})$. The operator $b : \mathcal{S} \to \mathcal{S}$ is now defined as:

$$b_i(S)[\theta_i] = \left\{ m_i \in M_i \;\middle|\; \exists \lambda_i \text{ s.th.:} \begin{array}{ll} (1) & \lambda_i(m_{-i}, \theta_{-i}) > 0 \Rightarrow m_j \in S_j(\theta_j),\ \forall\, j \neq i \\ (2) & \sum\limits_{m_{-i},\theta_{-i}} \lambda_i(m_{-i}, \theta_{-i})\, u_i(g(m_i, m_{-i}), (\theta_i, \theta_{-i})) \\ & \geq \\ & \sum\limits_{m_{-i},\theta_{-i}} \lambda_i(m_{-i}, \theta_{-i})\, u_i(g(m_i, m_{-i}), (\theta_i, \theta_{-i})),\ \forall m_i' \in M_i \end{array} \right\}.$$

We observe that b is increasing by definition: i.e., $S \leq S' \Rightarrow b(S) \leq b(S')$. By Tarski's fixed point theorem, there is a largest fixed point of b, which we label $S^{\mathcal{M}}$. Thus (i) $b\left(S^{\mathcal{M}}\right) = S^{\mathcal{M}}$ and

[6] We label these "interim" equilibria rather than "Bayesian" equilibria in light of the fact that our type space does not have a common prior. Dekel, Fudenberg and Levine (2002) argue that learning justifications that support the equilibrium assumption will - under reasonable assumptions - also imply a common prior on types. We nonetheless maintain the equilibrium solution concept for comparison with the literature. But note that in any case we end up using the fact that with rich type spaces, equilibrium does not have any bite above iterated deletion.

(ii) $b(S) = S \Rightarrow S \leq S^{\mathcal{M}}$. We can also construct the fixed point $S^{\mathcal{M}}$ by starting with $\overline{S}$ - the largest element of the lattice - and iteratively applying the operator b. If the message sets are finite, we have

$$S_i^{\mathcal{M}}(\theta_i) \triangleq \underset{n\geq 1}{\cap} b_i\left(b^n\left(\overline{S}\right)\right)[\theta_i].$$

But because the mechanism $\mathcal{M}$ may be infinite, transfinite induction may be necessary to reach the fixed point.[7] Thus $S_i^{\mathcal{M}}(\theta_i)$ are the set of messages surviving (transfinite) iterated deletion of never best responses. If message sets are finite (or compact), a well known duality argument implies that never best responses are equivalent to strictly dominated actions. However, the equivalence does not hold with infinite (non-compact) message sets.[8] In a compact message analysis, Chung and Ely (2001) consider a version of this solution concept in an incomplete information mechanism design context with dominated (not strictly dominated) messages deleted at each round. We observe that the solution concept defined through the iterative application of the operator b is weaker than the notion of interim rationalizability for a given type space $\mathcal{T}$.[9] Under b, every agent i is allowed to hold arbitrary beliefs about Θ_{-i} and is not restricted to a particular posterior distribution over Θ_{-i}. On the other hand, if the type space $\mathcal{T}$ were the universal type space, then $S_i^{\mathcal{M}}(\theta_i)$ would be equal to the union of all interim rationalizable actions of agent i over all types $t_i \in T_i$ whose payoff type profile coincides with θ_i, or $\widehat{\theta}_i(t_i) = \theta_i$.

For brevity and for lack of a better expression, we refer to the messages $m_i \in S_i^{\mathcal{M}}(\theta_i)$ as *rationalizable actions.*

2.5 Implementation

We now define the notions of interim, robust and iterative implementation.

Definition 2 (Interim Implementation)
Social choice function f is interim implemented on type space $\mathcal{T}$ by mechanism $\mathcal{M}$ if the game

[7] Lipman (1994) contains a formal description of the transfinite induction required (for the case of complete information, but nothing important changes with incomplete information). As he notes "we remove strategies which are never a best reply, taking limits where needed".

[8] The following simple example (suggested to us by Andrew Postlewaite) illustrates the non-equivalence. Players 1 and 2 each choose a non-negative integer, k_1 and k_2 respectively. The payoff to player 1 from $k_1 = 0$ is 1. The payoff to player 1 from action $k_1 \geq 1$ is 2 if $k_1 > k_2$, 0 otherwise. For any belief that player 1 has about 2's actions, there is a (sufficiently high) action from player 1 that gives him a payoff greater than 1. Thus action 0 is never a best response for player 1. However, for any mixed strategy of player 1, there is a (sufficient high) action of player 2 such that action 0 is a better response for player 1 than the mixed strategy. Thus action 0 is not strictly dominated.

[9] For the notion of interim rationalizability, see Battigalli and Siniscalchi (2003) and Dekel, Fudenberg, and Morris (2005).

$(\mathcal{T}, \mathcal{M})$ has an equilibrium and every equilibrium σ of the game $(\mathcal{T}, \mathcal{M})$ satisfies

$$\sigma(m|t) > 0 \Rightarrow g(m) = f\left(\widehat{\theta}(t)\right).$$

We note that a tradition in the implementation literature commonly restricts attention to pure strategy equilibria, but we allow mixed strategy equilibria.[10]

Definition 3 (Robust Implementation)
Social choice function f is robustly implemented by mechanism $\mathcal{M}$ if, for every $\mathcal{T}$, f is interim implemented on type space $\mathcal{T}$ by mechanism $\mathcal{M}$.

We observe that the notion of robust implementation requires that we can find a mechanism $\mathcal{M}$ which implements f for every type space $\mathcal{T}$. A weaker requirement would be to ask that for every type space $\mathcal{T}$ there exists a, possibly different, mechanism $\mathcal{M}$ such that f is implemented. This weaker notion would still lead to the same necessary condition as the stronger implementation version we pursue here, and we believe that it would not lead to a substantial change in the sufficiency conditions either.

We shall establish the necessary and sufficient conditions for robust implementation via the iterative deletion process outlined above.

Definition 4 (Iterative Implementation)
Social choice function f is iteratively implemented by mechanism $\mathcal{M}$ if, for all θ, $S^{\mathcal{M}}(\theta) \neq \varnothing$ and if for all θ and m, $m \in S^{\mathcal{M}}(\theta) \Rightarrow g(m) = f(\theta)$.

3 Examples

We precede the formal results with four examples which are meant to illustrate the main insights of the paper. At the same time, they will facilitate a brief review of the key results in the implementation literature.

The first example is a model of majority rule introduced by Palfrey and Srivastava (1989). It highlights the difficulty of Bayesian implementation in a world of interdependent values. Ex post implementation and virtual implementation on some type spaces are possible, but interim implementation is impossible for some type spaces and thus robust implementation is impossible.

The second example builds around a simple coordination game. It shows that robust implementation, even though it is strong requirement, is substantially weaker than dominant strategy implementation. It also highlights the role of augmented mechanisms in achieving implementation.

[10] We discuss the pure strategy / mixed strategy modelling choice in Section 9.1.

The third example involves the provision of a public good with quasilinear utility. It demonstrates that robust implementation can frequently be achieved in the direct rather than the augmented mechanism. In addition, the example illustrates the relationship between robust implementation and rationalizability. In fact, we obtain a tight condition, in terms of the interdependence for robust implementation. Conversely, if the conditions fails, we show that even robust virtual implementation is impossible.

The fourth and final example investigates a single unit auction with symmetric bidders. The generalized Vickrey-Groves-Clark mechanism only satisfies weak rather than strict incentive compatibility constraints. We therefore propose an ε-efficient allocation rule with strict ex post incentive constraints. The ε efficient allocation rule can also be interpreted as virtual implementation of the efficient rule. In either case, we show that the rule itself can be either robustly implemented or robust virtually implemented, respectively, if there is not too much interdependence among the payoff types.

3.1 Majority Rule

In the first example, introduced by Palfrey and Srivastava (1989), there are three agents and each has two possible "payoff types", θ_a or θ_b. There are two possible choices for society, a or b. All agents have identical preferences. If a majority of agents (i.e., at least two) are of type θ_y, then every agent gets utility 1 from outcome y and utility 0 from the other outcome. The social choice function agrees with the common preferences of the agents. Thus $f : \{\theta_a, \theta_b\}^3 \to \{a, b\}$ satisfies $f(\theta) = y$ if and only if $\#\{i : \theta_i = \theta_y\} \geq 2$. It is useful to use this simple example to review the existing implementation literature and understand the role of interdependent types.

Clearly, incentive compatibility is not a problem in this example. The problem is that in the "direct mechanism" - where all agents simply announce their types - there is the possibility that all agents will choose to always announce θ_a. Since no agent expects to be pivotal, he has no incentive to truthfully announce his type when he is in fact θ_b. What happens if we allow more complicated mechanisms?

If there were complete information about agents' preferences, then the social choice function is clearly implementable: the social planner could pick an agent, say agent 1, and simply follow that agent's recommendation.

But suppose instead that there is incomplete information about agents' preferences. In particular, suppose it is common knowledge that each agent's type is θ_b with independent probability q, with $q^2 > \frac{1}{2}$. This example fails the Bayesian monotonicity condition of Postlewaite and Schmeidler (1986) and Jackson (1991). Palfrey and Srivastava (1989) observe that it is also not possible

to implement in undominated Bayesian Nash equilibrium in this example.

Bergemann and Morris (2005) have analyzed the alternative "more robust" solution concept of ex post equilibrium in this context. It is easy to construct an augmented mechanism whose only ex post equilibrium delivers the social choice function. Let each agent send a message $m_i \in \{\theta_a, \theta_b\} \times \{\text{truth, lie}\}$, with the interpretation that an agent is announcing his own type and also sends the message "truth" if he thinks that others are telling the truth and sends the message "lie" if he thinks that someone is lying. Outcome y is implemented if a majority claim to be type θ_y and all agents announce "truth"; or if either 1 or 3 agents claim to be type θ_y and at least one agent reports lying.

There is a truthtelling ex post equilibrium where each agent truthfully announces his type and also announces "truth". Now suppose there exists an ex post equilibrium such that at some type profile, the desired outcome is not chosen. Note that whatever the announcements of the other agents, each agent always has the ability to determine make the outcome y, by sending the message "lie" and - given the announcements of the other agents - choosing his message so that an odd number of players have claimed to be type θ_y. So this is not consistent with ex post equilibrium.

Serrano and Vohra (2005) show that virtual Bayesian implementation is possible in this example. The idea is that it is possible to exploit common knowledge of q to design a bet as a function of q that will give each player an incentive to truthfully announce his type. This bet may lead to a small but positive probability that the wrong outcome is realized in equilibrium. But virtual implementation is possible.

Robust implementation is impossible in this example. Consider the type space where there is common knowledge that whenever an agent is type θ_y, he assigns probability $\frac{1}{2}$ to both of the other agents being type $y' \neq y$ and probability $\frac{1}{2}$ to one being type y and the other being y'. Thus every type of every player thinks there is a 50% chance that outcome a is better and a 50% chance that b is better. Evidently, there is no way of designing a mechanism that ensures that agents do not fully pool. But if they fully pool, robust implementation is not possible.

3.2 Coordination

This example establishes that although robust implementation is a strong requirement, it is weaker than dominant strategies. There are two agents. Each agent i has two possible types, θ_i and θ_i'. There are six possible outcomes: $Z = \{a, b, c, d, z, z'\}$. The payoffs of the agents are a function of

the allocation and the true payoff type profile, given by:

a	θ_2	θ_2'
θ_1	$3,3$	$0,0$
θ_1'	$0,0$	$1,1$

b	θ_2	θ_2'
θ_1	$0,0$	$3,3$
θ_1'	$1,1$	$0,0$

c	θ_2	θ_2'
θ_1	$0,0$	$1,1$
θ_1'	$3,3$	$0,0$

d	θ_2	θ_2'
θ_1	$1,1$	$0,0$
θ_1'	$0,0$	$3,3$

and

z	θ_2	θ_2'
θ_1	$2,2$	$2,0$
θ_1'	$2,2$	$2,0$

z′	θ_2	θ_2'
θ_1	$2,0$	$2,2$
θ_1'	$2,0$	$2,2$

.

The social choice function is given by the efficient outcome:

f	θ_2	θ_2'
θ_1	a	b
θ_1'	c	d

.

Clearly, the social choice function is strictly ex post incentive compatible. But in the "direct mechanism" where each agent simply reports his type, there will always be an equilibrium where each type of each agent misreports his type, and each agent gets a payoff of 1. This is also strictly ex post incentive compatible. The social choice function f which selects among $\{a, b, c, d\}$ embeds a coordination game. We further observe that the payoff for agent 1 from allocations z and z' are equal and constant for all type profiles. On the other hand, the payoff of agent 2 from z and z' depends on his type but not on the type of the other agent.

We now consider the following augmented simple mechanism:

g	θ_2	θ_2'
θ_1	a	b
θ_1'	c	d
ζ	z	z'

The corresponding incomplete information game has the following payoffs:

	type	θ_2		θ_2'	
type	action	θ_2	θ_2'	θ_2	θ_2'
θ_1	θ_1	$3,3$	$0,0$	$0,0$	$3,3$
	θ_1'	$0,0$	$1,1$	$1,1$	$0,0$
	ζ	$2,2$	$2,0$	$2,0$	$2,2$
θ_1'	θ_1	$0,0$	$1,1$	$1,1$	$0,0$
	θ_1'	$3,3$	$0,0$	$0,0$	$3,3$
	ζ	$2,2$	$2,0$	$2,0$	$2,2$

Suppose we iteratively remove actions for each type that could never be a best response given the type action profiles remaining. Thus in the first round, we would observe that type θ_1 would never send message θ_1' and type θ_1' would never send message θ_1. Knowing this, we could conclude that type θ_2 would never send message θ_2' and type θ_2' would never send message θ_2. This in turn implies that neither type of agent 1 will ever send message ζ. Thus the only remaining message for each type of each agent is truth-telling. But now they must behave this way in any equilibrium on any type space.

3.3 Public Good

The third example describes the provision of a public good with quasilinear utility. The utility of each agent is given by:

$$u_i(\theta, x, y) = \left(\theta_i + \gamma \sum_{j \neq i} \theta_j\right) x + y_i,$$

where x is the level of public good provided and y_i is the monetary transfer to agent i. The utility of agent i depends on his own type $\theta_i \in [0,1]$ and the type profile of other agents, with $\gamma \geq 0$. The cost of establishing the public good is given by $c(x) = \frac{1}{2}x^2$. The planner must choose $(x, y_1, ..., y_I) \in \mathbb{R}_+ \times \mathbb{R}^I$ to maximize social welfare, i.e., the sum of gross utilities minus the cost of the public good:

$$\left((1 + \gamma(I-1)) \sum_{i=1}^{I} \theta_i\right) x - \frac{1}{2}x^2.$$

The socially optimal level of the public good is therefore equal to

$$f_0(\theta) = (1 + \gamma(I-1)) \sum_{i=1}^{I} \theta_i.$$

We choose essentially unique (up to a constant) transfers that give rise to ex post incentive compatibility:

$$f_i(\theta) = -(1 + \gamma(I-1)) \left(\gamma \left(\sum_{j \neq i} \theta_j\right) \theta_i + \frac{1}{2}\theta_i^2\right).$$

We first argue that if $\gamma < \frac{1}{I-1}$, the social choice function f is robustly implementable in the *direct mechanism* where each agent reports his payoff type θ_i and the planner chooses outcomes according to f on the assumption that agents are telling the truth. Consider an iterative deletion procedure. Let $S^0(\theta_i) = [0,1]$ and, for each $k = 1, 2, ..,$ let $S^k(\theta_i)$ be the set of reports that agent i might send, for some conjecture over his opponents' types and reports, with the only restriction on his conjecture being that each type θ_j of agent j sends a message in $S^{k-1}(\theta_j)$.

Suppose that agent i has payoff type θ_i, has a point conjecture that other agents have type profile θ_{-i} and report their types to be θ'_{-i}, and he reports himself to be type θ'_i. Then his expected payoff is a constant $(1+\gamma(I-1))$ times

$$\left(\theta_i+\gamma\sum_{j\neq i}\theta_j\right)\left(\theta'_i+\sum_{j\neq i}\theta'_j\right)-\left(\gamma\left(\sum_{j\neq i}\theta'_j\right)\theta'_i+\frac{1}{2}\left(\theta'_i\right)^2\right).$$

The first order condition with respect to θ'_i is then

$$\theta_i+\gamma\sum_{j\neq i}\theta_j-\gamma\left(\sum_{j\neq i}\theta'_j\right)-\theta'_i=0,$$

so he would wish to set

$$\theta'_i=\theta_i+\gamma\sum_{j\neq i}\left(\theta_j-\theta'_j\right).$$

Note that this calculation verifies the strict ex post incentive compatibility of f. The quadratic payoff / linear best response nature of this problem means that we can characterize $S^k(\theta_i)$ restricting attention to such point conjectures. In particular, we will have

$$S^k(\theta_i)=\left[\underline{\beta}^k(\theta_i),\overline{\beta}^k(\theta_i)\right],$$

where

$$\begin{aligned}\overline{\beta}^k(\theta_i) &= \min\left\{1,\theta_i+\ \gamma\max_{\left\{(\theta'_{-i},\theta_{-i}):\theta'_j\in S^k(\theta_j)\text{ for all }j\neq i\right\}}\sum_{j\neq i}\left(\theta_j-\theta'_j\right)\right\}\\ &= \min\left\{1,\theta_i+\ \gamma\max_{\theta_{-i}}\sum_{j\neq i}\left(\theta_j-\underline{\beta}^{k-1}(\theta_j)\right)\right\}.\end{aligned}$$

Analogously,

$$\underline{\beta}^k(\theta_i)=\max\left\{0,\theta_i-\ \gamma\max_{\theta_{-i}}\sum_{j\neq i}\left(\overline{\beta}^{k-1}(\theta_j)-\theta_j\right)\right\}.$$

Thus

$$\overline{\beta}^k(\theta_i)=\min\left\{1,\theta_i+\left(\gamma(I-1)\right)^k\right\},$$

and

$$\underline{\beta}^k(\theta_i)=\max\left\{0,\theta_i-\left(\gamma(I-1)\right)^k\right\}.$$

Thus $\theta'_i\neq\theta_i\Rightarrow\theta'_i\notin S^k(\theta_i)$ for sufficiently large k, provided that $\gamma<\frac{1}{I-1}$.

