DEFINABLE AND CONTRACTIBLE CONTRACTS

MICHAEL PETERS AND BALÁZS SZENTES

Abstract. This paper analyzes a normal form game in which actions as well as contracts are contractible. The contracts are required to be representable in a formal language. We prove a folk-theorem for games with complete and incomplete information. This is accomplished by constructing contracts which are definable functions of the Godel code of every other player's contract. We use this to illustrate the 'meet the competition' argument from Industrial Organization and the 'principle of reciprocity' from Trade and Public Finance.

1. Self Referential Strategies and Reciprocity in Static Games

The idea that players in a game might simultaneously commit themselves to react to their competitors actions is heuristically compelling. The best known expression of this idea is well known in the industrial organization literature (e.g. [8]) as the 'meet the competition' clause. A similar idea appears in trade theory as the principle of *reciprocity* ([1]). This takes the form of trade agreements like GATT that require countries to match tariff cuts by other countries. Finally, tax treaties sometimes have this flavor - for example, out of state residents who work in Pennsylvania are exempt from Pennsylvania tax as long as they live in a state that has a reciprocal agreement that exempts out of state residents (presumably from Pennsylvania) from state taxes.[1]

One way to model reciprocity is to embed it in a dynamic game. For example the tit for tat strategy in the repeated prisoners' dilemma makes each player's action depend on the action of the other. The 'meet the competition' argument could be supported formally by having one firm acts as a Stackleberg leader, offering a contract that commits it to an action that depends explicitly on the action of the second mover. Tax reciprocity could again be accomplished by embedding the problem in a repeated game in which states keep lists of other states whom they consider to have an appropriate tax treaty, deleting a state from the list if they observe some kind of bad behavior. Our interest here is whether this same kind of reciprocal behavior could be modelled in a completely static game.

To illustrate the problem, and our solution, focus first on the meet the competition argument. The Stackleberg leader, call it firm A, offers to sell at a very high price provided its competitor, firm B, also offers that high price in the second round. If B in the second round offers any price below the highest price, A commits itself to sell at marginal cost. If B believes this commitment, then his best reply is to set the highest price. If the firms move simultaneously, then the logic of the argument becomes clouded. A could certainly write a contract that commits it to a high price

[1] http://www.revenue.state.pa.us/revenue/cwp/view.asp?A=238&Q=244681

if B sets the same high price. However suppose that B's strategy is simply to set this high price and that for some reason this is a best reply to A's contract. Then A should deviate and simply undercut firm B. To support the high price outcome, firm B would have to offer a contract similar to A's in order to prevent A's deviation. A naive argument would suggest that B should simply offer the same contract as A, a high price if A sets a high price, and marginal cost otherwise. Casually, two outcomes seem consistent with these contracts - both firms price at marginal cost or both firms set the high price. This seems to violate a fairly fundamental property of game theory which is that for each pair of actions (contracts in this case), there is a unique payoff to every player.[2] More to the point, A's contract doesn't actually say what A would do if B offers a contract that promises to set a high price unless A sets a lower price, etc. The specification of the problem itself seems to be ambiguous about payoffs.

The reciprocal tax agreement also nicely illustrates the difficulty in a static game. State A wants to exempt residents of state B from state taxes provided B exempts residents of state A from taxes. To write the law A exempts residents from any state that has a 'reciprocal' agreement with state A. The question is what exactly is a 'reciprocal' agreement. It is clear enough what the intention is - create a situation in which both states take the mutually beneficial action of exempting one another in a way that eliminates any incentive for either of them to deviate. As mentioned above, it isn't enough to assume that state B unconditionally exempts residents of state A from tax because A would not longer have any incentive to exempt state B. State B has to have a law like the law in state A, in other words, a reciprocal agreement.

It seems that to resolve this kind of problem one needs to define the term 'reciprocal contract' as follows:

$$\text{reciprocal contract} \equiv \begin{cases} \text{exempt if the other state offers a reciprocal contract,} \\ \text{don't otherwise} \end{cases}$$

This kind of definition is familiar from the Bellman equation in dynamic programming where the value function is defined in a self referential way. It is tempting to model this in the following naive way: start by defining a collection of contracts that seem economically sensible. For example, it is reasonable that a state could write a contract that simply fixes any tax rate independent of what the other states do. Let $\overline{C}$ be the set of contracts that simply fix some unconditional tax rate. Append to this set of feasible contracts the reciprocal contract, call it r, defined above. Now model the set of feasible contracts as $\overline{C} \cup \{r\}$. The reciprocal contract above is just r, while 'otherwise' means any contract with a fixed tax rate. Define a normal form game in which the strategies are $\overline{C} \cup \{r\}$ and declare the outcome if both states offer r to be $(\text{exempt}, \text{exempt})$. Voila, there is an equilibrium in which the states mutually exempt (assuming they jointly want to).

[2] One paper that allows multiple payoffs to be associated with each array of actions is [9] who use this approach to support equilibrium when it might not otherwise exist.

We would argue that this is unsatisfactory for a number of reasons. First, it is undesirable to restrict the set of feasible conracts in order to support the outcome you are looking for. The approach described above amount to little more than saying that r is the only feasible contract, then claiming it is an equilibrium for both states to offer r. A more satisfactory approach is to define a set of actions that seem economically meaningful, then to allow the broadest set of contracts possible. In the same manner that the value function emerges endogenously from the economic environment, the reciprocal contract should be derived from economic fundamentals.

One complication that makes this problem conceptually more difficult than the Bellman problem is that it isn't clear what the appropriate set of feasible contracts should be. Clearly states can go further than simple unconditional tax rates. Existing laws do allow them to make things contingent on laws in other states, as the reciprocal tax agreement illustrates. One contribution of our approach is to provide a potential framework for thinking about this set of feasible contracts. As we describe in more detail below, definable functions can all be written as finite sentences in a formal language. In a heuristic sense, this seems exactly what the set of contracts should look like. Of course, in the way it is used here, definability is very abstract. Yet it is abstract in the way that direct mechanisms are abstract - it captures in a formal way 'indirect' contracts that look much more familiar.

Second, the approach described above misses the essence of reciprocity which is the infinite regress involved in self referential objects. A contract that makes formal sense is the following:

$$\text{C} = \begin{cases} \text{exempt if other State exempts any State who exempts any State who exempts}\ldots \\ \text{don't otherwise} \end{cases}$$

where the statement in the top line is repeated ad infinitum. Arguably, the contract C is a reciprocal contract since it would exempt any State offering a reciprocal contract. Yet it simply isn't feasible under the naive description given above.

Of course, in the spirit of the ad hoc approach above, we could try to add the contract C to r and $\overline{C}$. This approach breaks down once the game becomes asymmetric. For example, if State A is supposed to exempt, while state B is supposed to take some other action, say 'partly exempt', then to support the right outcome, the contracts should look something like the following:

$$\text{reciprocal contract}_A \equiv \begin{cases} \text{exempt if other State offers reciprocal contract}_B \\ \text{don't exempt otherwise} \end{cases}$$

and

$$\text{reciprocal contract}_B \equiv \begin{cases} \text{partially exempt if other State offers reciprocal contract}_A \\ \text{don't exempt otherwise} \end{cases}$$

Now the contracts are not directly self referential, as is the Bellman equation, instead they are cross referential. A single self referential or reciprocal contract simply doesn't go far enough. Furthermore, the contracts above use a blanket punishment for deviations. Desirable or interesting equilibrium allocations may not look like this. For example, in a Bayesian game between the states, State A might want to do something different for each different thing that state B might do. This might arise if the action and contract chosen by B convey some information to A that affects A's most desirable action.