On the other hand if $\gamma > \frac{1}{I-1}$, then we argue that there exist type spaces where the social choice function f is not virtually implementable. Consider a type space where it is common knowledge whenever agent i has type θ_i, he is convinced that the types of other players θ_{-i} are such that

$$\sum_{j \neq i} \theta_j = \frac{1}{\gamma}\left(\frac{1}{2} - \theta_i\right).$$

Observe that $\gamma > \frac{1}{I-1}$ implies that we can choose the θ_j to be in the interval $[0,1]$. Agent i's preferences are independent of his type on this type space. Now fix any mechanism and restrict each player to a pooling strategy, i.e., sending the same message independent of his type. Since all types now have identical preferences over outcomes, this pooling strategy is an equilibrium.[11]

3.4 Private Good

The final example is a single unit auction with symmetric bidders. There are I agents and agent i's payoff type is $\theta_i \in [0,1]$. If the profile type profile is θ, agent i's valuation of the object is

$$\theta_i + \gamma \sum_{j \neq i} \theta_j,$$

where $0 \leq \gamma \leq 1$.

In Bergemann and Morris (2005), we showed that Maskin monotonicity fails in this example, so complete information implementation is impossible. This in turn implies that robust implementation is impossible. But Bergemann and Morris (2005) also showed that ex post implementation occurs in the direct mechanism if there are at least three agents.

We next argue that virtual robust implementation is possible in this example if $\gamma < \frac{1}{I-1}$. An allocation rule is a function $x : \Theta \to [0,1]^I$, where $x_i(\theta)$ is the probability that agent i gets the object and so $\sum_i x_i(\theta) \leq 1$. The symmetric efficient allocation rule is the following:

$$x_i^*(\theta) = \begin{cases} \frac{1}{\#\{j : \theta_j \geq \theta_k \text{ for all } k\}}, & \text{if} \quad \theta_i \geq \theta_k \text{ for all } k, \\ 0, & \text{if} \qquad \text{otherwise.} \end{cases}$$

A symmetric ε-efficient allocation rule is the following:

$$x_i^{**}(\theta) = \varepsilon \frac{\theta_i}{I} + (1-\varepsilon)\, x_i^*(\theta).$$

In the Appendix, we verify that the resulting generalized VCG transfers satisfy strict ex post incentive compatibility and show that this ε-efficient allocation is robustly implementable. Under

[11] This argument is complete only if we know that an equilibrium exists in the game where players are restricted to pooling strategies. This will be true for well-behaved mechanisms. Our general negative result holds even if this is not guaranteed.

this allocation rule, the object is not allocated with probability $\frac{\varepsilon}{2}$. At the cost of some additional algebra, we could replace this rule with

$$x_i^{**}(\theta) = \varepsilon \frac{\theta_i}{\sum_j \theta_j} + (1-\varepsilon) x_i^*(\theta)$$

which allocates the object with probability 1.

However, if $\frac{1}{I-1} < \gamma \leq 1$, only constant allocations are robust implementable, by the same argument as in the public good case: we can construct beliefs for each type such that types are indistinguishable.

4 Interim Equilibrium and Iterative Elimination

The notion of robust implementation requires that a social choice function f can be interim implemented for all type spaces $\mathcal{T}$. As we look for necessary and sufficient conditions for robust implementation, conceptually there are (at least) two approaches to obtain the conditions.

One approach would be to simply look at the interim implementation conditions for every possible type space $\mathcal{T}$ and then try to characterize the intersection or union of these conditions for all type spaces. This is the approach we initially pursued, and it works in brute force kind of way. In Section 9.1, we review what happens under this approach.

But we focus our analysis on a second, more elegant, approach. We first establish an equivalence between robust and iterative implementation and then derive the necessary conditions for robust implementation as an implication of iterative implementation. The advantage of the second approach is that after establishing the equivalence, we do not need to argue in terms of large type spaces, but rather derive the results from a novel argument using the iterative elimination process.

A complicating element in the implementation context is the fact that the augmented mechanisms often have infinite message spaces and that best responses may not exist. These complications are inherent to the entire implementation literature and we therefore have to carefully address these issues before we establish the implementation results.

4.1 Best Response

We start with the fixed point $S^{\mathcal{M}}$ of the iterative elimination procedure. Recall that by definition, $S^{\mathcal{M}}$ is a fixed point of b, and thus for all $m_i \in S_i^{\mathcal{M}}(\theta_i)$, there exists $\lambda_i \in \Delta(M_{-i} \times \Theta_{-i})$ such that

$$\begin{aligned} &(1)\ \lambda_i(m_{-i}, \theta_{-i}) > 0 \Rightarrow m_j \in S_j^{\mathcal{M}}(\theta_j) \text{ for each } j \neq i \\ &(2)\ \sum_{m_{-i}, \theta_{-i}} \lambda_i(m_{-i}, \theta_{-i}) \begin{bmatrix} u_i(g(m_i, m_{-i}), (\theta_i, \theta_{-i})) \\ -u_i(g(m_i', m_{-i}), (\theta_i, \theta_{-i})) \end{bmatrix} \geq 0 \text{ for all } m_i' \in M_i \end{aligned} \tag{1}$$

Proposition 1 (Rationalizable Actions)
$m_i \in S^{\mathcal{M}}(\theta_i)$ if and only if there exists a type space $\mathcal{T}$, an interim equilibrium σ of the game $(\mathcal{T}, \mathcal{M})$ and a type $t_i \in T_i$ such that (i) $\sigma_i(m_i|t_i) > 0$ and (ii) $\widehat{\theta}_i(t_i) = \theta_i$.

Proof. ($\Rightarrow$) Suppose $m_i^* \in S^{\mathcal{M}}(\theta_i^*)$. Now consider the following type space $\mathcal{T}$ defined through:

$$T_i = \left\{(m_i, \theta_i) : m_i \in S_i^{\mathcal{M}}(\theta_i)\right\}.$$

Let

$$\widehat{\theta}_i(m_i, \theta_i) \triangleq \theta_i.$$

By (1), we know that for each $m_i \in S_i^{\mathcal{M}}(\theta_i)$, there exists $\lambda_i^{m_i,\theta_i} \in \Delta(M_{-i} \times \Theta_{-i})$ such that:

$$\lambda_i^{m_i,\theta_i}(m_{-i}, \theta_{-i}) > 0 \Rightarrow m_j \in S_j^{\mathcal{M}}(\theta_j) \text{ for each } j \neq i;$$

and

$$\sum_{m_{-i},\theta_{-i}} \lambda_i^{m_i,\theta_i}(m_{-i}, \theta_{-i}) \left[u_i(g(m_i, m_{-i}), (\theta_i, \theta_{-i})) - u_i\left(g\left(m_i', m_{-i}\right), (\theta_i, \theta_{-i})\right)\right] \geq 0, \ \forall m_i' \in M_i.$$

Let

$$\widehat{\pi}_i(m_{-i}, \theta_{-i})[m_i, \theta_i] \triangleq \lambda_i^{m_i,\theta_i}(m_{-i}, \theta_{-i}).$$

Now by construction, there is a pure strategy equilibrium s with $s_i(m_i, \theta_i) = m_i$. But now $s_i(m_i^*, \theta_i^*) = m_i^*$ and $\widehat{\theta}(m_i^*, \theta_i^*) = \theta_i^*$.

($\Leftarrow$) Suppose there exists a type space $\mathcal{T}$, an equilibrium σ of $(\mathcal{T}, \mathcal{M})$, and $m_i^* \in M_i$ and $t_i^* \in T_i$ such that $\sigma_i(m_i^*|t_i^*) > 0$ and $\widehat{\theta}_i(t_i^*) = \theta_i^*$. Let

$$S_i(\theta_i) = \left\{m_i : \sigma_i(m_i|t_i) > 0 \text{ and } \widehat{\theta}_i(t_i) = \theta_i \text{ for some } t_i \in T_i\right\}.$$

Now interim equilibrium conditions ensure that $b(S) \geq S$. Thus $S \leq S^{\mathcal{M}}$. Thus $m_i^* \in S_i^{\mathcal{M}}\left(\widehat{\theta}_i(t_i^*)\right)$, which concludes the proof. ■

Brandenburger and Dekel (1987) showed an equivalence for finite action complete information games between the set of actions surviving iterated deletion of strictly dominant actions and the set of actions that could be played in a subjective correlated equilibrium. Proposition 1 is a straightforward generalization of Brandenburger and Dekel (1987) to incomplete information and infinite actions. The infinite action extension (for complete information) was shown in Lipman (1994). The finite action incomplete information extension is reported in a recent paper of Battigalli and Siniscalchi (2003) (following an earlier analysis in Battigalli (1999)).

Notice that there is no guarantee that $S^{\mathcal{M}}(\theta_i)$ is non-empty or that the game $(\mathcal{T}, \mathcal{M})$ has an equilibrium: the Proposition holds vacuously in this case. But for implementation results, we care

about existence. We have the following two conditions that relate existence of equilibrium on all type spaces to the actions surviving iterated deletion.

Definition 5 (Ex Post Best Response)
Message correspondence S satisfies the ex post best response property if, for all i and $\theta_i \in \Theta_i$, there exists $m_i^* \in S_i(\theta_i)$ such that

$$m_i^* \in \arg\max_{m_i \in M_i} u_i\left(g\left(m_i, m_{-i}\right), \left(\theta_i, \theta_{-i}\right)\right),$$

for all θ_{-i} and $m_{-i} \in S(\theta_{-i})$.

We observe that for S to satisfy the ex post best response property, S_i must be non-empty for all i and all θ_i.

Definition 6 (Interim Best Response)
Message correspondence S satisfies the interim best response property if, for all i and $\psi_i \in \Delta(\Theta_{-i})$, there exists $\lambda_i \in \Delta(M_{-i} \times \Theta_{-i})$ such that:

1. *$\lambda_i(m_{-i}, \theta_{-i}) > 0 \Rightarrow m_j \in S_j(\theta_j)$ for each $j \neq i$;*
2. *for all $\theta_{-i} \in \Theta_{-i}$:*
$$\sum_{m_{-i} \in M_{-i}} \lambda_i\left(m_{-i}, \theta_{-i}\right) = \psi_i\left(\theta_{-i}\right);$$
3. *for all $\theta_i \in \Theta_i$ there exists $m_i^* \in S_i(\theta_i)$ such that*
$$m_i^* \in \arg\max_{m_i \in M_i} \sum_{m_{-i}, \theta_{-i}} \lambda_i\left(m_{-i}, \theta_{-i}\right) u_i\left(g\left(m_i, m_{-i}\right), \left(\theta_i, \theta_{-i}\right)\right).$$

The interim best response property only requires that for every conjecture over payoff type spaces, there exists some beliefs over messages consistent with the message correspondence S, such that a best response is in the message correspondence. In particular, it does not require that a best response exists for all possible beliefs over message profiles. Note that the ex post best response property is a stronger requirement than the interim best response property. Also note that the interim best response property implies that $S_i^{\mathcal{M}}(\theta_i)$ is non-empty for all i and θ_i.

Proposition 1 links every action profile in the set of rationalizable actions to an equilibrium action for some type space $\mathcal{T}$. Proposition 2 strengthens the relationship between rationalizable and equilibrium actions, after imposing some structure on the best response property of rationalizable and equilibrium actions, respectively.

Proposition 2 (Best Response Properties)

1. *If $S^{\mathcal{M}}$ satisfies the ex post best response property, then $(\mathcal{T}, \mathcal{M})$ has an equilibrium for each $\mathcal{T}$.*

2. *If $(\mathcal{T}, \mathcal{M})$ has an equilibrium for each $\mathcal{T}$, then $S^{\mathcal{M}}$ satisfies the interim best response property.*

Proof. (1.) By the ex post best response property, there exists, for each i, $s_i^* : \Theta_i \to M_i$ such that

$$s_i^*(\theta_i) \in \arg\max_{m_i \in M_i} u_i\left(g\left(m_i, s_{-i}^*(\theta_{-i})\right), (\theta_i, \theta_{-i})\right)$$

for all θ_{-i}. Now fix any type space. The strategy profile s with

$$s_i(t_i) = s_i^*\left(\widehat{\theta}_i(t_i)\right)$$

is an equilibrium of the game $(\mathcal{T}, \mathcal{M})$.

(2.) Suppose $(\mathcal{T}, \mathcal{M})$ has an equilibrium for each $\mathcal{T}$. Fix any i and $\psi_i \in \Delta(\Theta_{-i})$. Fix any type space $\mathcal{T}$ with, for each $\theta_i \in \Theta_i$, a type $t_i^*(\theta_i)$ such that (a) $\widehat{\theta}_i(t_i^*(\theta_i)) = \theta_i$ for each θ_i, (b) $\widehat{\pi}_i(t_i^*(\theta_i)) = \pi_i$ for all θ_i and (c)

$$\sum_{\left\{t_{-i}:\widehat{\theta}_{-i}(t_{-i})=\theta_{-i}\right\}} \pi_i(t_{-i})[t_i^*] = \psi_i(\theta_{-i}) \tag{2}$$

for all θ_i and θ_{-i}. The game has an equilibrium σ. Let m_i be any message with $\sigma_i(m_i|t_i^*(\theta_i)) > 0$. Let

$$\lambda_i(m_{-i}, \theta_{-i}) = \sum_{\left\{t_{-i}\in T_{-i}:\widehat{\theta}_{-i}(t_{-i})=\theta_{-i}\right\}} \sigma_{-i}(m_{-i}|t_{-i})\,\pi_i(t_{-i})[t_i^*].$$

Now $\sigma_i(m_i|t_i^*(\theta_i)) > 0$ implies

$$m_i(\theta_i) \in \arg\max_{m_i \in M_i} \sum_{m_{-i},\theta_{-i}} \lambda_i(m_{-i}, \theta_{-i})\, u_i(g(m_i, m_{-i}), (\theta_i, \theta_{-i})).$$

Proposition 1 implies that every message profile m_j which is played in equilibrium by type θ_j is part of the set $S^{\mathcal{M}}$, or that:

$$\lambda_i(m_{-i}, \theta_{-i}) > 0 \Rightarrow m_j \in S_j^{\mathcal{M}}(\theta_j) \text{ for each } j \neq i.$$

By construction of the type space $\mathcal{T}$, in particular property (c) as expressed by (2), this implies that

$$\sum_{m_{-i}\in M_{-i}} \lambda_i(m_{-i}, \theta_{-i}) = \psi_i(\theta_{-i}) \text{ for all } \theta_{-i} \in \Theta_{-i}.$$

Since these properties hold for arbitrary i and $\psi_i \in \Delta(\Theta_{-i})$, $S^{\mathcal{M}}$ satisfies the interim best response property, which concludes the proof. ■

It is unfortunate that there is a gap between the necessary and sufficient conditions in this Proposition. However, an example in the Appendix shows that it is possible to construct (admittedly silly) mechanisms where $(\mathcal{T}, \mathcal{M})$ has an equilibrium for each $\mathcal{T}$, but $S^{\mathcal{M}}$ fails the ex post best response property. The ex post best response property must be satisfied if the mechanism is nice, i.e., best responses always exist.

4.2 Implementation

The first part of the definition of implementation is the requirement that all outcomes that occur in equilibrium are consistent with the social choice function. The first definition checks if this is true for some fixed type space $\mathcal{T}$ and mechanism $\mathcal{M}$.

Definition 7 (Interim Material Implementation)
Social choice function f is interim materially implemented on type space $\mathcal{T}$ by mechanism $\mathcal{M}$ if every equilibrium σ of the game $(\mathcal{T}, \mathcal{M})$ satisfies

$$\sigma(m|t) > 0 \Rightarrow g(m) = f\left(\widehat{\theta}(t)\right),$$

for all t.

The next definition checks if the same property holds for every type space:

Definition 8 (Robust Material Implementation)
Social choice function f is robustly materially implemented by mechanism $\mathcal{M}$ if, for every $\mathcal{T}$, f is interim materially implemented on type space $\mathcal{T}$ by mechanism $\mathcal{M}$.

Finally, we ask if iterated deletion delivers acceptable outcomes:

Definition 9 (Iterative Material Implementation)
Social choice function f is iteratively materially implemented by mechanism $\mathcal{M}$ if, for all θ, $m \in S^{\mathcal{M}}(\theta) \Rightarrow g(m) = f(\theta)$.

Now Proposition 1 immediately implies an equivalence between robust and iterative implementation.

Corollary 1 (Equivalence)
Social choice function f is iteratively materially implemented by $\mathcal{M}$ if and only if f is robustly materially implemented by mechanism $\mathcal{M}$.

In all the above definitions, we qualified that implementation as being "material" because the premise of the definition might be vacuous: the mechanism $\mathcal{M}$ might have the property that on any type space, there is no equilibrium and no messages surviving iterated deletion.[12]

Proposition 2 gave the slightly messier result relating equilibrium existence and properties of messages surviving iterated deletion. The following corollary gives the immediate implications for our implementation definitions:

Corollary 2 (Necessary Conditions)

1. *If social choice function f is iteratively materially implemented by mechanism $\mathcal{M}$ and $S^{\mathcal{M}}$ satisfies the ex post best response property, then f is robustly implemented by $\mathcal{M}$.*

2. *If f is robustly implemented by $\mathcal{M}$, then f is iteratively materially implemented by mechanism $\mathcal{M}$ and $S^{\mathcal{M}}$ satisfies the interim best response property.*

The 'material' qualification will only be used in the necessity part of Theorem 2 where we shall invoke the above Corollary 2.2. There we shall use the fixed-point property of $S^{\mathcal{M}}$, stated earlier in (1), to derive the robust monotonicity condition. In the sufficiency part of the proof, a non-empty set $S^{\mathcal{M}}$ is obtained in the augmented mechanism by virtue of ex post incentive compatibility. Similarly, for the direct implementation results, a non-empty set $S^{\mathcal{M}}$ exists by ex post incentive compatibility and the 'material' qualification is not needed at all. The following implication of robust implementability will be used to establish robust monotonicity in Theorem 2.

Lemma 1 (Truthtelling as Best Response)
If f is iteratively materially implemented by mechanism $\mathcal{M}$ and $S^{\mathcal{M}}$ satisfies the interim best response property, then for all i and $\theta_{-i} \in \Theta_{-i}$, there exists $\nu_i \in \Delta\left(S_{-i}^{\mathcal{M}}(\theta_{-i})\right)$,

$$u_i\left(f\left(\theta_i, \theta_{-i}\right), \left(\theta_i, \theta_{-i}\right)\right) \geq \sum_{m_{-i}} \nu_i\left(m_{-i}\right) u_i\left(g\left(m_i, m_{-i}\right), \left(\theta_i, \theta_{-i}\right)\right) \tag{3}$$

for all $m_i \in M_i$ and $\theta_i \in \Theta_i$.

Proof. Applying the definition of the interim best response property for i and the degenerate distribution putting probability 1 on θ_{-i}, we have that there exists $\nu_i \in \Delta\left(S_{-i}^{\mathcal{M}}(\theta_{-i})\right)$ such that

$$\varnothing \neq \arg\max_{m_i} \sum_{m_{-i}, \theta_{-i}} \nu_i\left(m_{-i}\right) u_i\left(\left(m_i, m_{-i}\right), \left(\theta_i, \theta_{-i}\right)\right) \subseteq S_i^{\mathcal{M}}\left(\theta_i\right) \text{ for all } \theta_i \in \Theta_i.$$

[12] Our terminology mirrors the language of modal logic where proposition A materially implies B whenever A is false, as well as when both A and B are true, see Hughes and Creswell (1996).

But by iterative material implementability, $m \in S^{\mathcal{M}}(\theta) \Rightarrow g(m) = f(\theta)$. So

$$u_i(f(\theta_i, \theta_{-i}), (\theta_i, \theta_{-i})) \geq \sum_{m_{-i}} \nu_i(m_{-i}) u_i((m_i, m_{-i}), (\theta_i, \theta_{-i})),$$

for all $m_i \in M_i$ and $\theta_i \in \Theta_i$. ■

Lemma 1 shows how small the gap between the ex post and interim best response property is. It establishes that truthtelling is a best response against some beliefs over messages m_{-i} for any given payoff type profile θ_{-i}.

5 Direct Implementation

We first characterize when robust direct implementation is possible. By "direct mechanism", we mean the mechanism where each agent simply reports his payoff type θ_i, and so $M_i = \Theta_i$ for all i and $g(\theta) = f(\theta)$ for all θ. Thus we assume that agents do not report their "higher order belief" types. "Truth-telling behavior" is the strategy of always truthfully reporting your payoff type.

Definition 10 (Robust Direct Implementation)
SCF f is robustly directly implementable if truth-telling is an equilibrium of the direct mechanism, and f is interim implementable on any type space $\mathcal{T}$ by the direct mechanism.

A *deception* is a profile $\beta = (\beta_1, .., \beta_I)$, where:

$$\beta_i : \Theta_i \to 2^{\Theta_i} / \varnothing,$$

with $\theta_i \in \beta_i(\theta_i)$ for all i, θ_i.

Definition 11 (Acceptable / Unacceptable Deception)
A deception is acceptable if $\theta' \in \beta(\theta) \Rightarrow f(\theta') = f(\theta)$. A deception is unacceptable if it is not acceptable.

The inverse mapping of the deception β_i represents the set of true type profiles θ_i which could lead to a deception θ_i' and we write

$$\beta_i^{-1}(\theta_i') = \{\theta_i : \theta_i' \in \beta_i(\theta_i)\}.$$

A deception is a message correspondence profile in the special case of a direct mechanism.

Definition 12 (Ex Post Incentive Compatibility)
Social choice function f satisfies ex post incentive compatibility if

$$u_i(f(\theta_i, \theta_{-i}), (\theta_i, \theta_{-i})) \geq u_i(f(\theta_i', \theta_{-i}), (\theta_i, \theta_{-i})),$$

for all i, θ_i, θ_i' and θ_{-i}.

Definition 13 (Direct Robust Monotonicity)
Social choice function f satisfies direct robust monotonicity if, for every unacceptable deception β, there exist i, θ_i, $\theta_i' \in \beta_i(\theta_i)$ such that for all $\theta_{-i}' \in \Theta_{-i}$ and $\rho_i \in \Delta\left(\theta_{-i}', \beta_{-i}^{-1}\left(\theta_{-i}'\right)\right)$, there exists $\theta_i'' \in \Theta_i$ such that

$$\sum_{\theta_{-i} \in \Theta_{-i}} \rho_i\left(\theta_{-i}', \theta_{-i}\right) u_i\left(f\left(\theta_i'', \theta_{-i}'\right), (\theta_i, \theta_{-i})\right) > \sum_{\theta_{-i} \in \Theta_{-i}} \rho_i\left(\theta_{-i}', \theta_{-i}\right) u_i\left(f\left(\theta_i', \theta_{-i}'\right), (\theta_i, \theta_{-i})\right). \quad (4)$$

We will show direct robust monotonicity is a necessary and sufficient condition for implementation in the direct mechanism. In consequence, the designer can only offer those rewards to agent i which can be generated through the social choice function f by a report of agent i, say θ_i''. The focus on rewards which can be generated through type reports in the direct mechanism implies that the incentive compatibility condition for claiming the reward y in the correct circumstances (if and only if the other agents deceive) is automatically satisfied, provided that the social choice function f satisfies ex post incentive compatibility.

Lemma 2 (Robust Implementation as Fixed Point)
f satisfies direct robust monotonicity if and only if $\beta \leq b(\beta) \Rightarrow \beta$ acceptable.

Proof. ($\Rightarrow$) The proof is by contradiction. Thus suppose that f satisfies direct robust monotonicity and that the deception β is a fixed point under the mapping b, but β is unacceptable. In the direct mechanism, the set of messages is $M_i = \Theta_i$ for all i. Then by hypothesis of direct robust monotonicity, there exists i, $\theta_i, \theta_i' \in \beta_i(\theta_i)$ such that for all $\theta_{-i}' \in \Theta_{-i}$ and $\rho_i \in \Delta\left(\theta_{-i}', \beta_{-i}^{-1}\left(\theta_{-i}'\right)\right)$, there exists $\theta_i'' \in \Theta_i$ satisfying the strict inequality (4). But this implies that $\theta_i' \notin b_i(\beta)[\theta_i]$ which contradicts the fixed point property of β.

($\Leftarrow$) The proof is again by contradiction. Thus suppose that the fixed point property $\beta \leq b(\beta)$ indeed implies that β is acceptable, but that f does not satisfy direct robust monotonicity. In other words, let us suppose that there exists an unacceptable deception β for which we cannot find i, θ_i and $\theta_i' \in \beta_i(\theta_i)$ such that the inequality (4) can be satisfied. By hypothesis, β is unacceptable, and it follows that the premise of the hypothesis, namely the fixed point property cannot be satisfied by β. But this implies that there exists i and θ_i' with $\theta_i' \notin b_i(\beta)[\theta_i]$. But the exclusion means that for every $\rho_i \in \Delta\left(\Theta_{-i}' \times \Theta_{-i}\right)$ such that $\rho_i\left(\theta_{-i}', \theta_{-i}\right) > 0 \Rightarrow \theta_j' \in \beta_j(\theta_j)$ for each $j \neq i$, there exists θ_i'' such that

$$\sum_{\theta_{-i}', \theta_{-i}} \rho_i\left(\theta_{-i}', \theta_{-i}\right)\left[u_i\left(f\left(\theta_i'', \theta_{-i}'\right), (\theta_i, \theta_{-i})\right) - u_i\left(f\left(\theta_i', \theta_{-i}'\right), (\theta_i, \theta_{-i})\right)\right] > 0,$$

but this delivers the desired contradiction. ■

Theorem 1 (Robust Direct Implementation)
Social choice function f is robustly directly implementable if and only if f satisfies EPIC and direct robust monotonicity.

Proof. The existence of a truthtelling equilibrium on every type space is equivalent to EPIC (see Bergemann and Morris (2004), Proposition 2). With the existence of a truthtelling equilibrium, verifying robust implementation reduces to verifying iterative implementation by Proposition 1. Iterative implementation requires that the largest fixed point of b is acceptable. But Lemma 2 shows that this is guaranteed by direct robust monotonicity. ■

6 Robust Implementation

A "robust monotonicity" condition is key to our main result. In the direct mechanism, where agents other than i report themselves to be types θ_{-i}, agent i can obtain outcomes $f\left(\theta_i', \theta_{-i}\right)$ for any θ_i'. But once we allow augmented mechanisms, we could conceivably offer agent i a larger set of lotteries if he reports deviant behavior of his opponents. We need to identify, for any given report θ_{-i}, the set of lotteries with the property that whatever agent i's actual type, he would never prefer such an allocation to what he would obtain under the social choice function if other agents were reporting truthfully. Thus:

$$Y_i\left(\theta_{-i}\right) = \left\{y \in Y : u_i\left(y, \left(\theta_i', \theta_{-i}\right)\right) \leq u_i\left(f\left(\theta_i', \theta_{-i}\right), \left(\theta_i', \theta_{-i}\right)\right) \text{ for all } \theta_i' \in \Theta_i\right\}.$$

Henceforth, we refer to the set $Y_i\left(\theta_{-i}\right)$ as the *reward set* (for agent i).

To understand the robust monotonicity condition, it is useful to first think about agents playing the direct mechanism. Suppose that it was common knowledge that in the direct mechanism, type θ_i of player i will send a report $\theta_i' \in \beta_i\left(\theta_i\right)$. If β is acceptable, we would know that f was being implemented. But if β is unacceptable, we must find a type of some agent who is prepared to report that other players are misreporting. But for the "whistle-blower" who is going to report that we are in a bad equilibrium, we cannot know what he believes about the types of the other players, nor can we know what message he expects to hear except that it is a message consistent with the deception. Finally, the reward that he is offered must not mess up the truth-telling behavior in the good equilibrium. This gives the following condition:

Definition 14 (Robust Monotonicity)
Social choice function f satisfies robust monotonicity if for every unacceptable deception β, there exist i, θ_i, $\theta_i' \in \beta_i\left(\theta_i\right)$ such that, for all $\theta_{-i}' \in \Theta_{-i}$ and $\psi_i \in \Delta\left(\beta_{-i}^{-1}\left(\theta_{-i}'\right)\right)$, there exists y such that:

$$\sum_{\theta_{-i} \in \Theta_{-i}} \psi_i\left(\theta_{-i}\right) u_i\left(y, \left(\theta_i, \theta_{-i}\right)\right) > \sum_{\theta_{-i} \in \Theta_{-i}} \psi_i\left(\theta_{-i}\right) u_i\left(f\left(\theta_i', \theta_{-i}'\right), \left(\theta_i, \theta_{-i}\right)\right), \tag{5}$$

while

$$u_i\left(f\left(\theta_i'', \theta'_{-i}\right), \left(\theta_i, \theta'_{-i}\right)\right) \geq u_i\left(y, \left(\theta_i'', \theta'_{-i}\right)\right), \quad \textit{for all } \theta_i'' \in \Theta_i. \tag{6}$$

Notice that this condition has a very similar form to the direct robust monotonicity condition. We simply allow a richer set of rewards in an augmented mechanism. In addition, in the augmented mechanism, agent i can propose an allocation conditional on the misreport θ'_{-i} of the other agents. It therefore suffices that the reward y agent i proposes in the event of report θ'_{-i} is successful for all possible distributions $\psi_i(\theta_{-i})$ over the set of types $\beta_{-i}^{-1}\left(\theta'_{-i}\right)$ which could have reported θ'_{-i} under the deception β_{-i}. In contrast, if we were to seek robust direct implementation, the report of agent i has to lead to rewards which work for all possible misreports θ'_{-i} and underlying true type profiles θ_{-i}. In consequence, the expectation had to be taken both with respect to the reports and the true types, i.e. $\rho\left(\theta_{-i}, \theta'_{-i}\right)$.

If we compare the notion of robust monotonicity with the notion of Bayesian monotonicity for a given type space (see Definition 24), then three major differences appear. First, the notion of a deception is set-valued rather than point-valued. Second, the reward has to be successful against all possible distributions $\psi_i(\theta_{-i})$ over true payoff profile rather than a single distribution, the posterior derived from the common prior. Third, the incentive constraints for the reward, (6), have to be satisfied ex post rather than interim. All three modifications directly stem from the robustness concern. The deception has to be set valued as in a rich type, a given payoff type θ_i can now generate different misreports θ_i' through distinct types t_i who all share the same true payoff type. Similarly, in a rich type space, there might be many distributions over payoff types, $\psi_i(\theta_{-i})$, which adopt a given misreport θ_i'. Finally, the ex post incentive constraints regarding the reward y are necessary to maintain the truthtelling equilibrium in the direct revelation segment of the augmented mechanism. This is the robustness property of the ex post constraints in the direct mechanism developed in Bergemann and Morris (2004).

Robust monotonicity is strictly stronger than both Maskin monotonicity and ex post monotonicity (a necessary and almost sufficient condition for ex post implementation described in Bergemann and Morris (2005)). To get a sense of the strength of the condition, we can return to the examples of Section 3:

1. In the majority rule example, Maskin monotonicity and ex post monotonicity are both satisfied, but robust monotonicity fails.

2. In the coordination example, robust monotonicity (and thus Maskin and ex post monotonicity) are satisfied.

3. In the public goods example, robust monotonicity (and thus Maskin and ex post monotonicity) are satisfied if there is not too much interdependence of preferences. If there is too much interdependence of preferences, robust monotonicity fails but both Maskin and ex post monotonicity are satisfied.

4. In the private good example, ex post monotonicity holds, but Maskin monotonicity (and thus robust monotonicity) fails. However, in a perturbed version of this example, we again have robust monotonicity satisfied if only if there is not too much interdependence of preferences.

Although we will not use it extensively, notice that if M_i and Θ_i are finite, then standard duality arguments imply the following alternative characterization: social choice function f satisfies *robust monotonicity* if for every unacceptable deception β, there exist i, θ_i, $\theta_i' \in \beta_i(\theta_i)$ such that, for all $\theta'_{-i} \in \Theta_{-i}$, there exists $y \in Y_i\left(\theta'_{-i}\right)$ such that:

$$u_i(y,(\theta_i,\theta_{-i})) > u_i\left(f\left(\theta_i',\theta'_{-i}\right),(\theta_i,\theta_{-i})\right),$$

for all θ_{-i} such that $\theta'_{-i} \in \beta_{-i}(\theta_{-i})$. Note that this characterization uses the requirement that Y is a lottery space.

Finally, we need an extremely weak economic condition to ensure that it is always possible to reward and punish each player independently of the others.

Definition 15 (Bad Outcome)
The bad outcome property is satisfied if, for each i, there exists $\underline{y}_i \in Y$ such that, for all $\theta_i \in \Theta_i$, $\psi_i \in \Delta(\Theta_{-i})$ and $\theta'_{-i} \in \Theta_{-i}$, there exists $y \in Y_i\left(\theta'_{-i}\right)$ such that

$$\sum_{\theta_{-i}} \psi_i(\theta_{-i})\, u_i(y,(\theta_i,\theta_{-i})) > \sum_{\theta_{-i}} \psi_i(\theta_{-i})\, u_i\left(\underline{y}_i,(\theta_i,\theta_{-i})\right).$$

This property says that there exists a bad outcome $\underline{y}_i$ for each player i such that we can always offer him a lottery that makes him better off whatever his beliefs about other players' types and whatever reports other players are making. The bad outcome property, together with the use of lotteries, allows us to dispense with any no veto property which typically appear in the sufficient conditions. In addition, we can omit the usual cardinality assumption of $I \geq 3$.

Theorem 2 (Robust Implementation)

1. *If f is robustly implementable, then f satisfies EPIC and robust monotonicity;*

2. *if f satisfies EPIC, robust monotonicity and the bad outcome property, then f is robustly implementable.*

Proof. (1.) We first prove that robust implementability implies EPIC and robust monotonicity. We do so by appealing to the necessary conditions for robust implementation in Corollary 2.

We first establish EPIC. By Lemma 1, there exists $\nu_i \in \Delta\left(S_{-i}^{\mathcal{M}}\left(\theta_{-i}\right)\right)$,

$$u_i\left(f\left(\theta_i,\theta_{-i}\right),\left(\theta_i,\theta_{-i}\right)\right) \geq \sum_{m_{-i}} \nu_i\left(m_{-i}\right) u_i\left(g\left(m_i,m_{-i}\right),\left(\theta_i,\theta_{-i}\right)\right),$$

for all $m_i \in M_i$ and $\theta_i \in \Theta_i$. If we choose $m_i \in S_i^{\mathcal{M}}\left(\theta_i'\right)$, iterative material implementation implies that $g\left(m_i,m_{-i}\right) = f\left(\theta_i',\theta_{-i}\right)$ for all $m_{-i} \in S_{-i}^{\mathcal{M}}\left(\theta_{-i}\right)$. So

$$u_i\left(f\left(\theta_i,\theta_{-i}\right),\left(\theta_i,\theta_{-i}\right)\right) \geq u_i\left(f\left(\theta_i',\theta_{-i}\right),\left(\theta_i,\theta_{-i}\right)\right),$$

for all $\theta_i' \in \Theta_i$.