In this paper, we offer a formalism that provides a way to think about self-referentiality and reciprocity. Suppose there are N players in a normal form game in which each player has a countable number of actions. Endow players with a formal language that they can use to write contracts and think of the set of feasible contracts as the set of finite sequences of characters in this formal language. It is well known that there are bijections from the set of finite texts into $\mathbb{N}$. One such a mapping is called the *Godel Coding*. Provided the language includes all the natural numbers and the usual arithmetic operations, it is possible for players to write contracts that are *definable* functions from $\mathbb{N}^{N-1}$ into that player's action space. Since definable functions can be written as finite sequences of characters in the language, they have Godel codes associated with them. Hence we could interpret the definable functions as contracts that make the players action depend on the Godel code of the other player's contract.

To make the argument easier to relate to conventional contract theory, we assume below that the contract space for each player is the set of definable functions from $\mathbb{N}^{N-1}$ into the subsets of the player's action spaces. Implicitly, this approach makes it possible for players to offer any finite text as a contract. We defer discussion of this point to later in the paper. Every definable function can be associated with a unique integer, and conversely if the integer n is associated with a definable function, then it is associated with a unique text. Now for each array of functions chosen by the players, compute the Godel Code of each such function. Fit the codes of the other players' strategies into each player's strategy to determine aunique subset of actions for every player. Then, players simultaneously take actions from these subsets.

Our objective is to try to characterize the set of equilibria of this game. To see how it works, we might as well restrict attention to a two player prisoner's dilemma. Call the players 1 and 2, and the actions C and D with the usual payoff structure in which D is a dominant strategy and both players are strictly better off if they both play C than they are if they both play D. A strategy c for a player is a definable function from $\mathbb{N}$ to $\{C, D\}$. One obvious equilibrium of this game occurs when both players use a strategy that chooses action D no matter what the Godel code of the other player's strategy.

Every definable strategy has a Godel code. Let $[c]$ denote the Godel code of the strategy c and refer to $[c]$ as the 'encoding' of c. Since the Godel coding is an injection from the set of definable strategies to the set of integers. For any pair of strategies c_1 and c_2, the action (C or D) taken by player 1 is $c_1([c_2])$ and similarly for player 2. Since every pair of actions determines a payoff, this procedure associates a unique payoff with every pair of strategies.

There are many things that aren't definable strategies that also have Godel codes. We want to make use of some of these other things. In particular, we want to use definable strategies with *free* *variables*. For example, there is a subclass of definable strategies for player 1 defined parametrically by

$$\gamma_x(n) = \begin{cases} C & n = x, \\ D & \text{otherwise.} \end{cases}$$

This is simply a definable strategy with a *free variable* x, where x is the target code of the other player's strategy that will trigger the cooperative action. Definable strategies with free variables are also definable, and so they too have Godel codes. The strategy with free variable that we want is a slight modification of the one above, in particular

$$c_x(n) = \begin{cases} C & n = \left[\langle x \rangle^{(x)}\right], \\ D & \text{otherwise.} \end{cases} \tag{1.1}$$

The mapping $< x >^{(x)}$is the composition of two functions. First, the function $\langle x \rangle$ is the inverse operation to the Godel coding. That is, $< n >$ is the text whose Godel code is n. Second, if ϕ is a text with one free variable, then $\phi^{(n)}$ is the same text where the value of the free variable is set to be n. Hence, if n is a Godel code of a definable strategy with one free variable, then $< n >^{(n)}$ is itself a definable strategy (without a free variable). $\left[\langle n \rangle^{(n)}\right]$ is just the Godel code of whatever this definable strategy happens to be. Notice that in this case, $[\langle x \rangle (x)]$ won't be equal to x since a definable strategy must have a different Godel code from a definable strategy with one free variable because of the fact that the Godel coding is injective.

We want to define a strategy by fixing a value for x in (1.1). In particular, the value of x we are interested in is $[c_x]$. Since $[c_x]$ is the Godel code of a strategy with a free variable, the right hand side of (1.1) requires that we decode $[c_x]$ to get c_x, then fix x at $[c_x]$ to get the contract $c_{[c_x]}$. Putting all this together gives

$$c_{[c_x]}(n) = \begin{cases} C & n = \left[c_{[c_x]}\right] \\ D & \text{otherwise} \end{cases}$$

So

$$c_{[c_x]}([c_2]) = \begin{cases} C & [c_2] = \left[c_{[c_x]}\right] \\ D & \text{otherwise} \end{cases}$$

is a the 'reciprocal' or self-referential contract mentioned above. Now we simply need to verify what happens when both players use strategy $c_{[c_x]}$.

If player 2 uses strategy $c_{[c_x]}$, then $[c_2] = \left[c_{[c_x]}\right]$, which evidently triggers the cooperative action by player 1. The same argument applies for player 2. Player 2 can deviate to any alternative definable strategy c' that she likes. Since every definable strategy has a Godel code, the reaction of player 1, and consequently both players payoffs are well defined. As the Godel coding is injective, $c' \neq c_{[c_x]}$ implies the Godel code of c' is not equal to $\left[c_{[c_x]}\right]$, and the deviation by 2 induces 1 to respond by switching from C to D.

Notice that this argument makes use of an encoding of the strategy with free variable c_x, which isn't a definable strategy. One might have expected the target code number to be associated with a strategy instead of a strategy with a free variable. For example, it seems that to enforce cooperation there needs to be a definable strategy c^* with encoding $[c^*] = n^*$ such that

$$c^* = \begin{cases} C & [c_2] = n^* \\ D & \text{otherwise} \end{cases}$$

Of course, for arbitrary n^* it will be false that $[c_{n^*}] = n^*$. This leads to a fixed point problem that, in fact, does not have a solution in general. More generally, one could try to construct a self-referential contract by finding a fixed point of the the following problem. For each n, consider

$$c_n([c_2]) = \begin{cases} C & \text{if } [c_2] = g(n), \\ D & \text{otherwise,} \end{cases}$$

where g is a definable function. If there exists an n^* such that $[c_{n^*}] = g(n^*)$, then c_{n^*} is obviously a self-referential contract. Indeed, what we did above is that we chose $g(n)$ to be $[< n >^{(n)}]$ and showed that $n^* = [c_x]$ is a corresponding fixed point.

To see how the strategy with free variable c_x works, recall the reciprocal tax agreement

$$\text{reciprocal contract} \equiv \begin{cases} \text{exempt} & \text{other State offers reciprocal contract} \\ \text{don't exempt} & \text{otherwise} \end{cases}$$

and its recursive counterpart

$$\text{C} = \begin{cases} \text{exempt if other State exempts any State who exempts any State who exempts...} \\ \text{don't otherwise} \end{cases}$$

The 'reciprocal contract' is $c_{[c_x]}$ and the statement "other state offers reciprocal contract" is $[c_2] = [c_{[c_x]}]$.

State A wants to exempt any state whose law fullfills a condition. For example, if the condition it is looking for is that the other state simply exempts State S, then it would compute the Godel code $n_0 = [\forall n; \overline{c}(n) = C]$ then use the strategy

$$c_{n_0} = \begin{cases} C & [c_2] = n_0 \\ D & \text{otherwise} \end{cases}$$

If it does that, then it can't be an equilibrium as explained above. So what it needs to do is to exempt any State whose law fullfills a condition that exempts any state whose law fullfills a condition. For example, if it wanted to exempt State B if and only if State B's law exempts state A if and only if State A unconditionally exempts state B, then it would adopt the strategy $c_{[c_{n_0}]}$, and so on.

This is where the particular structure of the contract c_x comes into play. Recall that

$$c_x(n) = \begin{cases} C & n = \left[\langle x \rangle^{(x)}\right], \\ D & \text{otherwise.} \end{cases}$$

It specifies exemption if and only if a condition is fullfilled, but it doesn't seem to specify what the condition is. However, it does require that whatever the condition x is, if x in turn depends on a condition, then the condition that it depends on must be the same as the condition itself. To see if x depends on a condition, we first decode it and find the statement $\langle x \rangle$ that the integer x corresponds to. Then if it depends on some condition, we require that that condition be x itself, which is the meaning of $\langle x \rangle^{(x)}$. So now we can do the infinite regress. State A adopts a law that exempts state B if and only if the Godel code of State B's law is $\left[c_{[c_x]}\right]$. This means that state B's law must be $c_{[c_x]}$, or that B exempt A if and only if the Godel code of State A's law is $\left[c_{[c_x]}\right]$, i.e., the same condition that A requires.

2. Literature

The approach we develop here is not the only way to sustain cooperation without repetition. An alternative logic has been developed in the theory of common agency ([6] or [7]) in which punishments are carried out through an agent. In problems in which principals interact through many agents, this logic can be used to prove 'folk theorems'. The argument appears in a recent paper by [5] and more generally in the working paper by [11]. The basic logic in the latter paper works as follows - each principle offers a contract with a message space consisting of all the actions that he could take. He then asks the agents to tell him what to do. If all but one of the agents who interact with him names the same action, the principle takes that action, otherwise he takes some default action. It pays the agents to agree with everyone else, because they expect everyone to agree and accomplish nothing by deviating. If all principals offer this contract, then every agent makes the same report to any given principle telling the principal to take some action that, along with the instructions given to the other principals, provides the principle a payoff at least as large as his minmax payoff. If any principle deviates and offers any alternative mapping from messages to outcomes, the agents instruct the other principals to minmax the deviator.

In a very specialized environment, [5] makes an even simpler argument. Working in an environment in which agents take actions on behalf of principles (as in, for example, [3]), Katz imposes enough quasi-linearity and separability such that for any fixed action, the principals can offer the agent a contract such that the optimal effort for the agent under that contract implements the desired action and provides the agent exactly his reservation payoff. Now take any collection of actions for the principles that provides each principal at least his or her minmax payoff. Each principal then offers the agent a contract with a binary message. If the agent sends the message 1, the principle offers the agent a contract that implements his part of the collusive outcome. If the agent sends the message zero, the principle offers a contract that implements the action that minmaxes the other principal. Each agent sends his or her principal the message 1 as long as his principle offers this contract - deviations cause agents to send message zero, and lead to punishment.

One objection to contracts like this, and the folk theorems that they generate, is that they rely heavily on agents coordinating their messages. Yamashita's paper shows this in the most striking way. Not only must agents coordinate on the same message, but the must also coordinate on the message that the principals believe is driving the game before they offer their contracts. Not only is such a strong reliance on equilibrium selection sensitive to common knowledge assumptions, it is also very sensitive to possible collusion or unmodelled communication between agents. All the arguments in this literature basically allow agents to tell principals what they should do. Katz's argument deals with a somewhat simpler environment in which a single agent is hired to carry out an action on the part of the principal. The environment satisfies the assumptions in [4], so that coordinated actions by the principals can be carried out by having each principal offer the agent a menu of options, as they would in common agency. Agents punish deviations by changing the choice they make from this menu. Since agents must be indifferent between the choices in the menu to make this work, the environment is very restricted. Indifference again means that principals must rely on agents choosing appropriately among actions to which they are indifferent.

The literature that most closely resembles what we do uses more general commitment devices than those that can be implemented by hiring and communicating with self interested agents. The general approach is illustrated in [5] who essentially treat commitment devices as if they were messages in a mechanism design problem. Players in a game submit their commitment devices to a process that acts as the principal would in mechanism design, and resolves the commitment devices to an unambiguous outcome. It eliminates (by definition) any problem with circularity or ambiguity of contracts that we discussed above. They illustrate that for any joint distribution of actions for the players that gives each player his minmax payoff, the 'principal' can design a set of commitment devices such that is is an equilibrium for players to input a set of devices that support that joint distribution as an equilibrium outcome.

We focus here on a different issue, which is to describe a set of commitment devices that are perhaps more natural than those described in [5] and which will 'work' no matter what game the players are involved with. In this sense, our paper is more closely related to the paper by [10]. He models the set of commitment devices as a set of computer programs. The programs are submitted to a central computer which then uses them to determine an array of actions for the players. The important point that he illustrates is that these computer programs can use other players' programs as bits of data they can use to commit themselves to actions. He shows that joint distributions of actions in which all players actions are independent can be supported as equilibrium outcomes in the game in programs provided that each player receives his minmax payoff. The way this is done is to have each program implement some action, say the cooperative action in the prisoner's dillemma, provided that the other player's program is syntactically identical to his. To be syntactically identical, the other program must specify cooperation if the other program is syntactically identical, etc.

Our approach is quite similar to this. Intuitively we replace the computer that Tennenholtz uses to implement the outcome with a legal system. Beyond this, there are a couple of more important differences. First, the problem of designing commitment devices is a problem in mechanism design.

If the mechanism designer can compel participation, then there is no real need for commitment devices at all, at least with perfect information. The mechanism designer can simply instruct players which actions to use. Commitment devices only play a role in situations in which the mechanism designer is unable to force players to take actions, and instead has to reply on the players themselves to enforce cooperation. We accomplish this by allowing players to write contracts that specify a subset of all their feasible actions from which they must ultimately make a choice. We then assume that if more than one player has specified a contract that leads to a subset, then actions will be selected from these subsets non-cooperatively. Contracts uniquely determine sets without any circularity or ambiguity, while actual outcomes might require additional choices. This device means that a player can write a contract that specifies that he reserves the right to select his action ex post no matter what contracts the others use. This ensures that all players are always willing to participate in the contracting process. Of course, this approach also allows participants to commit themselves to the way they will respond to non-participants. This is the sense in which the mechanism designer uses the players themselves to enforce cooperation.