We next establish robust monotonicity. Fix an unacceptable deception β and suppose that f is iteratively materially implementable. There must exist a message correspondence profile S such that

$$b\left(S\right) \leq S,$$

and

$$S_i^{\mathcal{M}}\left(\theta_i'\right) \subseteq S_i\left(\theta_i\right), \tag{7}$$

for all i, θ_i and $\theta_i' \in \beta_i\left(\theta_i\right)$; but

$$S_i^{\mathcal{M}}\left(\theta_i'\right) \nsubseteq b_i\left(S\right)\left[\theta_i\right], \tag{8}$$

for all i, θ_i and $\theta_i' \in \beta_i\left(\theta_i\right)$. The existence of such an S can be established constructively. Clearly $\overline{S}$ satisfies (7). Iteratively apply the operator b. By iterative implementation, there exists k (perhaps transfinite) such that:

$$S \triangleq b^k\left(\overline{S}\right) \tag{9}$$

satisfies (8). Thus there exists k such that $b^k\left(\overline{S}\right)$ satisfies (7) and $b^{k+1}\left(\overline{S}\right)$ satisfies (8).

By (8), simply pick

$$\widehat{m}_i \in S_i\left(\theta_i\right) \cap S_i^{\mathcal{M}}\left(\theta_i'\right) \quad \text{and} \quad \widehat{m}_i \notin b_i\left(S\right)\left[\theta_i\right] \cap S_i^{\mathcal{M}}\left(\theta_i'\right).$$

Since message $\widehat{m}_i \notin b_i\left(S\right)\left[\theta_i\right]$, we know that for every $\lambda_i \in \Delta\left(M_{-i} \times \Theta_{-i}\right)$ such that

$$\lambda_i\left(m_{-i},\theta_{-i}\right) > 0 \Rightarrow m_j \in S_j\left(\theta_j\right) \text{ for all } j \neq i,$$

there exists m_i^* such that

$$\sum_{m_{-i},\theta_{-i}} \lambda_i\left(m_{-i},\theta_{-i}\right) u_i\left(g\left(m_i^*,m_{-i}\right),\left(\theta_i,\theta_{-i}\right)\right) > \sum_{m_{-i},\theta_{-i}} \lambda_i\left(m_{-i},\theta_{-i}\right) u_i\left(g\left(\widehat{m}_i,m_{-i}\right),\left(\theta_i,\theta_{-i}\right)\right). \tag{10}$$

Next we identify a particular belief $\lambda_i(m_{-i}, \theta_{-i})$ for which the inequality (10) holds. By (3) in Lemma 1, there exists $\nu_i \in \Delta\left(S_{-i}^{\mathcal{M}}\left(\theta'_{-i}\right)\right)$ such that

$$\sum_{m_{-i}} \nu_i(m_{-i})\, u_i\left(g(m_i, m_{-i}), \left(\theta''_i, \theta'_{-i}\right)\right) \leq u_i\left(f\left(\theta''_i, \theta'_{-i}\right), \left(\theta''_i, \theta'_{-i}\right)\right), \tag{11}$$

for all $m_i \in M_i$ and $\theta''_i \in \Theta_i$. Thus for any $\psi_i \in \Delta(\Theta_{-i})$, we can set

$$\lambda_i(m_{-i}, \theta_{-i}) = \nu_i(m_{-i})\, \psi_i(\theta_{-i}).$$

Applying the above claim (10), there exists m_i^* such that:

$$\sum_{\theta_{-i}, m_{-i}} \psi_i(\theta_{-i})\, \nu_i(m_{-i})\, u_i(g(m_i^*, m_{-i}), (\theta_i, \theta_{-i})) > \sum_{\theta_{-i}, m_{-i}} \psi_i(\theta_{-i})\, \nu_i(m_{-i})\, u_i(g(\widehat{m}_i, m_{-i}), (\theta_i, \theta_{-i})).$$

But $\nu_i(m_{-i}) > 0 \Rightarrow (\widehat{m}_i, m_{-i}) \in S^{\mathcal{M}}\left(\theta'\right)$, so by iterative material implementation:

$$g(\widehat{m}_i, m_{-i}) = f\left(\theta'\right).$$

We also observe that as we defined S to be the set obtained after the $k-th$ iteration of the operator b, see (9), if $\theta'_{-i} \in \beta_{-i}(\theta_{-i})$, then $\nu_i(m_{-i}) > 0 \Rightarrow m_{-i} \in S_{-i}(\theta_{-i})$. Thus for every $\psi_i \in \Delta\left(\beta_{-i}^{-1}\left(\theta'_{-i}\right)\right)$, there exists m_i^* such that

$$\sum_{\theta_{-i}, m_{-i}} \psi_i(\theta_{-i})\, \nu_i(m_{-i})\, u_i(g(m_i^*, m_{-i}), (\theta_i, \theta_{-i})) > \sum_{\theta_{-i}, m_{-i}} \psi_i(\theta_{-i})\, \nu_i(m_{-i})\, u_i\left(f\left(\theta'\right), (\theta_i, \theta_{-i})\right). \tag{12}$$

Now, the inequality (12) essentially establishes guarantees the reward inequality for robust monotonicity. We can complete the argument by letting y be the lottery with

$$y(z) \equiv \sum_{m_{-i}} g(m_i^*, m_{-i})\, \nu_i(m_{-i}).$$

We now have established that for each $\theta'_{-i} \in \beta_{-i}(\theta_{-i})$ and $\psi_i \in \Delta\left(\beta_{-i}^{-1}\left(\theta'_{-i}\right)\right)$, there exists y such that (by (11))

$$u_i\left(y, \left(\theta''_i, \theta'_{-i}\right)\right) \leq u_i\left(f\left(\theta''_i, \theta'_{-i}\right), \left(\theta''_i, \theta'_{-i}\right)\right),$$

for all $\theta''_i \in \Theta_i$, and thus $y \in Y_i\left(\theta'_{-i}\right)$.[13] And by (12) we then have:

$$\sum_{\theta_{-i}} \psi_i(\theta_{-i})\, u_i(y, (\theta_i, \theta_{-i})) > \sum_{\theta_{-i}} \psi_i(\theta_{-i})\, u_i\left(f\left(\theta'\right), (\theta_i, \theta_{-i})\right).$$

[13] Note that this step implies that even if we had restricted attention to mechanisms with deterministic outcomes, our robust monotonicity would only have established that there exists a lottery (not necessarily a deterministic outcome) sufficient to reward a whistle-blower.

(2.) We now prove that EPIC, robust monotonicity and the bad outcome property imply robust implementation. We do so by explicitly constructing the implementing mechanism.

Each agent i sends a message $m_i = \left(m_i^1, m_i^2, m_i^3, m_i^4\right)$, where $m_i^1 \in \Theta_i$, $m_i^2 \in \mathbb{Z}^+$, $m_i^3 : \Theta_{-i} \to \Delta(Y)$ with $m_i^3(\theta_{-i}) \in Y_i(\theta_{-i})$, $m_i^4 \in Y$. The outcome $g(m)$ is determined by the following rules.

Rule 1: If $m_i^2 = 1$ for all i, pick $f\left(m^1\right)$.

Rule 2: If there exists $j \in I$ such that $m_i^2 = 1$ for all $i \neq j$ and $m_j^2 > 1$, then pick $m_j^3\left(m_{-j}^1\right)$ with probability $1 - \frac{1}{m_j^2+1}$ and $\underline{y}_i$ with probability $\frac{1}{m_j^2+1}$.

Rule 3: In all other cases, for each i, with probability $\frac{1}{I}\left(1 - \frac{1}{m_i^2+1}\right)$ pick m_i^4, and with probability $\frac{1}{I}\left(\frac{1}{m_i^2+1}\right)$ pick $\underline{y}_i$.

We first show that it is never a best reply for type θ_i to send a message with $m_i^2 > 1$ (i.e., $m_i \in b_i\left(\overline{S}\right) \Rightarrow m_i^2 = 1$). Suppose that θ_i has conjecture $\lambda_i \in \Delta\left(M_{-i} \times \Theta_{-i}\right)$. We can partition the messages of other players as follows:

$$M_{-i}^*(\theta_{-i}) = \left\{m_{-i} : m_j^2 = 1 \text{ for all } j \neq i \text{ and } m_{-i}^1 = \theta_{-i}\right\},$$

and

$$\widehat{M}_{-i} = \left\{m_{-i} : m_j^2 > 1 \text{ for some } j \neq i\right\}.$$

By the bad outcome property, we know that there exists $m_i^4 \in Y$ such that, if

$$\sum_{m_{-i} \in \widehat{M}_{-i}, \theta_{-i} \in \Theta_{-i}} \lambda_i(m_{-i}, \theta_{-i}) > 0,$$

then

$$\sum_{m_{-i} \in \widehat{M}_{-i}, \theta_{-i} \in \Theta_{-i}} \lambda_i(m_{-i}, \theta_{-i})\, u_i\left(m_i^4, \theta\right) > \sum_{m_{-i} \in \widehat{M}_{-i}, \theta_{-i} \in \Theta_{-i}} \lambda_i(m_{-i}, \theta_{-i})\, u_i\left(\underline{y}_i, \theta\right).$$

And we know that there exists m_i^3 such that, if

$$\sum_{m_{-i} \in M_{-i}^*\left(\theta'_{-i}\right), \theta_{-i} \in \Theta_{-i}} \lambda_i(m_{-i}, \theta_{-i}) > 0,$$

then

$$\sum_{m_{-i} \in M_{-i}^*\left(\theta'_{-i}\right), \theta_{-i} \in \Theta_{-i}} \lambda_i(m_{-i}, \theta_{-i})\, u_i\left(m_i^3\left(\theta'_{-i}\right), \theta\right) > \sum_{m_{-i} \in M_{-i}^*\left(\theta'_{-i}\right), \theta_{-i} \in \Theta_{-i}} \lambda_i(m_{-i}, \theta_{-i})\, u_i\left(\underline{y}_i, \theta\right).$$

Thus if $\left(m_i^1, m_i^2, m_i^3, m_i^4\right)$ with $m_i^2 > 1$ were a best response, then $\left(m_i^1, m_i^2 + 1, m_i^3, m_i^4\right)$ would be an even better response, contradiction.

Now fix any S with $m_i \in S_i(\theta_i) \Rightarrow m_i^2 = 1$. Let

$$\beta_i(\theta_i) = \left\{\theta_i' : \left(\theta_i', 1, m_i^3, m_i^4\right) \in S_i(\theta_i) \text{ for some } \left(m_i^3, m_i^4\right)\right\}.$$

First observe that EPIC implies that $\theta_i \in \beta_i(\theta_i)$. We will argue that if β is not acceptable, then $b(S) \neq S$. By robust monotonicity, we know that there exists i, θ_i, $\theta_i' \in \beta_i(\theta_i)$ such that, for all $\theta_{-i}' \in \Theta_{-i}$ and $\psi_i \in \Delta\left(\beta_{-i}^{-1}\left(\theta_{-i}'\right)\right)$, there exists $y \in Y_i\left(\theta_{-i}'\right)$ such that

$$\sum_{\theta_{-i} \in \Theta_{-i}} \psi_i(\theta_{-i}) u_i(y, (\theta_i, \theta_{-i})) > \sum_{\theta_{-i} \in \Theta_{-i}} \psi_i(\theta_{-i}) u_i\left(f\left(\theta_i', \theta_{-i}'\right), (\theta_i, \theta_{-i})\right).$$

But now for any conjecture $\lambda_i \in \Delta\left(\left\{(m_{-i}, \theta_{-i}) : m_j^2 = 1 \text{ for all } j \neq i\right\}\right)$, there exists m_i^3 (with $m_i^3(\theta_{-i}) \in Y_i(\theta_{-i})$) such that

$$\sum_{m_{-i}, \theta_{-i}} \lambda_i(m_{-i}, \theta_{-i}) u_i\left(m_i^3\left(m_{-i}^1\right), \theta\right) > \sum_{m_{-i}, \theta_{-i}} \lambda_i(m_{-i}, \theta_{-i}) u_i\left(f\left(\theta_i', m_{-i}^1\right), (\theta_i, \theta_{-i})\right).$$

Thus message $\left(\theta_i', 1, m_i^3, m_i^4\right)$ is never a best response for type θ_i.

We conclude that if

$$\beta_i(\theta_i) = \left\{\theta_i' : \left(\theta_i', 1, m_i^3, m_i^4\right) \in S_i^{\mathcal{M}}(\theta_i) \text{ for some } \left(m_i^3, m_i^4\right)\right\},$$

then β is acceptable. Thus f is iteratively materially implemented.

Finally observe that $S^{\mathcal{M}}$ must satisfy the ex post best response property, with type θ_i sending a message of the form $\left(\theta_i', 1, m_i^3, m_i^4\right)$, so robust implementation is possible by Corollary 2. ■

The proof directly uses the link between iterative and robust implementation for the necessity as well as the sufficiency part. We briefly sketch the idea of the necessity part of the proof. If f is robustly implementable, then it is iteratively implementable by Corollary 2. From iterative implementability, we then want to show that f satisfies strict robust monotonicity. We then consider a given *and* unacceptable deception β. We start the process of iterative elimination and stop it at a specific round, denoted by k. This round k is the first round at which we can find an agent i, a true type profile θ_i and a report $\theta_i' \in \beta_i(\theta_i)$, such that a message, denoted by $\widehat{m}_i$, which will survive the process of iterated elimination for type θ_i', fails to survive the k-th round of elimination for type θ_i. We then show that the elimination of message $\widehat{m}_i$ at round k implies that the social choice function f satisfies strict robust monotonicity with respect to the deception β. Briefly, if $\widehat{m}_i$ survives the process of elimination for type θ_i', the message $\widehat{m}_i$ acts in the mechanism so as to report a payoff type θ_i'. If it is eliminated at round k for payoff type θ_i, then this means that for any belief agent i has over the remaining agents, there exists a message m_i^* which leads to an allocation through g which is strictly preferred by agent i when he has a payoff type θ_i. The significance of

round k being the first round for which such an elimination relative to the deception β occurs, is that at this round, there do not yet exist any restrictions about message and payoff type profile regarding the other players deception. The fact then that $\widehat{m}_i$ can be eliminated allows us to use full strength of the elimination argument to establish robust monotonicity. In the context of the proof it is interesting to note that the key step from iterative elimination to robust monotonicity is an argument which involves the early stages of the elimination process rather than the limit of iteration process.

The results of Theorem 2 rely both on allowing lotteries and the bad outcome property. In Section 9.1, both assumptions are discussed and a simple example satisfying EPIC and robust monotonicity but not robustly implementable without lotteries is described.

The novel difficulty in obtaining the necessary results for implementation arise from the robustness requirement. If f is robustly implemented, the mechanism which achieves implementation could be badly behaved with respect to the existence of best responses against all possible beliefs about action and type profiles of the other agents. This difficulty did not appear in the direct robust implementation as direct implementation guaranteed ex post incentive compatibility and inside the direct mechanism the (non-empty) existence of $S^{\mathcal{M}}$. In Section 9.2 we reconsider the robust implementation result by restricting attention to *nice mechanisms*, mechanisms in which best responses always exist for all beliefs over payoff type and message profiles.

Finally, in the Appendix we provide a direct proof that interim monotonicity on all type spaces is equivalent to robust monotonicity. The result, contained in Proposition 6, explicitly constructs a type space to show that if robust monotonicity fails then there also exists a type space $\mathcal{T}$ for which Bayesian monotonicity fails.

7 Virtual Implementation

In this section, we extend our robustness analysis from interim to virtual implementation. This section is more limited in scope than the previous sections. The notion of virtual Bayesian implementation is widely considered to be much more permissive than interim implementation (see Abreu and Matsushima (1992b), Duggan (1997) and Serrano and Vohra (2005)).

We shall define a robust version of virtual implementation and then give a simple condition, type indistinguishability, to obtain an impossibility result for robust virtual implementation. The impossibility result will reappear in Section 8. There we consider an environment in which the payoff types can be linearly aggregated with respect to the impact of the types on the utility of each agent. In this linear environment, we will be able to derive an exact and sharp bound for the possibility of robust implementation. Conversely, if robust implementation fails the bound, then

robust virtual implementation will also be impossible. With respect to virtual implementation, a message of the robustness analysis consequently will be that the difference between interim and virtual implementation shrinks substantially once we impose the robustness requirement on implementation.

Definition 16 (Virtual Implementation)
Social choice function f is ε implementable by mechanism $\mathcal{M}$ on type space $\mathcal{T}$, if there exists an equilibrium of the game $(\mathcal{T}, \mathcal{M})$ and every equilibrium σ of the game $(\mathcal{T}, \mathcal{M})$ satisfies

$$\sum_{m \in M} g\left(f\left(\widehat{\theta}\left(t\right)\right)|m\right)\sigma\left(m|t\right) \geq 1-\varepsilon.$$

Social choice function f is virtually implementable if it is $\varepsilon-$implementable for all $\varepsilon > 0$.

We denote by $g\left(y\,|m\right)$ the probability that the outcome y is realized conditional on receiving message m. We next present the robust version of virtual implementation.