Secondly, we deal with randomization differently than [10]. As he does, we allow each player to use the syntax of the other player's contract to determine his own action. To make it possible to check this syntax, we don't allow players to write contracts that involve infinite expansions like π, or to use irrational numbers that have to be checked against an uncountable set of possibilities. Instead, we require that contracts consist of a finite set of words (in a language that may contain a countable set of words). Since we ultimately require players to take actions independently, as Tennenholtz does, this restricts us to contracts that always specify pure actions. Our characterization theorems are restricted to pure actions as a result. We don't view this as a critical restriction for two reasons. First, we can view the set of pure actions over which our players contract to be a finite approximation of the uncountable set of mixtures over some underlying set of pure actions. In this sense all of our theorems can be interpreted as approximation theorems in an appropriate way. Secondly, if the contracts are viewed as messages in an underlying mechanism design problem, for example by viewing the mechanism designer as the government, and the mechanism as the law, then the law could in principle use the government to further correlate the actions of the agents using some public randomizing device.

Finally, we show how to apply the approach to problems in which the players are imperfectly informed, or in which they interact with agents who have private information.

3. The Language and the Gödel Coding

We consider a formal language, which is sufficiently rich to allow its user to state propositions in arithmetic. Furthermore, the set of statements in this language is closed under the finite applications of the Boolean operations: $\urcorner$, $\vee$, and $\wedge$. This implies that one can express, for example, the following statement:

$$\forall n, x, y, z \left\{ [(n \geq 3) \vee (x \neq 0) \vee (y \neq 0) \vee (z \neq 0)] \rightarrow (x^n + y^n \neq z^n)] \right\}.$$

In addition, one can also express statements in the language that involve any finite number of free variables. For example, "x is a prime number" is a statement in the language. The symbol x is a free variable in the statement. Another example for a predicate that has one free variable is "$x < 4$." One can substitute any integer into x and then the predicate is either true or false. This particular one is true if $x = 0, 1, 2, 3$ and false otherwise.

Let $\mathfrak{L}$ be the set of all formulas of the formal language. Each of its element is a finite string of symbols. It is well known that one can construct a one-to-one function $\mathfrak{L} \to \mathbb{N}$. Let $[\varphi]$ be the value of this function at $\varphi \in \mathfrak{L}$, and call it the Gödel Code of the text φ.

Definition 3.1. The function $f : \mathbb{N}^k \to 2^{\mathbb{N}}$ is said to be *definable* if there exists a first-order predicate ϕ in $k+1$ free variables such that $b \in f(a_1, ..., a_k)$ if and only if $\phi(a_1, ..., a_k, b)$ is true.

In the definition, the mapping f is a correspondence from $\mathbb{N}^k$ to $\mathbb{N}$. Of course, if $f(n)$ is a singleton for all $n \in \mathbb{N}^k$, then f is a function. We illustrate the previous definition with an example.

Example. Consider the following function defined on $\mathbb{N}$:

$$f(a) = \begin{cases} 0 & \text{if } a \text{ is an even number,} \\ 1 & \text{if } a \text{ is an odd number.} \end{cases}$$

We show that this function is definable by constructing the corresponding predicate ϕ.

$$\phi(x, y) \equiv \{\{y = 1\} \wedge \{y = 0\}\} \vee \{\exists z : 2z = y + x\}.$$

Notice that ϕ indeed has two free variables. (The variable z is not free because there is a quantifier front of it.) The first part of ϕ states that y is either one or zero. The second part says that $x + y$ is divisible by two. Notice that $f(a) = 0$ if and only if $\phi(a, 0)$ is true. To see this, first notice that $\phi(a, b)$ is false whenever $b \notin \{0, 1\}$. (This is because the first part of ϕ requires b to be zero or one.) If $b = 0$ then $\phi(a, 0)$ is indeed true. If $b = 1$, then the second part of ϕ becomes false because $a + b$ is an odd number.

4. A Normal Form Contracting Game

Suppose there are m players. Each player has a finite action space A_i. The payoff of Player i is $u_i(a_1, \ldots, a_m)$. We use the conventional notation that $u_i(a_i, a_{-i})$ is the payoff to player i if he takes action a_i while the other players take action a_{-i}. Each player simultaneously submits a *contract,* which is a definable correspondences from $\mathbb{N}^m$ to $2^{\mathbb{N}}$, where 'definable' is to be understood in the sense of Definition 3.1. The correspondences, as describe above, uniquely and unambiguously determine an array of sets from which players then select actions non-cooperatively. **At stage two, players take actions simultaneously from subsets of their actions spaces. These subsets are determined by the first-stage contracts as follows.** If at stage one player j submitted contract c_j $(j = 1, ..., m)$, then player i can only take action a^i_k at stage two if $k \in c_i([c_1], ..., [c_m])$. We restrict attention to pure-strategy subgame perfect equilibra of this game.

Our objective is to prove a folk theorem for this contracting game. The lowest payoff for any player in any pure strategy equilibrium of the game in which players choose actions from A is

$$u_i^* = \min_{a_{-i} \in A_{-i}} \max_{a_i \in A_i} u_i \left(a_i, a_{-i} \right),$$

Let a_j^* be any one of the actions that j uses to attain his minmax payoff. Let us fix an action $a_{j_i}^i$ for player i, such that,

$$\left(a_{j_1}^1, ..., a_{j_m}^m \right) \in \arg\min_{a_{-i}} u_j \left(a_j^*, a_{-j} \right).$$

That is, $a_{j_i}^i$ is the action that player i uses to punish player j. In addition, define $i_i = 1$ for all $i \in \{1, ..., m\}$.

Theorem 4.1. *Let $\left\{ a_{k_1}^1, \ldots, a_{k_m}^m \right\}$ be any array of actions. These actions are supportable as an equilibrium outcome in the contracting game with pure strategy SPNE if and only if $u_i(a) \geq u_i^*$ for each i.*

Before we proceed with the proof of the theorem, we recall two pieces notations from the introduction. First, if $n \in \mathbb{N}$ then $< n >$ denotes the text whose Gödel code is n. That is, $[< n >] = n$. Second, for any text φ, let $\varphi^{(n_1, \ldots, n_k)}$ denote the statement where if the letter x_i stands for a free variable in φ then x_i is evaluated at n_i in φ for $i = 1, ..., n$. For example, if φ is $x > y$ and $n = 2$, then $\varphi^{(n)}$ is $2 > y$. Consider now the following text in one free variable: $< x >^{(x)}$. One can evaluate this statement at any integer. Since the Godel coding was a bijection $< n >$ is a text for each $n \in \mathbb{N}$. In addition, $\varphi^{(n)}$ is defined for all φ and n. In addition, it is a well-known result in Mathematical Logic, that if $f(n) = \left[< n >^{(n)} \right]$, then f is a definable function.