Definition 17 (Virtual Robust Implementation)
Social choice function f is $\varepsilon-$robustly implementable by mechanism $\mathcal{M}$ if, on every type space $\mathcal{T}$, there exists an equilibrium of the game $(\mathcal{T}, \mathcal{M})$ and every equilibrium σ of the game $(\mathcal{T}, \mathcal{M})$ satisfies

$$\sum_{m \in M} g\left(f\left(\widehat{\theta}\left(t\right)\right)|m\right)\sigma\left(m|t\right) \geq 1-\varepsilon.$$

Social choice function f is virtually robustly implementable if it is $\varepsilon-$robustly implementable for all $\varepsilon > 0$.

In this section, we restrict attention to mechanisms that satisfy a best response property.

Definition 18 (Nice Mechanism)
Mechanism $\mathcal{M}$ is nice if for all i and $\theta_i \in \Theta_i$,

$$\begin{aligned} S^{\mathcal{M}}\left(\theta_i\right) &\neq \varnothing, \\ \text{and } \arg\max_{m_i} \sum_{m_{-i},\theta_{-i}} \lambda_i\left(m_{-i},\theta_{-i}\right) u_i\left(g\left(m_i,m_{-i}\right),\left(\theta_i,\theta_{-i}\right)\right) &\neq \varnothing, \end{aligned}$$

for all $\lambda_i \in \Delta\left(M_{-i} \times \Theta_{-i}\right)$.

Thus rationalizable sets are non-empty and best responses exist to all conjectures. This restriction is related to restrictions studied in the complete information literature, such as the bounded

mechanism condition of Jackson (1992), but we have not yet identified the exact relation. Note that this property is automatically satisfied if message spaces are finite. Next we present a sufficient condition to obtain an impossibility result for robust virtual implementation.[14]

Definition 19 (Indistinguishable Payoff Types)
Payoff types are indistinguishable if, for all i, there exists $\psi_i : \theta_i \to \Delta(\Theta_{-i})$ such that for all $\theta_i, \theta_i' \in \Theta_i$ and y, $y' \in Y$,

$$\begin{aligned} &\sum_{\theta_{-i}} u_i\left(y,(\theta_i,\theta_{-i})\right)\psi_i\left(\theta_{-i}|\theta_i\right) \geq \sum_{\theta_{-i}} u_i\left(y',(\theta_i,\theta_{-i})\right)\psi_i\left(\theta_{-i}|\theta_i\right) \\ \Leftrightarrow &\sum_{\theta_{-i}} u_i\left(y,\left(\theta_i',\theta_{-i}\right)\right)\psi_i\left(\theta_{-i}|\theta_i'\right) \geq \sum_{\theta_{-i}} u_i\left(y',\left(\theta_i',\theta_{-i}\right)\right)\psi_i\left(\theta_{-i}|\theta_i'\right). \end{aligned}$$

We note that any two types, θ_i and θ_i', have to agree in their ranking of the alternatives only for some specific posterior distribution, $\psi_i\left(\theta_{-i}|\theta_i\right)$ and $\psi_i\left(\theta_{-i}|\theta_i'\right)$, but not for all posterior distributions. On the other hand, the ranking has to be constant for any arbitrary pair, $\theta_i, \theta_i' \in \Theta_i$.

Proposition 3 (Failure of Virtual Robust Implementation)
If types are indistinguishable, then f is virtually robustly implementable by a nice mechanism if and only if it is constant.

Proof. Clearly f is virtually robustly implementable if it is constant. Suppose that types are not indistinguishable. Then there exists $v_i : Y \to \mathbb{R}$ such that type θ_i with beliefs $\psi_i(\theta_{-i})$ prefers y to y' if and only if $v_i(y) \geq v_i(y')$. We argue by induction that for all i and k, there exists m_i^k such that $m_i^k \in S_i^k(\theta_i)$ for all $\theta_i \in \Theta_i$. It is clearly true for $k = 0$ and to begin the inductive step suppose it is true for k. Fix any i and fix

$$m_i^{k+1} \in \arg\max_{m_i} \; v_i\left(g\left(m_i, m_{-i}^k\right)\right),$$

for some $m_{-i}^k \in S_{-i}^k$. Now $m_i^{k+1} \in S_i^{k+1}(\theta_i)$ for all θ_i. Thus for each i there exists $m_i^* \in S_i^{\mathcal{M}}(\theta_i)$ for all θ_i. Thus for any mechanism $\mathcal{M}$, there exists $y \in Y$ such that on any type space, outcome y is always realized. Thus virtual robust implementation requires that for every $\varepsilon > 0$, there exists $y \in Y$ such that $\|f(\theta) - y\| \leq \varepsilon$. This in turn requires that f is constant. ■

The same argument will imply that if types are indistinguishable, then a social choice function cannot be implemented by iterated deletion of "weakly ex post dominated" messages, as in Chung and Ely (2001).

[14] This condition can be further weakened with an iterative indistinguishability condition that is the robust analogue of the measurability condition in Abreu and Matsushima (1992b).

8 Linear Aggregation of Interdependent Types

In this section, we present a class of environments with interdependent preferences, for which we can derive precise implementation results. The environment is defined to be the class of interdependent preferences where the payoff types can be aggregated linearly for the utility representation. This class of preferences allows us to clearly illustrate the link between iterative and robust implementation. With the linear aggregation of the payoff types, we obtain two very distinct implementation results, separated by a sharp bound on the size of the interdependence in the valuations. If there is not too much interdependence, then robust implementation is possible. Conversely, if there is too much interdependence, then robust implementation will be impossible, but more surprisingly, even robust virtual implementation is impossible.

Thus we consider preferences $u_i(y,\theta)$ which permit linear aggregation of the payoff types as follows:

$$u_i(y,\theta) \triangleq v_i\left(y, \theta_i + \gamma \sum_{j \neq i} \theta_j\right),$$

with $\gamma \in \mathbb{R}$ and $\Theta_i = [0,1]$ for all i. We define the value of the linear aggregator for agent i as μ_i:

$$\mu_i : \mathbb{R}^I \to \mathbb{R},$$

and accordingly can write the preferences of agent i as a function of the allocation and the linear aggregator μ_i, or $v_i(y,\mu_i)$.

With the linear aggregation of the payoff types, we obtain two very distinct implementation results, separated by a sharp bound on the size of the interdependence in the valuations. We begin the impossibility result under "too much" interdependence.

Theorem 3 (Virtual Implementation)
If $\gamma > \frac{1}{I-1}$, then types are indistinguishable and virtual robust implementation is impossible.

Proof. Suppose $\gamma > \frac{1}{I-1}$. Let type θ_i always put probability 1 on a profile θ_{-i} with

$$\sum_{j \neq i} \theta_j = \frac{1}{\gamma}\left(\frac{1}{2} - \theta_i\right).$$

Then

$$\begin{aligned} \theta_i + \gamma \sum_{j \neq i} \theta_j &= \theta_i + \gamma\left(\frac{1}{\gamma}\left(\frac{1}{2} - \theta_i\right)\right) \\ &= \theta_i + \frac{1}{2} - \theta_i \\ &= \frac{1}{2}. \end{aligned}$$

This implies that types are indistinguishable. While Proposition 3 is written for discrete types, it extends straightforwardly to compact type spaces. ■

On the other hand, we have positive implementation results when there is not too much interdependence in preferences. Before we can state the positive implementation results, we need to make the usual assumptions regarding single crossing to guarantee ex post incentive compatibility in the direct mechanism.

Definition 20 (Single Crossing)
v_i satisfies single crossing if $v_i(y', \mu) = v_i(y, \mu)$ and $v_i(y', \mu') > v_i(y, \mu')$ imply $v_i(y', \widetilde{\mu}) > v_i(y, \widetilde{\mu})$ for all $\widetilde{\mu} > \mu$ (if $\mu' > \mu$) and for all $\widetilde{\mu} < \mu$ (if $\mu' < \mu$).

This condition requires that for any pair of lotteries y and y', there is at most one value of the aggregate of individual types, μ, where preferences over the lotteries switch. We state the single crossing condition in terms of the linear aggregator μ as we do not make any ordering assumption about the set of allocations Y.

Definition 21 (Non-degeneracy)
v_i is nondegenerate if for all y, $\mu \neq \mu'$, there exists y' such that $v_i(y', \mu) = v_i(y, \mu)$ and $v_i(y', \mu') > v_i(y, \mu')$.

With robust implementation, we need to strengthen the ex post incentive constraints in the direct mechanism to be strict inequalities.

Definition 22 (Strict Ex Post Incentive Compatibility)
Social choice function f satisfies strict ex post incentive compatibility if $f\left(\theta_i, \theta'_{-i}\right) \neq f\left(\theta'_i, \theta'_{-i}\right)$ for some θ'_{-i} implies

$$u_i\left(f\left(\theta_i, \theta_{-i}\right), \left(\theta_i, \theta_{-i}\right)\right) > u_i\left(f\left(\theta'_i, \theta_{-i}\right), \left(\theta_i, \theta_{-i}\right)\right)$$

for all i and θ_{-i}.

We can now establish the converse to Theorem 3.

Theorem 4 (Robust Implementation)
If $\gamma < \frac{1}{I-1}$, strict EPIC holds and each v_i satisfies single crossing and non-degeneracy, then robust monotonicity is satisfied.

Proof. Consider the deception β. Let

$$\Delta = \sup_{i, \theta_i, \theta'_i \in \beta_i(\theta_i)} \left|\theta'_i - \theta_i\right|.$$

If $\Delta = 0$, then the deception is acceptable and we are done. If $\Delta > 0$, fix any $0 < \varepsilon < 1 - \gamma(I-1)$ and (i, θ_i, θ_i') with $|\theta_i' - \theta_i| > \Delta(1-\varepsilon)$. Fix any θ_{-i}'. By non-degeneracy of v_i, there exists y' such that $u_i(y', \theta') = u_i(f(\theta'), \theta')$ and $u_i(y', (\theta_i, \theta_{-i}')) > u_i(f(\theta'), (\theta_i, \theta_{-i}'))$. Let y^n be a lottery putting probability $\frac{1}{n}$ on y' and probability $1 - \frac{1}{n}$ on $f(\theta')$. Now $y^n \to f(\theta')$, $u_i(y^n, \theta') = u_i(f(\theta'), \theta')$ and $u_i(y^n, (\theta_i, \theta_{-i}')) > u_i(f(\theta'), (\theta_i, \theta_{-i}'))$ for all n.

We first establish that for sufficiently large n, $y^n \in Y_i^*(\theta_{-i}')$. If this wasn't true, then for every n, there would exist $\widetilde{\theta}_i$ such that

$$u_i\left(y^n, \left(\widetilde{\theta}_i, \theta_{-i}'\right)\right) > u_i\left(f\left(\widetilde{\theta}_i, \theta_{-i}'\right), \left(\widetilde{\theta}_i, \theta_{-i}'\right)\right).$$

By continuity, we must have

$$u_i\left(f(\theta'), \left(\widetilde{\theta}_i, \theta_{-i}'\right)\right) \geq u_i\left(f\left(\widetilde{\theta}_i, \theta_{-i}'\right), \left(\widetilde{\theta}_i, \theta_{-i}'\right)\right),$$

for some $\widetilde{\theta}_i$, contradicting strict EPIC.

Now suppose that $\theta_{-i}' \in \beta_{-i}(\theta_{-i})$. Observe that

$$\left(\theta_i + \gamma \sum_{j \neq i} \theta_j\right) - \left(\theta_i' + \gamma \sum_{j \neq i} \theta_j'\right) = (\theta_i - \theta_i') + \gamma \sum_{j \neq i} (\theta_j' - \theta_j).$$

But by hypothesis:

$$|\theta_i - \theta_i'| > \Delta(1-\varepsilon) > \gamma(I-1)\Delta,$$

while

$$\begin{aligned} \gamma \left| \sum_{j \neq i} (\theta_j' - \theta_j) \right| &\leq \gamma \sum_{j \neq i} |\theta_j' - \theta_j| \\ &\leq \gamma(I-1)\Delta. \end{aligned}$$

So the sign of

$$\left(\theta_i + \gamma \sum_{j \neq i} \theta_j\right) - \left(\theta_i' + \gamma \sum_{j \neq i} \theta_j'\right),$$

equals the sign of

$$\theta_i - \theta_i'.$$

But now $u_i(y^n, \theta') = u_i(f(\theta'), \theta')$ and $u_i(y^n, (\theta_i, \theta_{-i}')) > u_i(f(\theta'), (\theta_i, \theta_{-i}'))$ implies, by single crossing of v_i, $u_i(y^n, (\theta_i, \theta_{-i})) > u_i(f(\theta'), (\theta_i, \theta_{-i}))$ for all θ_{-i} such that $\theta_{-i}' \in \beta_{-i}(\theta_{-i})$. This establishes robust monotonicity. ■

The proof of Theorem 4 use the following four types, $\theta_i, \widetilde{\theta}_i, \theta_i^n, \theta_i'$ of agent i,ordered on the real line:

$$\underrightarrow{\qquad\theta_i \qquad \widetilde{\theta}_i \qquad \theta_i^n \qquad \theta_i' \qquad}_{\mathbb{R}}$$

The current implementation results were obtained within an environment of linear and symmetric aggregation:

$$\mu_i = \theta_i + \gamma \sum_{j \neq i} \theta_j.$$

We can generalize the implementation results in this section to a general, not necessarily symmetric model of linear aggregation:

$$\mu_i = \theta_i + \sum_{j \neq i} \gamma_{ij} \theta_j.$$

The bound on the interdependence then comes through the eigenvalue of the interaction matrix $\boldsymbol{\gamma} = \{\gamma_{ij}\}$. We suspect that an appropriate linearization argument would extend the technique to general non-linear environments.

Finally, if we impose some additional structure on the payoffs as a function of the reports in the direct mechanism, then we can use the same argument as in Theorem 4 to obtain direct robust monotonicity results.

Definition 23 (Strictly Concave Deviations)
For any θ, $u_i\left(f\left(\widetilde{\theta}_i, \theta_{-i}\right), (\theta_i, \theta_{-i})\right)$ is strictly concave in $\widetilde{\theta}_i$.

Proposition 4 (Direct Implementation)
If $\gamma < \frac{1}{I-1}$, strict EPIC and strictly concave deviations hold and each v_i satisfies single crossing, non-degeneracy, then robust direct monotonicity is satisfied.

Proof. The argument in the previous theorem goes through where we instead to define $y^n = f\left(\theta_i^n, \theta_{-i}'\right)$ with $\theta_i^n \to \theta_i'$ and the sign of $\theta_i^n - \theta_i'$ equal to the sign of $\theta_i - \theta_i'$. Since only the direct mechanism messages are used, the proof then establishes direct robust monotonicity rather than robust monotonicity. ■

9 Extensions, Variations and Discussion

9.1 Lotteries, Pure Strategies and Bayesian Implementation

In this section, we discuss how our main Theorem 2 is related to the classic literature on Bayesian implementation developed by Postlewaite and Schmeidler (1986), Palfrey and Srivastava (1989)

and Jackson (1991). These authors asked whether it was possible to implement a social choice function in equilibrium on a fixed type space $\mathcal{T}$.[15] These authors analyzed the classic problem where attention was restricted to pure strategy equilibria and deterministic mechanisms. Thus the social choice function is a mapping $f : \Theta \to Z$.

Having fixed a type space, the natural notion of a pure strategy deception on the fixed type space is a collection $\alpha = (\alpha_1, .., \alpha_I)$, with each $\alpha_i : T_i \to T_i$. Thus $\alpha : T \to T$ is defined by $\alpha(t) = (\alpha_i(t_i))_{i=1}^I$. The key monotonicity notion, translated into our language, was then the following:

Definition 24 (Bayesian Monotonicity)
Social choice function f satisfies Bayesian monotonicity on type space $\mathcal{T}$ if, for every deception α with $f\left(\widehat{\theta}(t)\right) \neq f\left(\widehat{\theta}(\alpha(t))\right)$ for some t, there exists i, t_i and $h : T \to Z$ such that

$$\sum_{t_{-i} \in T_{-i}} u_i\left(h(\alpha(t)), \widehat{\theta}(t)\right) \widehat{\pi}_i(t_{-i})[t_i] > \sum_{t_{-i} \in T_{-i}} u_i\left(f\left(\widehat{\theta}(\alpha(t))\right), \widehat{\theta}(t)\right) \widehat{\pi}_i(t_{-i})[t_i], \tag{13}$$

and

$$\begin{aligned} & \sum_{t_{-i} \in T_{-i}} u_i\left(f\left(\widehat{\theta}\left(t_i', t_{-i}\right)\right), \widehat{\theta}\left(t_i', t_{-i}\right)\right) \widehat{\pi}_i(t_{-i})\left[t_i'\right] \\ \geq & \sum_{t_{-i} \in T_{-i}} u_i\left(h\left(\alpha_i(t_i), t_{-i}\right), \widehat{\theta}\left(t_i', t_{-i}\right)\right) \widehat{\pi}_i(t_{-i})\left[t_i'\right], \ \forall t_i'. \end{aligned} \tag{14}$$

Jackson shows that this condition is necessary for Bayesian implementation, and that a slight strengthening, Bayesian monotonicity no veto, is sufficient.

In the Appendix (Section 11.3), we show that our robust monotonicity condition is equivalent to the requirement that Bayesian monotonicity is satisfied on all type spaces. Thus we have an alternative way of showing the necessity of robust monotonicity for robust implementation. Figure 1 gave a graphical representation of how the results fit together.

But note that this line of argument would establish the necessity of robust implementation if the planner is restricted to deterministic mechanisms (a disadvantage) but he can assume that players follow pure strategies (an advantage). How do these assumptions matter?