Proof. First, we prove the only if part. Fix an equilibrium in the contracting game. Let c_j denote the equilibrium contract of player j $(j = 1, ..., m)$ and let u_i denote player i's equilibrium payoff. Notice, that player i can always offer a contract that does not restrict his action space. That is, he can offer $\overline{c} : \mathbb{N}^m \to \mathbb{N}$, such that $\overline{c}(n_1, ..., n_m) = \mathbb{N}$ for all $(n_1, ..., n_m) \in \mathbb{N}^m$. The contract $\overline{c}$ is obviously definable. [3] We show that if $u_i < u_i^*$, player i can profitably deviate at the first stage by offering $\overline{c}$ instead of c_i. Let $\widetilde{c}_j = c_j$ if $j \neq i$ and $\widetilde{c}_i = \overline{c}$. Let $\widetilde{A}_j = \left\{ a_k^j : k \in \widetilde{c}_j \left(\left[\widetilde{c}_1 \right], ..., \left[\widetilde{c}_m \right] \right) \right\}$. That is, $\widetilde{A}_j$ is the action space of player j in the subgame generated by the contract profile $(\widetilde{c}_1, ..., \widetilde{c}_m)$. Also notice that $\widetilde{A_i} = A_i$. The payoff of player i in any pure strategy equilibrium of this subgame is weakly larger than

$$\min_{a_{-i} \in \widetilde{A}_{-i}} \max_{a_i \in A_i} u_i \left(a_i, a_{-i} \right) \geq \min_{a_{-i} \in A_{-i}} \max_{a_i \in A_i} u_i \left(a_i, a_{-i} \right).$$

The weak inequality follows from $\widetilde{A}_j \subseteq A_j$ for all j. Therefore, player i can always achieve his pure minmax value by offering the contract $\overline{c}$.

[3]For example, the predicate

$$\{x_1 = x_1\} \wedge ... \wedge \{x_m = x_m\} \wedge \{y = y\}$$

defines $\overline{c}$. That is, for all $y \in \mathbb{N}$ the predicate is true no matter how the free variables are evaluated.

For the if part, consider the following contract of Player i, $c^i_{x_i,x_{-i}}$, in m free variables:

$$c^i_{x_1,\ldots,x_m}\left(\left\{\left[c^j\right]_{j\neq i}\right\}\right)=$$

$$\left\{\begin{array}{ll} k_i & \text{if } \left|\left\{k:\left[<x_k>^{(x_1,\ldots,x_m)}\right]\neq\left[c^k\right]\ \right\}\right|\neq 1, \\ j_i & \text{if } \left\{k:\left[<x_k>^{(x_1,\ldots,x_n)}\right]\neq\left[c_k\right]\ \right\}=\{j\} \end{array}\right. \tag{4.1}$$

This contract with free variables is a definable function with free variables from $\mathbb{N}^{m-1}$ to $\mathbb{N}$ as long as the actions are replaced with their indices.

The expression (4.1) is not a contract, but rather a contract with free variables. Each such expression has a Godel code, so let $\gamma_i=\left[c^i_{x_1,\ldots,x_m}\right]$. The functions $\left\{c^i_{\gamma_1,\ldots,\gamma_m}\right\}_i$ have no free variables, so they constitute a set of contracts. We will now show that $\left\{c^i_{\gamma_1,\ldots,\gamma_m}\right\}_{i=1}^m$ constitutes an equilibrium profile of contracts which support the outcome $\left\{a^1_{k_1},\ldots,a^m_{k_m}\right\}$. First observe what happens when all players use contract $c^i_{\gamma_1,\ldots,\gamma_m}$. Notice that

$$c^i_{\gamma_1,\ldots,\gamma_m}\left(\left\{\left[c^j\right]_{j\neq i}\right\}\right)=\left\{\begin{array}{ll} k_i & \text{if } \left|\left\{k:\left[<\gamma_k>^{(\gamma_1,\ldots,\gamma_m)}\right]\neq\left[c_k\right]\ \right\}\right|\neq 1, \\ j_i & \text{if } \left\{k:\left[<\gamma_k>^{(\gamma_1,\ldots,\gamma_m)}\right]\neq\left[c_k\right]\ \right\}=\{j\}. \end{array}\right.$$

Player i needs to check whether the Godel code of$<\gamma_k>^{(\gamma_1,\ldots,\gamma_m)}$ is equal to the Godel code of $c^k_{\gamma_1,\ldots,\gamma_m}$. The integer γ_k is the Godel code of the contract with free variable $c^i_{x_1,\ldots,x_m}$. Player i's contract says to take this contract with free variable, fix the free variables at $\gamma_1,\ldots,\gamma_m$ (which gives the contract $c^k_{\gamma_1,\ldots,\gamma_m}$), then evaluate its Godel code. This is what is to be compared with the Godel code of the contract offered by k. Of course, these are the same. Since this is the case for all $m-1$ of the other players, player i ends up taking action a_i. So these contracts support the outcome we want if everyone uses them.

Player j can deviate to any definable contract mapping $\mathbb{N}$ into $\mathbb{N}$. However, any such contract will have a different Godel code, and so will induce the punishment $\left\{a^i_{j_i}\right\}_{i\neq j}$ from the other players. Since $u_j(a)\geq u^*_j$ this deviation will be unprofitable. ■

One might argue that restricting the space of contracts to be definable functions of Godel codes is both arbitrary and unnatural. Indeed, there is no reason for a judge to interpret a contract as a description of a mapping from the Godel codes of the contracts offered by the other players to the actions space of the player. For that matter, the judge might not even know about the Godel coding. It is important to note that the salient feature of definable contracts is that they can be written as texts that use a finite number of words in a formal language. The set of finite texts seems a very natural description of the set of feasible contracts. In fact, from this perspective it seems that *any* reasonable description of the set of feasible contracts should allow any such text.

The complication with such a broad description of the set of contracts is that to properly define a game, one must fully describe the mappings from profiles of texts into payoffs. Many texts will be complete nonsense and some modelling decision has to be taken about how these would translate into actions and payoffs. The contracts that we specify above are definable texts that have two advantages in this regard. First, since every finite text has a Godel code, they tie down the action of the player who offers such a contract even if the other players in the game offer contracts involving texts that make no economic sense. Furthermore, if all players offer contracts from the set we specify, an outcome for every player is uniquely determined. So no matter how ambiguous outcomes are set when players offer non-sense texts as contracts, the equilibrium outcome we describe above will persist.

From the perspective of the judge who has never heard of a Godel code, Theorem 4.1 and the theorems that follow have a kind of normative implication. This is simply that self or cross referential contracts that to involve an infinite regress can nonetheless be unambiguously written using finite texts.

Generalizations.— Everything about this theorem involves pure strategies. This imposes limits on its application. Next, we discuss how to extend our result to the case when players can mix over their restricted action space at the second stage of the game but cannot randomize over the contracts they offer at the first stage. Allowing such mixing expands the set of payoff profiles that can be supported by equilibria for two reasons. First, since players can randomize certain convex combinations of payoff profiles can now be supported. Second, players can use mixing when punishing a deviator, and hence the minmax value of the players will be smaller.

Formally, for all $S=\times_i S_i$, $S_i \subset A_i$, define a game, G_S, where the action space of player i is S_i, and the payoff function of player i is u_i restricted S. Let $E(S)$ denote the set of mixed equilibria in G_S. Define the minmax value of player i, u_i^*, as

$$u_i^* = \min_{\substack{S_{-i}\subset A_{-i} \\ S_{-i}=\times_{j\neq i} S_j}} \max_{S_i \subset A_i} \min_{\sigma \in E(S_{-i}\times A)} \int u_i(a)\, d\sigma(a).$$

The idea is that in the contracting game, players can restrict their action spaces arbitrarily, hence, when they punish player i they can choose S_{-i} arbitrarily. On the other hand, their second-stage actions must be best responses, and that is why we have to consider equilibrium payoffs in the restricted game. An argument identical to the proof of Theorem 1 shows that the random allocation $\sigma \in \Delta(A)$ can be supported as an equilibrium if

(i) $\exists S_i \subset A_i$ for all i, such that $\sigma \in E(\times_i S_i)$, and

(ii) $\int u_i(a)\, d\sigma(a) \geq u_i^*$ for all i.

What happens if players are allowed to randomize over the contracts they offer? It is possible to show that part (i) can be completely relaxed. That is, the distribution over the outcomes does not have to be an equilibrium in a G_S, and it does not even have to be generated by independent randomizations of the action spaces of the players. The construction of mixed equilibria in our contracting game that supports correlated outcomes is entirely based on Kalai et.al. (2008). The authors consider a two-person game similar to ours. Instead of taking actions, players submit commitment devices from a certain set. The devices then determine the action profile. The authors construct a set of devices such that any individually rational correlated outcome can be implemented as a mixed equilibrium in the game. (That is, although the players mix independently over their devices, the distribution over the actions profiles will be correlated.) It is not hard to extent their results to our model and obtain the following theorem.

Theorem 1. Suppose that $\sigma \in \Delta(A)$, and $\sigma(a) \in \mathbb{Q}$ for all $a \in A$. The σ can be supported as a mixed-strategy equilibrium outcome in the contracting game if and only if $\int u_i(a)\, d\sigma(a) \geq u_i^*$ for all $i \in \{1, ..., m\}$.

Another question is why we use definable functions as opposed to programs or Turing machines. One might want to require that the contracts must be computable and assume that the set of available contracts is the set (or a subset) of Turing machines. In such a model, if player i $(i = 1, 2)$ chooses machine τ_i, then τ_i runs on the description of τ_j, and the output will be a subset of the action space of player i. It is well-known, that one can construct self- and cross-referential contracts (machines) in this space too.[4] In fact, this construction is essentially identical to our construction of cross-referential definable functions. Most importantly, the equilibrium contracts we construct to support individually rational allocations are, in fact, recursive functions, and hence they are computable by Turing machines. Therefore, if the reader insists on computability, he can restrict attention to the space of Turing machines.

There are, however, several advantages of our approach over modelling contracts with Turing machines. Let us explain.

1. Turing machines do not always halt. Therefore, it is not clear how one can define our contracting game properly. In particular, it is not streighforward how to define the restriction on the action space of a player, if his machine does not halt. One might suggest that if a player submitts a Turing machine that does not halt, then define his second stage action space to be the whole space (or a default subset). We think that such a definition might be arbitrary. In addition, the problem whether or not a Turing machine halts is an undecidable problem. That is, there is no Turing machine which can determine whether a player deviated or not. An alternative way to handle the halting problem is to restrict the space of Turing machines to be the set of machines that always halts. We find such restrictions also arbitrary. Instead of restricting the space of recursive functions, we expanded it to be the set of definable functions and avoided the halting problem that way.

2. Another problem with Turing machines is that they can only condition on the actual description of the machines submitted by the other players but cannot condition on the functions what the machines compute. Take the example of the prisoner dilemma. It is possible to construct a Turing machine, τ, such that

$$\tau([\tau_2]) = \begin{cases} C & \text{if } [\tau_2] = [\tau] \\ D & \text{otherwise.} \end{cases}$$

[4] Such machines were constructed even in the context of Game Theory, see Anderlini 1990 and Canning 1992.

The problem is that if player 2 submitts a machine, say τ', which is computationally equivalent with τ, but has a different description, then player 1 would defect. In fact, it is not possible to construct a machine which does not suffer from this problem. That is, the equilibrium contract is sensitive to the way it is written. A player does not only require the other player to have the right intentions, but also requires him to express himiself in a unique way. This feature makes us doubt whether machine contracts is the right way of modeling contractible contracts.

We avoid such problems with definable functions. Indeed, it is possible to express contracts that do not condition on the actual way the other contract is written, but on the function itself that the other contract describes. Consider

$$c_1([c_2]) = \begin{cases} C & \text{if } c_2^* \Leftrightarrow c_2, \\ D & \text{otherwise.} \end{cases}$$

The contract c_1 is obviously definable, but does not condition on the actual form of c_2. As long as c_2 represents the same function as c_2^*, cooperation is prescribed.

5. Contracting in a Bayesian Environment

This section shows how to extent the result of the previous section to games with incomplete information. We shall show that the set of allocations that can be supported by contractible contracts is limited because the contracts reveal public information about the players and this makes it harder to punish deviations.

The model is the same as in the previous section, with the addition of player types. There are n players. Player i's actions space is a finite set denoted by A^i. Each player i has a type t_i drawn from a finite set T^i. The joint distribution types is common knowledge. The payoff of player i is $u_i(a_i, a_{-i}, t)$ where $t \in T^1 \times \cdots \times T^n$.

Our goal is to characterize the set of equilibria of the contracting game (to be defined formally later) in this physical environment. Our strategy is to define a mechanism design problem, and then to show that the set of allocations that can be implemented by these mechanisms are identical to the set of equilibrium outcomes in the contracting game. To this end, we first describe a class of mechanisms.

Public Message Mechanisms.— Consider the following class of three-stage mechanisms. First, players simultaneously decide whether or not to participate in the mechanism. Those players who decide to participate send *public* messages to the mechanism designer (MD). Player i's message space is a countable set. At the same time, players who do not participate publicly submit a function which will impose a restriction on their action spaces as a function of the public messages sent by the participating players. (It is important to note that these functions must be submitted before players can observe the public messages.) At the second stage, the MD can arbitrarily restrict the action spaces of those players who have decided to participate at the first stage, as a function of the messages. These restrictions, however, cannot depend on the functions submitted at the first stage by the nonparticipants. Finally, at the third stage, players take actions from their restricted actions spaces simultaneously.

We restrict attention to deterministic mechanisms and pure-strategy Weak Perfect Bayesian Nash Equilibria. By standard arguments in mechanism design, without loss of generality, one can restrict attention to mechanism-equilibrium pairs in which (i) each player participates, (ii) the messages space of player i is the set of elements of a certain partition of his type space, and (iii) each player reports the element of his partition which contains his true type at stage one.

In what follows we characterize the allocations that can be implemented by these mechanisms by constraints. There are three sets of constraints. The first one is the participation constraints which guarantees that each player prefers to participate in the mechanism independently of his type. The second one is the incentive compatibility constraints which guarantees that each player reports the element of the partition of his type space which contains his type. Finally, the third sets of constraints guarantee that each player takes an action at stage three which is a best response against the strategies of the others.