First, observe that the advantage of restricting attention to pure strategies goes away completely when we require implementation on all type spaces: if there is a mixed strategy equilibrium that results in a socially sub-optimal outcome on some type space, we can immediately construct a larger type space (purifying the original equilibrium) where the socially sub-optimal outcome is played in

[15] They allowed for more general social choice sets, but we restrict attention to functions for our comparison.

a pure strategy equilibrium. Thus our robust analysis conveniently removes that unfortunate gap between pure and mixed strategy implementation that has plagued the implementation literature.

We use the extension to stochastic mechanisms in just two places. Ex post incentive compatibility and robust monotonicity would remain necessary conditions even if we restricted attention to deterministic mechanisms (the arguments would be unchanged). But, as we note in Footnote 13, even if lotteries were not used in the implementing mechanism, the implied robust monotonicity condition would involve lotteries (as rewards for whistle-blowers). But if lotteries were not allowed, our sufficiency argument would then require a slightly strengthened version of the robust monotonicity condition, with the lottery y replaced by a deterministic outcome. Our sufficiency argument also uses lotteries under Rules 1 and 2. As in a recent paper by Benoit and Ok (2004) on complete information implementation, we use lotteries to significantly weaken the sufficient conditions, so that we require only the "bad outcome" property in addition to EPIC and robust monotonicity. If we did not allow lotteries in this part of the argument, we would require a much stronger economic condition in the spirit of Jackson's "Bayesian monotonicity no veto" condition. We have developed combined robust monotonicity and economic conditions (not reported here) sufficient for interim implementation on all full support types spaces. However, an additional complication is that, without lotteries in the implementing mechanism, we cannot establish sufficiency on type spaces where agents have disjoint supports.

It is possible to construct a simple example where EPIC and robust monotonicity are not sufficient for robust monotonicity without lotteries by taking the coordination example of Section 3.2 but removing the outcomes e and f. As we show in the Appendix (Section 11.4), robust implementation is then not possible in this example despite the fact that the social choice function selects a unique strictly Pareto-dominant outcome at every type profile.

9.2 Nice Mechanisms

In our analysis of robust implementation, we deliberately allowed for very badly behaved infinite mechanisms in order to make a tight connection with the existing literature and to get tight results. Many authors have argued that "integer game" constructions, like that we use in Theorem 2, should not be taken seriously (see, e.g., Abreu and Matsushima (1992a) and Jackson (1992)). In our analysis of virtual robust implementation in Section 7, we restricted attention to "nice" mechanisms with best responses always well defined. Much of our analysis of the relation between iterative and robust implementation, and the characterization of robust implementation, would be much simpler with the restriction to nice mechanisms. In fact with nice mechanisms we obtain the following stronger necessary conditions for robust implementation.

Definition 25 (Strict Robust Monotonicity)
Social choice function f satisfies strict robust monotonicity if for every unacceptable deception β, there exist i, θ_i, $\theta_i' \in \beta_i(\theta_i)$ such that, for all $\theta_{-i}' \in \Theta_{-i}$ and $\psi_i \in \Delta\left(\beta_{-i}^{-1}\left(\theta_{-i}'\right)\right)$, there exists y such that

$$\sum_{\theta_{-i}\in\Theta_{-i}} \psi_i(\theta_{-i})\, u_i(y,(\theta_i,\theta_{-i})) > \sum_{\theta_{-i}\in\Theta_{-i}} \psi_i(\theta_{-i})\, u_i\left(f\left(\theta_i',\theta_{-i}'\right),(\theta_i,\theta_{-i})\right), \tag{15}$$

while

$$u_i\left(f\left(\theta_i'',\theta_{-i}\right),\left(\theta_i',\theta_{-i}'\right)\right) > u_i\left(y,\left(\theta_i'',\theta_{-i}'\right)\right),\ \forall\theta_i''. \tag{16}$$

The proof of the subsequent robust implementation result closely follows the proof of necessity part of Theorem 2. But the existence of best responses in the definition of a nice mechanism allows the necessary conditions to be strengthened to their strict versions.

Proposition 5 (Robust Implementation)
If f is robustly implementable by a nice mechanism, then f satisfies strict EPIC and strict robust monotonicity.

Proof. See Appendix. ■

With the restriction to nice mechanisms, the relationship between iterative deletion and robust implementation emerges more directly. Finally, we should mention that we do not have general sufficient conditions for robust implementation by nice mechanisms, just instances where robust implementation is possible in the direct mechanism and examples where it is possible in nice augmented mechanisms (e.g., the coordination example of Section 3.2).

9.3 Extensions

The previous sections examined the importance of our assumptions about lotteries over outcomes and restrictions on mechanisms. We also restricted attention in our main analysis to the case of discrete but perhaps infinite payoff types Θ_i and types T_i, although our examples and linear aggregation results dealt with compact Θ_i.

Many of our results would extend easily to more general Θ_i and T_i. This is true of the direct implementation analysis, the necessary conditions for robust implementation and the virtual implementation analysis. The sufficiency for robust implementation might be more delicate.

10 Conclusion

This paper examined the robustness of the classic implementation problem. We formalized robustness by requiring that the implementation problem remains solvable as we gradually relax common

knowledge among the agents and the designer. The weakening of common knowledge was achieved by considering large type spaces in which the private information of the individual agents becomes more prominent.

In contrast to our earlier results on truthful implementation, Bergemann and Morris (2004), robust implementation is in general a more demanding notion of implementation than ex post implementation. It remains an open question whether a systematic relationships between ex post, interim and robust implementation do arise in specific environments such as single crossing or supermodular environments. The analysis of the environment with interdependent values and linear aggregation in Section 8 clearly suggests that a systematic relationship can be established for many interesting environments. We also extended the robustness argument to the notion of virtual implementation. While your analysis here was preliminary, it clearly offered evidence that the distance between interim and virtual implementation may shrink considerably after imposing robustness on the implementation concept.

11 Appendix

11.1 Virtual Implementation in the Single Unit Auction

We complete the iterative implementation argument for the single unit auction example of Section 3.4. We study the following symmetric ε-efficient allocation rule is the following:

$$x_i^*(\theta) = \varepsilon\frac{1}{I}\theta_i + (1-\varepsilon)\, x_i^{**}(\theta).$$

The corresponding essentially unique ex post transfer rule is:

$$y_i(\theta) = \varepsilon\frac{1}{I}\gamma\left(\sum_{j\neq i}\theta_j\right)\theta_i + \frac{1}{2I}\varepsilon(\theta_i)^2 + (1-\varepsilon)\left(\max_{j\neq i}\theta_j + \gamma\sum_{j\neq i}\theta_j\right)x_i^{**}(\theta).$$

Thus if the true type profile is θ and agents report themselves to be type profile θ', agent i's expected utility is

$$\left(\theta_i + \gamma\sum_{j\neq i}\theta_j\right)x_i^*(\theta') - y_i(\theta'),$$

or

$$\begin{aligned}&\left(\theta_i + \gamma\sum_{j\neq i}\theta_j\right)\left(\varepsilon\frac{1}{I}\theta_i' + (1-\varepsilon)\, x_i^{**}(\theta')\right)\\ &-\varepsilon\frac{1}{I}\gamma\left(\sum_{j\neq i}\theta_j'\right)\theta_i' - \frac{1}{2I}\varepsilon(\theta_i')^2 - (1-\varepsilon)\left(\max_{j\neq i}\theta_j' + \gamma\sum_{j\neq i}\theta_j'\right)x_i^{**}(\theta')\end{aligned}$$

or

$$\varepsilon\frac{1}{I}\theta_i'\left(\theta_i - \frac{1}{2}\theta_i' + \gamma\sum_{j\neq i}(\theta_j - \theta_j')\right) + (1-\varepsilon)\left(\left(\theta_i - \max_{j\neq i}\theta_j'\right) + \gamma\sum_{j\neq i}(\theta_j - \theta_j')\right)x_i^{**}(\theta').$$

The payoff gain to agent i of reporting himself to be type $\theta_i + x$ when his true type is θ_i is

$$\begin{aligned}&\varepsilon\frac{1}{I}\left((\theta_i+x)\left(\theta_i - \frac{1}{2}(\theta_i+x) + \gamma\sum_{j\neq i}(\theta_j-\theta_j')\right) - \theta_i\left(\frac{1}{2}\theta_i + \gamma\sum_{j\neq i}(\theta_j-\theta_j')\right)\right)\\ &+(1-\varepsilon)\begin{pmatrix}\left(\left(\theta_i - \max_{j\neq i}\theta_j'\right) + \gamma\sum_{j\neq i}(\theta_j-\theta_j')\right)x_i^{**}(\theta_i+x,\theta'_{-i})\\ -\left(\left(\theta_i - \max_{j\neq i}\theta_j'\right) + \gamma\sum_{j\neq i}(\theta_j-\theta_j')\right)x_i^{**}(\theta_i,\theta'_{-i})\end{pmatrix},\end{aligned}$$

which is equal to

$$\varepsilon\frac{1}{I}\left(x\gamma\sum_{j\neq i}\left(\theta_j-\theta'_j\right)-\frac{1}{2}x^2\right)$$
$$+\left(1-\varepsilon\right)\left(\left(\theta_i-\max_{j\neq i}\theta'_j\right)+\gamma\sum_{j\neq i}\left(\theta_j-\theta'_j\right)\right)\left(x_i^{**}\left(\theta_i+x,\theta'_{-i}\right)-x_i^{**}\left(\theta_i,\theta'_{-i}\right)\right).$$

Now, the first term is maximized by setting

$$x=\gamma\sum_{j\neq i}\left(\theta_j-\theta'_j\right).$$

and the second term is maximized by choosing $x>\max_{j\neq i}\theta'_j-\theta_i$ if

$$\theta_i>\max_{j\neq i}\theta'_j-\gamma\sum_{j\neq i}\left(\theta_j-\theta'_j\right);$$

anything if

$$\theta_i=\max_{j\neq i}\theta'_j-\gamma\sum_{j\neq i}\left(\theta_j-\theta'_j\right);$$

and $x<\max_{j\neq i}\theta'_j-\theta_i$ if

$$\theta_i<\max_{j\neq i}\theta'_j-\gamma\sum_{j\neq i}\left(\theta_j-\theta'_j\right).$$

Thus the whole expression is maximized setting

$$x=\gamma\sum_{j\neq i}\left(\theta_j-\theta'_j\right).$$

This is exactly the same best response property as we obtained in the public good game. Therefore we get robust implementation in the direct mechanism if $\gamma<\frac{1}{I-1}$ as in the example of the public good in Section 3.3.

11.2 A Badly Behaved Mechanism

The example illustrates the gap between the necessary and sufficient conditions in Proposition 2. Specifically, it shows that there can be an equilibrium for every type space $\mathcal{T}$ in a mechanism, yet $S^{\mathcal{M}}$ does not satisfy the ex post best response property.

In the example, there are two agents and there is complete information, so each agent has a unique type. There are a finite number of outcomes $Z=\{a,b,c\}$. The payoffs are given by the

following table:

	a	b	c
agent 1	-1	0	$+1$
agent 2	0	0	0

The planner's choice (in the unique payoff state) is a. Thus it is trivial to robustly implement the social choice function. But suppose that the planner chooses the following (strange) mechanism: $M_1 = \{1, 2, 3, ...\}$, $M_2 = \{1, 2\}$ and

$$g(m_1, m_2) = \begin{cases} a, & \text{if} \quad m_1 = 1 \\ b, & \text{if} \quad m_1 > 1 \text{ and } m_2 = 1 \\ \left[\frac{1}{m_1}, b; \left(1 - \frac{1}{m_1}\right), c\right], & \text{if} \quad m_1 > 1 \text{ and } m_2 = 2 \end{cases}$$

where $\left[\frac{1}{m_1}, b; \left(1 - \frac{1}{m_1}\right), c\right]$ is the lottery putting probability $\frac{1}{m_1}$ on b and probability $\left(1 - \frac{1}{m_1}\right)$ on c. Thus $g(m_1, m_2)$ can be represented by the following table:

g	1	2
1	a	a
2	b	$\left(\frac{1}{2}, b; \frac{1}{2}, c\right)$
3	b	$\left(\frac{1}{3}, b; \frac{2}{3}, c\right)$
$\vdots$	$\vdots$	$\vdots$
k	b	$\left(\frac{1}{k}, b; 1 - \frac{1}{k}, c\right)$
$\vdots$	$\vdots$	$\vdots$

Thus the agents are playing the following complete information game:

m_1/m_2	1	2
1	$0, 0$	$0, 0$
2	$-1, 0$	$\frac{1}{2}, 0$
3	$-1, 0$	$\frac{2}{3}, 0$
$\vdots$	$\vdots$	$\vdots$
k	$-1, 0$	$1 - \frac{1}{k}, 0$
$\vdots$	$\vdots$	$\vdots$

Now on any type space, there is always an equilibrium where player 1 chooses action 1 and player 2 chooses action 1, and outcome a is chosen. Moreover, on any type space, in any equilibrium, outcome a is always chosen: if player 1 ever has a best response not to play 1 then he has no best

response. So he always plays 1 in equilibrium. Thus the trivial social choice function is robustly implemented by this mechanism.

While only message 1 survives iterated deletion of never best responses for player 1, both messages survive iterated deletion of never best responses for player 2. Thus we have $S_1^{\mathcal{M}} = \{1\}$ and $S_2^{\mathcal{M}} = \{1, 2\}$. Note that $S^{\mathcal{M}}$ satisfies the interim best response property, see Definition 6, but not the ex post best response property, see Definition 5. For we observe that

$$u_1\left(g\left(1,2\right)\right) = u_1\left(a\right) = 0 < \frac{1}{2} = u_1\left(g\left(2,2\right)\right),$$

violating the ex post best response property.

The insight of the example is that the quantifier "for every type space $\mathcal{T}$" does not necessarily guarantee that all actions which will be chosen with positive probability in some equilibrium and for some type space, will also be chosen with probability one in some equilibrium for some type space. For this reason, the quantifier "for every type space $\mathcal{T}$" does not allow us to establish a local, i.e. ex post best response property of every action in $S^{\mathcal{M}}$.

11.3 Bayesian Monotonicity

The next proposition establishes the equivalence between robust monotonicity and Bayesian monotonicity on every type space by means of a constructive proof (via a specific type space). The constructive element is the identification of a type space on which Bayesian monotonicity is guaranteed to fail if robust monotonicity fails. It is worthwhile to note that the specific type space is much smaller than the universal type.

In some sense, the notion of robustness is more subtle in the context of full rather than partial implementation. With partial implementation, i.e. truthtelling in the direct mechanism, the universal type space is by definition the most difficult type space to obtain truthtelling. In the universal type space, every agent has the maximal number of possible misreports and hence the designer faces the maximal number of incentive constraints. In the context of full implementation, the trade-off is ambiguous. As a larger type space contains by definition more types, it offers every agent more possibilities to misreport. But then, just as a larger type space made truthtelling more difficult to obtain, the other equilibria might also cease to exist after the introduction of additional types. This second part offers the possibility that larger type spaces facilitate rather than complicate the full implementation problem.

Proposition 6 (Equivalence)
Social choice function f satisfies Bayesian monotonicity on every type space if and only if it satisfies robust monotonicity.

Proof. ($\Rightarrow$) We will show that if robust monotonicity fails, we can construct a type space where Bayesian monotonicity fails. The argument will be constructive.

Fix an unacceptable deception β. Suppose that robust monotonicity fails. Then for each i, θ_i, $\theta_i' \in \beta_i(\theta_i)$, there exist

$$\theta_{-i}\left[\theta_i, \theta_i'\right] \in \Theta_{-i} \quad \text{and} \quad \psi_i\left[\theta_i, \theta_i'\right] \in \Delta\left(\beta_{-i}^{-1}\left(\theta_{-i}\left[\theta_i, \theta_i'\right]\right)\right) \tag{17}$$

such that:

$$u_i\left(f\left(\theta_i'', \theta_{-i}\left[\theta_i, \theta_i'\right]\right), \left(\theta_i'', \theta_{-i}\left[\theta_i, \theta_i'\right]\right)\right) \geq u_i\left(y, \left(\theta_i'', \theta_{-i}\left[\theta_i, \theta_i'\right]\right)\right), \quad \forall \theta_i'' \in \Theta_i \tag{18}$$

implies

$$\sum_{\theta_{-i} \in \Theta_{-i}} \psi_i(\theta_{-i})\left[\theta_i, \theta_i'\right] u_i\left(f\left(\theta_i', \theta_{-i}\left[\theta_i, \theta_i'\right]\right), (\theta_i, \theta_{-i})\right) \geq \sum_{\theta_{-i} \in \Theta_{-i}} \psi_i(\theta_{-i})\left[\theta_i, \theta_i'\right] u_i(y, (\theta_i, \theta_{-i})). \tag{19}$$

Now we construct a type space around θ_i, θ_i' and $\psi_i\left[\theta_i, \theta_i'\right]$ given by (17) for which Bayesian monotonicity fails. First, agent i has a set of "deception" types T_i^1 which are isomorphic to $\Xi_i = \left\{\left(\theta_i, \theta_i'\right) : \theta_i \in \Theta_i \text{ and } \theta_i' \in \beta_i(\theta_i)\right\}$; thus there exists a bijection $\xi_i^1 : T_i^1 \to \Xi_i$. The type responding to $\left(\theta_i, \theta_i'\right)$ has payoff type θ_i and believes that the other agents are of type:

$$\left(\left[\xi_j^1\right]^{-1}\left(\theta_j, \theta_{ij}\left[\theta_i, \theta_i'\right]\right)\right)_{j \neq i}$$

with probability $\psi_i(\theta_{-i})\left[\theta_i, \theta_i'\right]$. Second, agent i has a set of "pseudo-complete information types" T_i^2, which are isomorphic to Θ; thus there exists a bijection $\xi_i^2 : T_i^2 \to \Theta$. The type corresponding to θ has payoff type θ_i and he is convinced that each other agent j is type $\left[\xi_j^1\right]^{-1}(\theta_j, \theta_j)$.