Before we proceed with characterizing these constraints, we introduce some notations. Let $\tau_i : T^i \to 2^{T^i}$ denote a partition of player i's type space. That is, $t^i \in \tau_i\left(t^i\right)$ for all t^i, and if $t^i \in \tau_i\left(t^{i\prime}\right)$ then $\tau_i\left(t^i\right) = \tau_i\left(t^{i\prime}\right)$. Let τ, τ_{-i}, and τ_{-ij} denote $\times_{i=1}^{n}\tau_i$, $\times_{j\neq i}\tau_j$, and $\times_{k\neq i,j}\tau_k$ respectively. Let $r^i\left(t\right) \subset A^i$ denote the restricted action space of player i if each player participates, and the message sent by player j is $\tau_j\left(t^j\right)$. That is, r^i must be measurable with respect to τ. This means that $r^i\left(t\right) = r^i\left(t'\right)$ whenever $\tau_i\left(t^i\right) = \tau_i\left(t^{i\prime}\right)$ for all i. Similarly, $r_j^i\left(t^{-j}\right)$ denotes the restriction on the actions space of player i if all players but player j participate, and the message sent by player q is $\tau_q\left(t^q\right)$. The function $r_j^i\left(t^{-j}\right)$ is measurable with respect to τ_{-j}. Finally, let

$$F_i^{\tau} = \left\{ f_i \middle| \; f_i : \tau_{-i}\left(T^{-i}\right) \to 2^{A^i} \right\},$$

where $\tau_{-i}\left(T^{-i}\right) = \left\{\tau_{-i}\left(t^{-i}\right) : t^{-i} \in T^{-i}\right\}$. The set F_i^{τ} is the action space of player i at stage one if he does not participate in the mechanism. If player i submits $f_i\left(\in F_i^{\tau}\right)$ and player j reports $\tau_j\left(t^i\right)$ for $j \neq i$, then player i's restricted action space at stage three is $f_i\left(\tau_{-i}\left(t^{-i}\right)\right)$. Let s^i denote the strategy of player i at stage three if each player participates, that is, $s^i : T^i \times T^{-i} \to A^i$, such that $s^i\left(t^i, t^{-i}\right) \in r^i\left(t\right)$ for all t, and s^i is measurable with respect to τ_{-i}. That is, $s^i\left(t^i, t^{-i}\right) = s^i\left(t^i, t'^{-i}\right)$ if $\tau_{-i}\left(t^{-i}\right) = \tau_{-i}\left(t'^{-i}\right)$. Similarly, s_j^i denotes the strategy of player i at stage three if all players but player j participates. That is, $s_j^i : T^i \times T^{-ij} \times F_j \to A^i$ such that $s^i\left(t^i, t^{-ij}, f_j\right) \in r_j^i\left(t^{-j}\right)$, and s_j^i is measurable with respect to τ^{-ij}.

Best Response. In the last stage, players optimally choose their action given the others' strategies, and truthful reports at stage two. That is, for all i, $t \in T$:

$$s^i\left(t\right) = \arg\max_{a \in r^i(t)} E_{t^{-i}}\left(u_i\left(a, s^{-i}\left(t\right), t\right) : t^i, \tau^{-i}\left(t^{-i}\right)\right). \tag{5.1}$$

This constraint ensures that $\left\{s^i(t)\right\}_i$ is a Bayesian Equilibrium in the game where player i, with type t^i, observes $\tau(t)$.

Incentive Compatibility. At stage two, each player reports the element of the partition of his type space truthfully, given the strategies in the last stage and that everybody else reports truthfully. That is, for all i, t^i, and $t^{i\prime} \in T^i$:

(5.2)
$$E_{t^{-i}}\left(u_i\left(s\left(t,\tau\right),t\right):t^i\right) \geq E_{t^{-i}}\left(\max_{a\in r^i\left(t^{i\prime},t^{-i}\right)} E_{t^{-i}}\left(u_i\left(a,s^{-i}\left(t^{i\prime},t^{-i}\right),t\right):t^i,\tau_{-i}\left(t^{-i}\right)\right):t^i\right).$$

Participation Constraint. At stage one, players prefer to participate to opting out, given that everybody else participates, everybody reports truthfully, and the strategies at the final stage. To characterize this constraint, we first compute the payoff of player i with type t^i if he does not participate. For all i, t^i consider

$$\max_{f_i\in F_i} E_{t^{-i}}\left(\max_{a\in f_i\left(\tau_{-i}\left(t^{-i}\right)\right)} E_{t^{-i}}\left(u_i\left(a,s_i^{-i}\left(t^{-i},f_i\right),t\right):t^i,\tau_{-i}\left(t^{-i}\right)\right):t^i\right).$$

Let us denote the value of this problem by $\underline{u}_i\left(t^i\right)$. Then, player i with type t^i prefers to participate if and only if

$$E_{t^{-i}}\left(u_i\left(s\left(t,\tau\right),t\right):t^i\right) \geq \underline{u}_i\left(t^i\right). \tag{5.3}$$

Let $\alpha : T \to A$ be an allocation. This allocation can be implemented by the mechanism if there are partitions of the type spaces, $\{\tau_i\}$, restrictions of the MD, $\left\{r^i, r^i_j\right\}$, and the strategies $\left\{s^i, s^i_j\right\}$ such that $s(t) = \alpha(t)$ and the three sets of constraints (5.1), (5.2), and (5.3) are satisfied.

The Contracting Game.— The contracting game is the same as in the previous section. The game has two stages. In the first stage, players offer contracts simultaneously. A contract of player i is a definable function from $\mathbb{N}^n$ to subsets of A_i. The ith coordinate of the domain is the Godel code of the definable function offered by the player i. At stage two, players take actions simultaneously from the subsets of their actions spaces which is specified by the contracts. If player q offers a definable function c^q, then $c^i\left(\left[c^1\right],...,\left[c^n\right]\right) \subset A^i$ is the subset of the action space of player i pinned down by the contracts. Of course, at the second stage, players make inferences about the types of the other players from the contracts they have submitted. We restrict attention to pure-strategy Weak Perfect Bayesian Equilibria of this game.

Theorem 2. An allocation is implementable with the public-message mechanism if and only if it is implementable as an equilibrium in the contracting game.