Slightly more formally, we have

$$T_i = T_i^1 \cup T_i^2.$$

If $t_i \in T_i^1$ and $\xi_i^1(t_i) = \left(\theta_i, \theta_i'\right)$, then

$$\widehat{\theta}_i(t_i) = \theta_i;$$

if $t_i \in T_i^2$ and $\xi_i^2(t_i) = \theta$, then

$$\widehat{\theta}_i(t_i) = \theta_i.$$

If $t_i \in T_i^1$ and $\xi_i^1(t_i) = \left(\theta_i, \theta_i'\right)$, then

$$\pi_i^*(t_{-i})[t_i] = \begin{cases} \psi_i(\theta_{-i})\left[\theta_i, \theta_i'\right], & \text{if} \quad t_{-i} \in T_{-i}^1 \text{ and } \theta_{-i} = \left(\left[\xi_j^1\right]^{-1}\left(\theta_j, \theta_{ij}\left[\theta_i, \theta_i'\right]\right)\right)_{j \neq i} \\ 0, & \text{if} \quad \text{otherwise} \end{cases}$$

If $t_i \in T_i^2$ and $\xi_i^2(t_i) = \theta$, then

$$\pi_i^*(t_{-i})[t_i] = \begin{cases} 1, & \text{if} \quad t_{-i} \in T_{-i}^1 \text{ and } \theta_{-i} = \left(\left[\xi_j^1\right]^{-1}\left(\theta_j, \theta_{ij}\left[\theta_i, \theta_i'\right]\right)\right)_{j \neq i} \\ 0, & \text{if} \qquad\qquad\qquad\qquad \text{otherwise} \end{cases}$$

Now consider the Bayesian deception on this type space where each type $\left[\xi_i^1\right]^{-1}\left(\theta_i, \theta_i'\right)$ reports himself to be type $\left[\xi_i^1\right]^{-1}\left(\theta_i', \theta_i'\right)$, and all other types report their types truthfully. Thus

$$\alpha_i(t_i) = \begin{cases} \left[\xi_i^1\right]^{-1}\left(\theta_i', \theta_i'\right), & \text{if} \quad t_i = \left[\xi_i^1\right]^{-1}\left(\theta_i, \theta_i'\right) \\ t_i, & \text{if} \qquad \text{otherwise} \end{cases}.$$

Since β was unacceptable, we must have that $f\left(\widehat{\theta}(t)\right) \neq f\left(\widehat{\theta}(\alpha(t))\right)$ for some t. Thus the Bayesian monotonicity condition (Definition 24) for this type space requires that there exist i, t_i and $h : T \to Z$ such that

$$\sum_{t_{-i} \in T_{-i}} u_i\left(h(\alpha(t)), \widehat{\theta}(t)\right) \widehat{\pi}_i(t_{-i})[t_i] > \sum_{t_{-i} \in T_{-i}} u_i\left(f\left(\widehat{\theta}(\alpha(t))\right), \widehat{\theta}(t)\right) \widehat{\pi}_i(t_{-i})[t_i], \tag{20}$$

and

$$\begin{aligned} & \sum_{t_{-i} \in T_{-i}} u_i\left(f\left(\widehat{\theta}\left(t_i'', t_{-i}\right)\right), \widehat{\theta}\left(t_i'', t_{-i}\right)\right) \widehat{\pi}_i(t_{-i})\left[t_i''\right] \\ \geq & \sum_{t_{-i} \in T_{-i}} u_i\left(h\left(\alpha_i(t_i), t_{-i}\right), \widehat{\theta}\left(t_i'', t_{-i}\right)\right) \widehat{\pi}_i(t_{-i})\left[t_i''\right], \ \forall t_i''. \end{aligned} \tag{21}$$

The t_i cannot be an element of T_i^2, because such a type does not expect any deviation from truth-telling under the deception. So it must be an element of T_i^1, with $\xi_i^1(t_i) = \left(\theta_i, \theta_i'\right)$. Now condition (20) becomes

$$\begin{aligned} & \sum_{\theta_{-i} \in \Theta_{-i}} u_i\left(h\left(\left[\xi_i^1\right]^{-1}\left(\theta_i', \theta_i'\right), \left(\left(\left[\xi_j^1\right]^{-1}\left(\theta_{ij}\left[\theta_i, \theta_i'\right], \theta_{ij}\left[\theta_i, \theta_i'\right]\right)\right)_{j \neq i}\right)\right), \left(\theta_i, \theta_{-i}\right)\right) \psi_i(\theta_{-i})\left[\theta_i, \theta_i'\right] \\ > & \sum_{\theta_{-i} \in \Theta_{-i}} u_i\left(f\left(\theta_i', \theta_{-i}\left[\theta_i, \theta_i'\right]\right), \left(\theta_i, \theta_{-i}\right)\right) \psi_i(\theta_{-i})\left[\theta_i, \theta_i'\right]. \end{aligned} \tag{22}$$

But letting t_i'' in condition (21) be in T_i^2 with $\xi_i^2\left(t_i''\right) = \left(\theta_i'', \theta_{-i}\left[\theta_i, \theta_i'\right]\right)$, we have

$$\begin{aligned} & u_i\left(f\left(\theta_i'', \theta_{-i}\left[\theta_i, \theta_i'\right]\right), \left(\theta_i'', \theta_{-i}\left[\theta_i, \theta_i'\right]\right)\right) \\ \geq & u_i\left(h\left(\left[\xi_i^1\right]^{-1}\left(\theta_i', \theta_i'\right), \left(\left[\xi_j^1\right]^{-1}\left(\theta_{ij}\left[\theta_i, \theta_i'\right], \theta_{ij}\left[\theta_i, \theta_i'\right]\right)\right)_{j \neq i}\right), \left(\theta_i'', \theta_{-i}\left[\theta_i, \theta_i'\right]\right)\right) \end{aligned} \tag{23}$$

for all θ_i''. Setting

$$z = h\left(\left[\xi_i^1\right]^{-1}\left(\theta_i', \theta_i'\right), \left(\left(\left[\xi_j^1\right]^{-1}\left(\left(\theta_{ij}\left[\theta_i, \theta_i'\right], \theta_{ij}\left[\theta_i, \theta_i'\right]\right)\right)\right)_{j \neq i}\right)\right),$$

condition (22) becomes

$$\begin{aligned} & \sum_{\theta_{-i}\in\Theta_{-i}} u_i\left(z,\left(\theta_i,\theta_{-i}\right)\right)\psi_i\left(\theta_{-i}\right)\left[\theta_i,\theta_i'\right] \\ > & \sum_{\theta_{-i}\in\Theta_{-i}} u_i\left(f\left(\theta_i',\theta_{-i}\left[\theta_i,\theta_i'\right]\right),\left(\theta_i,\theta_{-i}\right)\right)\psi_i\left(\theta_{-i}\right)\left[\theta_i,\theta_i'\right]. \end{aligned}$$

while condition (23) requires $z \in Y_i\left(\theta_{-i}\left[\theta_i,\theta_i'\right]\right)$. But these latter claims contradict our initial assumption that robust monotonicity fails (i.e., (18)). Thus Bayesian monotonicity fails for this type space and the claim is proved.

($\Leftarrow$) Suppose f satisfies robust monotonicity. Fix any type space $\mathcal{T}$ and any deception α with $f\left(\widehat{\theta}\left(t\right)\right) \neq f\left(\widehat{\theta}\left(\alpha\left(t\right)\right)\right)$ for some t. Define $\boldsymbol{\beta}$ by

$$\beta_i\left(\theta_i\right) = \left\{\theta_i' : \exists t_i \text{ such that } \widehat{\theta}_i\left(t_i\right) = \theta_i \text{ and } \widehat{\theta}_i\left(\alpha_i\left(t_i\right)\right) = \theta_i'\right\}.$$

Deception $\boldsymbol{\beta}$ is unacceptable, so by robust monotonicity, there exist i, θ_i, $\theta_i' \in \beta_i\left(\theta_i\right)$ such that for every $\theta_{-i}' \in \Theta_{-i}$ and $\psi_i \in \Delta\left(\beta_{-i}^{-1}\left(\theta_{-i}'\right)\right)$, there exists $y\left[\theta_{-i}',\psi_i\right] \in Y_i\left(\theta_{-i}'\right)$ such that

$$\sum_{\theta_{-i}\in\Theta_{-i}} \psi_i\left(\theta_{-i}\right) u_i\left(y\left[\theta_{-i}',\psi_i\right],\left(\theta_i,\theta_{-i}\right)\right) > \sum_{\theta_{-i}\in\Theta_{-i}} \psi_i\left(\theta_{-i}\right) u_i\left(f\left(\theta_i',\theta_{-i}'\right),\left(\theta_i,\theta_{-i}\right)\right). \tag{24}$$

Now choose any t_i such that $\widehat{\theta}_i\left(t_i\right) = \theta_i$ and $\widehat{\theta}_i\left(\alpha_i\left(t_i\right)\right) = \theta_i'$. For every (mis-)report θ_{-i}', we now derive a distribution over payoff types θ_{-i} which represents the likelihood that the report θ_{-i}' comes from the true payoff type profile θ_{-i}, given the type space $\mathcal{T}$. For each θ_{-i}', define $\psi_i\left[\theta_{-i}'\right] \in \Delta\left(\Theta_{-i}\right)$ by

$$\psi_i\left(\theta_{-i}\right)\left[\theta_{-i}'\right] \triangleq \frac{\sum_{\left\{t_{-i}:\widehat{\theta}_j(\alpha_j(t_j))=\theta_j' \text{ and } \widehat{\theta}_j(t_j)=\theta_j,\ \forall j\neq i\right\}} \widehat{\pi}_i\left(t_{-i}\right)\left[t_i\right]}{\sum_{\left\{t_{-i}:\widehat{\theta}_j(\alpha_j(t_j))=\theta_j',\ \forall j\neq i\right\}} \widehat{\pi}_i\left(t_{-i}\right)\left[t_i\right]}. \tag{25}$$

Now let h satisfy

$$h\left(t_i',t_{-i}\right) \triangleq \begin{cases} y\left[\widehat{\theta}_{-i}\left(t_{-i}\right),\psi_i\left[\widehat{\theta}_{-i}\left(t_{-i}\right)\right]\right] & \text{if} \quad t_i' = \alpha_i\left(t_i\right) \\ f\left(\widehat{\theta}\left(t_i',t_{-i}\right)\right) & \text{if} \quad \text{otherwise} \end{cases}. \tag{26}$$

To establish Bayesian monotonicity, it is enough to show that the two inequalities of Bayesian monotonicity are satisfied, or:

$$\sum_{t_{-i}} u_i\left(h\left(\alpha\left(t\right)\right),\widehat{\theta}\left(t\right)\right)\widehat{\pi}_i\left(t_{-i}\right)\left[t_i\right] > \sum_{t_{-i}} u_i\left(f\left(\widehat{\theta}\left(\alpha\left(t\right)\right)\right),\widehat{\theta}\left(t\right)\right)\widehat{\pi}_i\left(t_{-i}\right)\left[t_i\right], \tag{27}$$

and

$$\begin{aligned} & \sum_{t_{-i}} u_i\left(f\left(\widehat{\theta}\left(t_i',t_{-i}\right)\right),\widehat{\theta}\left(t_i',t_{-i}\right)\right)\widehat{\pi}_i\left(t_{-i}\right)\left[t_i'\right] \\ \geq & \sum_{t_{-i}} u_i\left(h\left(\alpha_i\left(t_i\right),t_{-i}\right),\widehat{\theta}\left(t_i',t_{-i}\right)\right)\widehat{\pi}_i\left(t_{-i}\right)\left[t_i\right], \ \forall t_i'. \end{aligned} \tag{28}$$

By inserting the posterior beliefs ψ_i and the rewards $h\left(t_i', t_{-i}\right)$, as defined above in (25) and (26) respectively, we can rewrite the two sides of the inequality (27) as follows:

$$\begin{aligned}
&\sum_{t_{-i}} u_i\left(h\left(\alpha\left(t\right)\right), \widehat{\theta}\left(t\right)\right) \widehat{\pi}_i\left(t_{-i}\right)\left[t_i\right] \\
= &\sum_{\theta'_{-i}} \left(\sum_{\left\{t_{-i}: \widehat{\theta}_j(\alpha_j(t_j))=\theta'_j,\ \forall j\neq i\right\}} \widehat{\pi}_i\left(t_{-i}\right)\left[t_i\right] \right) \sum_{\theta_{-i}} \psi_i\left(\theta_{-i}\right)\left[\theta'_{-i}\right] u_i\left(y\left[\theta'_{-i}, \psi_i\left[\theta'_{-i}\right]\right], \theta\right)
\end{aligned}$$

and

$$\begin{aligned}
&\sum_{t_{-i}} u_i\left(f\left(\widehat{\theta}\left(\alpha\left(t\right)\right)\right), \widehat{\theta}\left(t\right)\right) \widehat{\pi}_i\left(t_i\right)\left[t_{-i}\right] \\
= &\sum_{\theta'_{-i}} \left(\sum_{\left\{t_{-i}: \widehat{\theta}_j(\alpha_j(t_j))=\theta'_j,\ \forall j\neq i\right\}} \widehat{\pi}_i\left(t_{-i}\right)\left[t_i\right] \right) \sum_{\theta_{-i}} \psi_i\left(\theta_{-i}\right)\left[\theta'_{-i}\right] u_i\left(f\left(\theta'\right), \theta\right)
\end{aligned}$$

so (27) follows from (24). Also

$$\begin{aligned}
&\sum_{t_{-i}} u_i\left(h\left(\alpha_i\left(t_i\right), t_{-i}\right), \widehat{\theta}\left(t_i', t_{-i}\right)\right) \widehat{\pi}_i\left(t_{-i}\right)\left[t_i'\right] \\
= &\begin{cases} \sum_{t_{-i}} u_i\left(y\left[\widehat{\theta}_{-i}\left(t_{-i}\right), \psi_i\left[\widehat{\theta}_{-i}\left(t_{-i}\right)\right]\right], \widehat{\theta}\left(t_i', t_{-i}\right)\right) \widehat{\pi}_i\left(t_{-i}\right)\left[t_i'\right] & \text{if} \quad t_i' = \alpha_i\left(t_i\right) \\ \sum_{t_{-i}} u_i\left(f\left(\widehat{\theta}\left(t_i', t_{-i}\right)\right), \widehat{\theta}\left(t_i', t_{-i}\right)\right) \widehat{\pi}_i\left(t_{-i}\right)\left[t_i'\right], & \text{if} \quad t_i' \neq \alpha_i\left(t_i\right) \end{cases}
\end{aligned}$$

Now $y\left[\widehat{\theta}_{-i}\left(t_{-i}\right), \psi_i\left[\widehat{\theta}_{-i}\left(t_{-i}\right)\right]\right] \in Y_i\left(\widehat{\theta}_{-i}\left(t_{-i}\right)\right)$ implies (28). ■

The proof may appear rather intricate in its details. We next give a brief outline of the basic steps to show how interim implies robust monotonicity. The proof proceeds by contrapositive. We start with an unacceptable deception β which by hypothesis fails robust monotonicity and hence satisfies the inequalities (18) and (19). For the given deception β, we then create a type space, consisting of two components for every agent i. The first component for agent i is created by the set of pairs of payoff types $\left(\theta_i, \theta_i'\right)$, where the first entry is the true payoff type and the second entry is a feasible deception (under β), or $\theta_i' \in \beta_i\left(\theta_i\right)$. For this reason, we refer to these types as "deception types." For every such pair $\left(\theta_i, \theta_i'\right)$ there exists at least one particular payoff profile θ'_{-i} which acts as a misreport. Under the deception β, this payoff profile θ'_{-i} could have been reported by all true payoff profiles which are in the support of ψ_i. Consequently, the belief component of type $\left(\theta_i, \theta_i'\right)$ is given by simply adopting $\psi_i\left(\cdot \left| \theta_i, \theta_i'\right.\right)$. The second component consists of "pseudo complete information types", described by $t_i = \theta \in \Theta$. Each such type has a belief that assigns probability one to the event that the true payoff profile is given by θ and that all other agents report the deception type (θ_j, θ_j), and hence the "pseudo" in the labelling.

Given this type space T_i, we then consider a particular deception $\alpha_i : T_i \to T_i$. The deception α_i is localized around the "deception types" and the "pseudo complete information types" report truthfully. The deception α_i consists of agent i always reporting his deception type rather than his true type, or $\alpha_i\left(\theta_i, \theta_i'\right) = \left(\theta_i', \theta_i'\right)$. We then verify whether f is interim monotone under α. The existence of the pseudo complete information types θ forces the interim incentive compatibility conditions to reduce to ex post incentive compatibility conditions. This guarantees the hypothesis in the robust monotonicity notion, namely inequality (18), and thus leads to the conclusion in form of the inequalities (19). But then we obtain a contradiction to the reward condition of interim monotonicity, unless the hypothesis for the interim monotonicity condition, namely $f \neq f \circ \alpha$, is not satisfied, i.e. $f = f \circ \alpha$ holds, but of course this implies that β is acceptable.

11.4 Coordination Example 2

The next example is the pure coordination game, which we first considered in Section 3.2, without the additional allocations, z and z'. It illustrates the importance of lotteries for robust implementation. The example will satisfy EPIC and robust monotonicity, yet it cannot be robustly implemented without the use of lotteries. On the other hand if lotteries are allowed then the lottery which selects each of the four possible outcomes with equal probability constitutes a bad outcome, and hence the sufficient conditions for robust implementation would be satisfied with lotteries.

The examples has two agents, $i = 1, 2$ and each agent i has two possible types, θ_i and θ_i'. There are four possible outcomes: $Z = \{a, b, c, d\}$. Agents' payoffs are given by:

a	θ_2	θ_2'
θ_1	$3, 3$	$0, 0$
θ_1'	$0, 0$	$1, 1$

b	θ_2	θ_2'
θ_1	$0, 0$	$3, 3$
θ_1'	$1, 1$	$0, 0$

c	θ_2	θ_2'
θ_1	$0, 0$	$1, 1$
θ_1'	$3, 3$	$0, 0$

d	θ_2	θ_2'
θ_1	$1, 1$	$0, 0$
θ_1'	$0, 0$	$3, 3$

The social choice function f selects the efficient outcome in every state:

f	θ_2	θ_2'
θ_1	a	b
θ_1'	c	d

As in the example in Section 3.2, the social choice function is strictly ex post incentive compatible but there is another equilibrium in the "direct mechanism" where each agent misreports his type, and each agent gets a payoff of 1.

Robust monotonicity is clearly satisfied even if the rewards y are restricted to be the deterministic allocations Z. We will show that robust implementation is not possible even in an infinite

mechanism if we restrict attention to deterministic mechanisms. Fix a mechanism $\mathcal{M}$. Let

$$S_i^*(\theta_i) = \{m_i : g(m_i, m_j) = f(\theta_i, \theta_j) \text{ for some } m_j, \theta_j\},$$

be the set of messages for agent i which would select the allocation recommended by the social choice function for some m_j, θ_j. Let $S_i^k(\theta_i) = b_i\left(b^{k-1}\left(\overline{S}\right)\right)[\theta_i]$ (using transfinite induction if necessary). We now show by induction that, $S_i^*(\theta_i) \subseteq S_i^k(\theta_i)$ for all k using the structure of the payoffs. Suppose that this is true for k. Then for any $m_i \in S_i^*(\theta_i) \subseteq S_i^k(\theta_i)$, there exists $m_j \in S_j^*(\theta_j) \subseteq S_j^k(\theta_j)$ such that $g(m_i, m_j) = f(\theta_i, \theta_j)$. Thus there does not exist $\nu_i \in \Delta(M_i)$ such that

$$\sum_{m_i'} \nu_i\left(m_i'\right) u_i\left(g\left(m_i', m_j\right), (\theta_i, \theta_j)\right) > u_i\left(g(m_i, m_j), (\theta_i, \theta_j)\right) = 3.$$

So $m_i \in S_i^{k+1}(\theta_i)$.

Thus we must have that $(m_1, m_2) \in S_1^*(\theta_1) \times S_2^*(\theta_2)$ implies $g(m_1, m_2) = f(\theta_1, \theta_2)$. Let $m_i^*(\cdot)$ be any selection from $S_i^*(\cdot)$. Now let k^* be the lowest k such that, for some i,

$$m_i^*\left(\theta_i'\right) \notin S_i^k(\theta_i).$$

Without loss of generality, let $i = 1$. Note $m_2^*\left(\theta_2'\right) \in S_2^{k-1}(\theta_2)$ by definition of k^*. If agent 1 was type θ_1 and was sure his opponent were type θ_2 and choosing action $m_2^*\left(\theta_2'\right)$, we know that he could guarantee himself a payoff of 1 by choosing $m_1^*\left(\theta_1'\right)$. Since $m_1^*\left(\theta_1'\right)$ is deleted for type θ_1 at round k^*, we know that there exists $\nu_1 \in \Delta(M_1)$ such that

$$\sum_{m_1'} \nu_1\left(m_1'\right) g_1\left(m_1', m_2^*\left(\theta_2'\right)\right) > 1,$$

and thus there exists m_1' such that $g_1\left(m_1', m_2^*\left(\theta_2'\right)\right) = f(\theta_1, \theta_2)$. This implies that $m_2^*\left(\theta_2'\right) \in S_2^*(\theta_2)$, a contradiction.

The example uses the fact that the social choice function always selects an outcome that is strictly Pareto-optimal and - paradoxically - it is this feature which inhibits iterative implementation in the current example. Borgers (1995) proves the impossibility of complete information implementation of non-dictatorial social choice functions in iteratively undominated strategies when the set of feasible preference profiles includes such unanimous preference profiles and the argument here is reminiscent of Borgers' argument.

11.5 Nice Mechanisms and Strict Robust Monotonicity

Proof of Proposition 5. The restriction to nice mechanisms ensures that $S^{\mathcal{M}}$ is non-empty. It follows that if mechanism $\mathcal{M}$ iteratively implements f, then, for each i, there exists $m_i^* : \Theta_i \to M_i$

such that

$$g\left(m^{*}\left(\theta\right)\right)=f\left(\theta\right) \text{ and } m^{*}\left(\theta\right)\in S^{\mathcal{M}}\left(\theta\right),$$

we can simply let $m_i^{*}\left(\theta_i\right)$ be any element of $S_i^{\mathcal{M}}\left(\theta_i\right)$.

We first establish strict EPIC. Suppose strict EPIC fails. Then there exists i, θ and θ_i' such that $f\left(\theta_i',\theta_{-i}\right)\neq f\left(\theta_i,\theta_{-i}\right)$ and

$$u_i\left(f\left(\theta_i',\theta_{-i}\right),\theta\right)\geq u_i\left(f\left(\theta\right),\theta\right).$$

Now, for any message m_i with

$$m_i\in \arg\max_{m_i'} u_i\left(g\left(m_i',m_{-i}^{*}\left(\theta_{-i}\right)\right),\left(\theta_i,\theta_{-i}\right)\right),$$

since $m_{-i}^{*}\left(\theta_{-i}\right)\in S_i^{\infty}\left(\theta_{-i}\right)$, we must have $m_i\in S_i^{\infty}\left(\theta_i\right)$ and thus $g\left(m_i,m_{-i}^{*}\left(\theta_{-i}'\right)\right)=f\left(\theta_i,\theta_{-i}'\right)$ for all θ_{-i}'. Thus let

$$m_i^{*}\left(\theta_i'\right)\in \arg\max_{m_i'} u_i\left(g\left(m_i',m_{-i}^{*}\left(\theta_{-i}\right)\right),\left(\theta_i,\theta_{-i}\right)\right),$$

and $f\left(\theta_i',\theta_{-i}'\right)=g\left(m_i^{*}\left(\theta_i'\right),m_{-i}^{*}\left(\theta_{-i}'\right)\right)=f\left(\theta_i,\theta_{-i}'\right)$ for all θ_{-i}', a contradiction.

Now we establish strict robust monotonicity. Fix an unacceptable deception β. Let $\widehat{k}$ be the largest k such that such that for every i, θ_i and $\theta_i'\in\beta_i\left(\theta_i\right)$,

$$S_i^{\infty}\left(\theta_i'\right)\subseteq S_i^{k}\left(\theta_i\right).$$

We know that such a $\widehat{k}$ exists because $S_i^{0}\left(\theta_i\right)\cap S_i^{\infty}\left(\theta_i'\right)=S_i^{\infty}\left(\theta_i'\right)$ and, since $\mathcal{M}$ iteratively implements f, we must have $S_i^{\infty}\left(\theta_i\right)\cap S_i^{\infty}\left(\theta_i'\right)=\varnothing$.

Now we know that there exists i and $\theta_i'\in\beta_i\left(\theta_i\right)$ such that

$$S_i^{\widehat{k}+1}\left(\theta_i\right)\cap S_i^{\infty}\left(\theta_i'\right)\neq S_i^{\infty}\left(\theta_i'\right).$$

Let

$$\widehat{m}_i\in S_i^{\widehat{k}}\left(\theta_i\right)\cap S_i^{\infty}\left(\theta_i'\right),$$

and

$$\widehat{m}_i\notin S_i^{\widehat{k}+1}\left(\theta_i\right)\cap S_i^{\infty}\left(\theta_i'\right).$$

Since message $\widehat{m}_i$ gets deleted for θ_i at round $\widehat{k}+1$, we know that for every $\lambda_i\in\Delta\left(M_{-i}\times\Theta_{-i}\right)$ such that

$$\lambda_i\left(m_{-i},\theta_{-i}\right)>0\Rightarrow m_j\in S_j^{\widehat{k}}\left(\theta_j\right) \text{ for all } j\neq i,$$

there exists m_i^* such that

$$\sum_{m_{-i},\theta_{-i}} \lambda_i\left(m_{-i},\theta_{-i}\right) u_i\left(g\left(m_i^*,m_{-i}\right),\left(\theta_i,\theta_{-i}\right)\right) > \sum_{m_{-i},\theta_{-i}} \lambda_i\left(m_{-i},\theta_{-i}\right) u_i\left(g\left(\widehat{m}_i,m_{-i}\right),\left(\theta_i,\theta_{-i}\right)\right).$$

Let

$$\widehat{m}_j \in S_j^{\infty}\left(\theta_j'\right)$$

for all $j \neq i$. Now the above claim remains true if we restrict attention to distributions λ_i putting probability 1 on $\widehat{m}_{-i}$. Thus for every $\psi_i \in \Delta\left(\Theta_{-i}\right)$ such that

$$\psi_i\left(\theta_{-i}\right) > 0 \Rightarrow \widehat{m}_j \in S_j^{\widehat{k}}\left(\theta_j\right) \text{ for all } j \neq i,$$

there exists m_i^* such that

$$\sum_{\theta_{-i}} \psi_i\left(\theta_{-i}\right) u_i\left(g\left(m_i^*,\widehat{m}_{-i}\right),\left(\theta_i,\theta_{-i}\right)\right) > \sum_{\theta_{-i}} \psi_i\left(\theta_{-i}\right) u_i\left(g\left(\widehat{m}_i,\widehat{m}_{-i}\right),\left(\theta_i,\theta_{-i}\right)\right).$$

But $\widehat{m} \in S^{\infty}\left(\theta'\right)$, so (since $\mathcal{M}$ iteratively implements f), $g\left(\widehat{m}_i,\widehat{m}_{-i}\right) = f\left(\theta'\right)$. Also observe that if $\theta_{-i}' \in \beta_{-i}\left(\theta_{-i}\right)$, then $\widehat{m}_{-i} \in S_{-i}^{\widehat{k}}\left(\theta_{-i}\right)$. Thus for every $\psi_i \in \Delta\left(\beta_{-i}^{-1}\left(\theta_{-i}'\right)\right)$, there exists m_i^* such that

$$\sum_{\theta_{-i}} \psi_i\left(\theta_{-i}\right) u_i\left(g\left(m_i^*,\widehat{m}_{-i}\right),\left(\theta_i,\theta_{-i}\right)\right) > \sum_{\theta_{-i}} \psi_i\left(\theta_{-i}\right) u_i\left(f\left(\theta'\right),\left(\theta_i,\theta_{-i}\right)\right),$$

which establishes the reward inequality, (15), of strict robust monotonicity.

Now suppose the incentive inequality, (16), are not satisfied strictly, and hence:

$$u_i\left(g\left(m_i^*,\widehat{m}_{-i}\right),\left(\widetilde{\theta}_i,\theta_{-i}'\right)\right) \geq u_i\left(f\left(\widetilde{\theta}_i,\theta_{-i}'\right),\left(\widetilde{\theta}_i,\theta_{-i}'\right)\right).$$

Now, for any

$$m_i \in \arg\max_{m_i'} u_i\left(g\left(m_i',\widehat{m}_{-i}\right),\left(\widetilde{\theta}_i,\theta_{-i}'\right)\right), \tag{29}$$

since $\widehat{m}_{-i} \in S_i^{\infty}\left(\theta_{-i}'\right)$, we must have $m_i \in S_i^{\infty}\left(\widetilde{\theta}_i\right)$ and thus $g\left(m_i,\widehat{m}_{-i}\right) = f\left(\theta_i,\theta_{-i}'\right)$. Thus from (29) we also know that m_i^* achieves the maximum:

$$m_i^* \in \arg\max_{m_i'} u_i\left(g\left(m_i',\widehat{m}_{-i}\right),\left(\widetilde{\theta}_i,\theta_{-i}'\right)\right)$$

and, for all $\widetilde{\theta}_i$, if

$$u_i\left(g\left(m_i^*,\widehat{m}_{-i}\right),\left(\widetilde{\theta}_i,\theta_{-i}'\right)\right) \geq u_i\left(f\left(\widetilde{\theta}_i,\theta_{-i}'\right),\left(\widetilde{\theta}_i,\theta_{-i}'\right)\right),$$

then $g\left(m_i^*,\widehat{m}_{-i}\right) = f\left(\widetilde{\theta}_i,\theta_{-i}'\right)$.

Now setting $y \equiv g\left(m_i^*, \widehat{m}_{-i}\right)$, we have established that for each $\theta'_{-i} \in \beta_{-i}\left(\theta_{-i}\right)$ and $\psi_i \in \Delta\left(\beta_{-i}^{-1}\left(\theta'_{-i}\right)\right)$, there exists y such that $y \in Y_i^*\left(\theta'_{-i}\right)$ and

$$\sum_{\theta_{-i}} \psi_i\left(\theta_{-i}\right) u_i\left(g\left(m_i^*, \widehat{m}_{-i}\right), \left(\theta_i, \theta_{-i}\right)\right) > \sum_{\theta_{-i}} \psi_i\left(\theta_{-i}\right) u_i\left(f\left(\theta'\right), \left(\theta_i, \theta_{-i}\right)\right),$$

which concludes the proof. ■

12 Notation

$\alpha_i(\theta_i)$ deception

$b_i(S)[\theta_i]$ never best response operator

$\beta_i(\theta_i)$ deception correspondence

$\underline{\beta}_i^k, \overline{\beta}_i^k$ lower and upper bound for best responses

f social choice function

g outcome function

γ interdependence parameter

I number of agents

$\lambda_i(\theta_{-i}, m_{-i})$ belief of agent i

μ_i linear aggregator of payoff types

$\nu_i(m_{-i})$ belief over messages

$\psi_i(\theta_{-i})$ belief over payoff types

$\psi_i[\cdot]$ specific belief in Bayesian monotonicity

$\pi_i(t_{-i})[t_i]$ conditioning event

$\rho\left(\theta'_{-i}, \theta_{-i}\right)$ belief over reports and types (in direct mechanism)

S_i, S message correspondence

S_i^k strategies surviving $k-th$ round of elimination

t_i, t type (profile)

θ_i, θ payoff type

$\theta_{-i}[\cdot,\cdot]$ payoff type in Bayesian monotonicity

u_i utility

v_i utility with linear aggregator

y allocation

$y[\cdot,\cdot]$ reward in Bayesian monotonicity

y_i monetary transfer

$Y = \Delta(Z)$ extended outcome space

$Y_i(\theta_{-i})$ incentive compatible reward set

$Y_i^*(\theta_{-i})$ strictly incentive compatible reward set

ζ message

Z outcome space

References

Abreu, D., and H. Matsushima (1992a): "Virtual Implementation in Iterative Undominated Strategies: Complete Information," *Econometrica*, 60, 993–1008.

——— (1992b): "Virtual Implementation In Iteratively Undominated Strategies: Incomplete Information," Discussion paper, Princeton University and University of Tokyo.

Battigalli, P. (1999): "Rationalizability in Incomplete Information Games," Discussion Paper ECO 99/17, European University Institute.

Battigalli, P., and M. Siniscalchi (2003): "Rationalization and Incomplete Information," *Advances in Theoretical Economics*, 3, Article 3.

Benoit, J., and E. Ok (2004): "Nash Implementation Without No Veto," Discussion paper, New York University.

Bergemann, D., and S. Morris (2004): "Robust Mechanism Design," *Econometrica*, forthcoming.

——— (2005): "Ex Post Implementation," Discussion Paper 1502, Cowles Foundation for Research in Economics, Yale University.

Borgers, T. (1995): "A Note on Implementation and Strong Dominance," in *Social Choice, Welfare and Ethics*, ed. by W. Barnett, H. Moulin, M. Salles, and N. Schofield. Cambridge University Press, Cambridge.

Brandenburger, A., and E. Dekel (1987): "Rationalizability and Correlated Equilibria," *Econometrica*, 55, 1391–1402.

Chung, K.-S., and J. C. Ely (2001): "Efficient and Dominance Solvable Auctions with Interdependent Valuations," Discussion paper, Northwestern University.

Dekel, E., D. Fudenberg, and S. Morris (2005): "Interim Rationalizability," Discussion paper, Tel-Aviv University, Harvard University and Yale University.

Duggan, J. (1997): "Virtual Bayesian Implementation," *Econometrica*, 65, 1175–1199.

Hughes, G., and M. Creswell (1996): *A New Introduction Into Modal Logic*. Routledge, London.

Jackson, M. (1991): "Bayesian Implementation," *Econometrica*, 59, 461–477.

JACKSON, M. (1992): "Implementation in Undominated Strategies: A Look at Bounded Mechanisms," *Review of Economic Studies*, 59, 757–775.

LIPMAN, B. (1994): "A Note on the Implications of Common Knowledge of Rationality," *Games and Economic Behavior*, 6, 114–129.

PALFREY, T., AND S. SRIVASTAVA (1989): "Mechanism Design with Incomplete Information: A Solution to the Implementation Problem," *Journal of Political Economy*, 97, 668–691.

POSTLEWAITE, A., AND D. SCHMEIDLER (1986): "Implementation in Differential Information Economies," *Journal of Economic Theory*, 39, 14–33.

SERRANO, R., AND R. VOHRA (2005): "A Charaterization of Virtual Bayesian Implementation," *Games and Economic Behavior*, 50, 312–331.

WILSON, R. (1987): "Game-Theoretic Analyses of Trading Processes," in *Advances in Economic Theory: Fifth World Congress*, ed. by T. Bewley, pp. 33–70, Cambridge. Cambridge University Press.