Proof of the "if" part. First, let us fix a mechanism and an equilibrium of it. Let us denote the partitions of the type spaces generated by the mechanism by $\{\tau_i\}_i$, the restrictions by $\left\{r^i, r^i_j\right\}_{i,j}$, and the strategies of the players by $\left\{s^i, s^i_j\right\}_{i,j}$. Consider now the following contract in $|T|$ free variables:

$$c^{t_i}_{\left(x^{t_j}_j\right)_{j,t_j}}\left(c^1,...,c^n\right)$$
$$= \begin{cases} r^i_j(t) & \text{if } \forall k \exists ! \tau_k \in \tau\left(T^k\right) \text{ s.t. } \left[< x^{t_k}_k >^{(x)}\right] = \left[c^k\right] \text{ if } t^k \in \tau_k, \\ r^i_j\left(t^{-j}\right) & \text{if } \left\{k : \nexists ! \tau_k \in \tau\left(T^k\right) \text{ s.t. } \left[< x^{t_k}_k >^{(x)}\right] = \left[c^k\right] \text{ if } t^k \in \tau_k\right\} = j, \\ A_i & \text{otherwise and if } k+1 > k \text{ if } k \in H\left(t_i\right), \end{cases}$$

where x denotes $\left(x^{t_j}_j\right)_{j,t_j}$ and $H\left(t^i\right) = \left\{j : \tau\left(t^j\right) = \tau\left(t^i\right)\right\}$. The last statement is in the third line is always true. Such a statement, however, makes it possible that a player with two different types offers two different but computationally equivalent contracts. Let $\gamma^{t_i}_i$ denote the Godel Code of this contract and let $\gamma = \left(\gamma^{t_i}_i\right)_{i,t^i}$. The equilibrium contract offered by player i with type t^i will be: $c^{t_i}_\gamma$. Then

$$c^{t_i}_\gamma\left(c^1,...,c^n\right)$$
$$= \begin{cases} r^i_j(t) & \text{if } \forall k \exists ! \tau_k \in \tau\left(T^k\right) \text{ s.t. } \left[< \gamma^{t_k}_k >^{(\gamma)}\right] = \left[c^k\right] \text{ if } t^k \in \tau_k, \\ r^i_j\left(t^{-j}\right) & \text{if } \left\{k : \nexists ! \tau_k \in \tau\left(T^k\right) \text{ s.t. } \left[< \gamma^{t_k}_k >^{(\gamma)}\right] = \left[c^k\right] \text{ if } t^k \in \tau_k\right\} = j, \\ A_i & \text{otherwise and if } k+1 > k \text{ if } k \in H\left(t_i\right), \end{cases}$$

Notice that $< \gamma^{t_q}_q >^{(\gamma)} = c^{t_q}_\gamma$. Therefore, the previous contract can be rewritten as

$$c^{t^i}_\gamma\left(c^1,...,c^n\right) \tag{5.4}$$
$$= \begin{cases} r^i_j(t) & \text{if } \forall k \exists ! \tau_k \in \tau\left(T^k\right) \text{ s.t. } \left[c^{t_k}_\gamma\right] = \left[c^k\right] \text{ if } t^k \in \tau_k, \\ r^i_j\left(t^{-j}\right) & \text{if } \left\{k : \nexists ! \tau_k \in \tau\left(T^k\right) \text{ s.t. } \left[c^{t_k}_\gamma\right] = \left[c^k\right] \text{ if } t^k \in \tau_k\right\} = j, \\ A_i & \text{otherwise and if } k+1 > k \text{ if } k \in H\left(t_i\right), \end{cases}$$

Next, we specify the strategies of the players in at the second stage. If for all j there is a $t^j \in T^j$ such that player j offeres a contract $c^{t_j}_\gamma$, then player i takes action $s_i(t)$. Suppose now that one player deviated, say player k, and he offered a contract c^k, and player j offered $c^{t^j}_\gamma$ for all $j \neq k$. Define $f_{c^k} : \tau$ as follows:

$$f_{c^k}\left(\tau_{-k}\left(t^{-k}\right)\right) = c^k\left(\left[c^k\right], \left[c^{t^{-k}}_\gamma\right]\right), \tag{5.5}$$

where $\left[c^{t^{-k}}_\gamma\right]$ denotes the vector of the Godel codes of players other than k. Then player *í*'s strategy is $s^i_k\left(t^i, t^{-ik}, f_k\right)$. Notice that by (5.4) these second-stage strategies are consistent with the restrictions imposed by the contracts.

We shall argue that the strategies described above constitute an equilibrium in the contracting game. First, the strategies $\left\{s^i\right\}_i$ are optimal in the second stage by (5.1). Hence, we only have to show that players do not have incentive to deviate at the contracting stage. Notice that if a player does not deviate then her payoff is the same as in the mechanism. Suppose now that player i with type t^i offers a contract c which is different from $c_\gamma^{t^i}$. We shall consider two cases. Case 1: $c = c_\gamma^{t^{i\prime}}$ but $\tau_i\left(t^i\right) \neq \tau_i\left(t^{i\prime}\right)$. Then, by the first line of (5.4) and by the definition of $\left\{s^j\right\}_j$, the deviator's payoff cannot exceed the payoff of player i with type t^i in the mechanism if she decided to report $\tau_i\left(t^{i\prime}\right)$ instead of $\tau_i\left(t^i\right)$ in the first stage. Since (5.2) holds, such a deviation is not profitable. Case 2: $c \neq c_\gamma^{t^{i\prime}}$ for all $t^{i\prime}$. (That is, player i offers an off-equilibrium contract.) Then, by the second line of (5.4), the restriction on player j's action space is $r_j^i\left(t^{-j}\right)$ $(j \neq i)$. The restriction on player i's action space is $c\left([c], \left[c_\gamma^{t^{-i}}\right]\right) = f_c\left(\tau_{-i}\left(t^{-i}\right)\right)$. That is, the restrictions are the same as if in the mechanism player i with would have not participated and submitted f_c at the first stage. In addition, the actions of players $-i$ at stage two are also the same as in the mechanism at stage three. Hence, by (5.3), such a deviation is not profitable. ■

The difficulty of proving the "only if" part of the theorem is the following. In the mechanism, even if a player does not participate, the MD can restrict the action space of the participating players only as a function of their messages. But these restrictions cannot depend on the non-participant player's function he submits at the first stage. This limits the severity of the punishment that players can impose on a deviator. Since contracts can explicitly depend on other contracts, one might think that a deviator in the contracting game can be punished as a function of his contract. That is, the restrictions on the action spaces of the players can depend on the deviator's contract. This observation suggests that the punishment for a deviation can be more severe in the contracting game than in the mechanism and, hence, a larger set of allocations cen be implemented by the contracting game. We show that, surprisingly, this is not true. More precisely, we show below that when player i deviates from his equilibrium contracts, players $-i$ cannot punish him more than by offering contracts which pin down a subset of A^{-i} which does not depend on the deviator's contract.

Consider an equilibrium in the contracting game. Denote the equilibrium contract of player i with type t^i by $\widetilde{c}_{t^i}$. Let us define a partition, τ_i, of player i's type space as follows: $\tau_i\left(t^i\right) = \left\{t^{i\prime} \in T^i : \widetilde{c}_{t^i} = \widetilde{c}_{t^{i\prime}}\right\}$. We shall also use the notation $c_{\tau_i(t^i)}$ for c_{t^i}. Denote the vector of equilibrium contracts of players $-i$ with type profile t_{-i} by $\widetilde{c}_{\tau_{-i}(t^{-i})}$. (That is, $\widetilde{c}_{t^{-i}}$ is a vector of definable functions whose coordinates are $\left\{\widetilde{c}_{\tau_j(t^j)}\right\}_{j\neq i}$, and $\widetilde{c}_{\tau_{-i}(t^{-i})} : \mathbb{N} \to A^{-i}$.) For all $\left(A_i^{\tau_{-i}(t_{-i})}\right)_{t^{-i}} \subset \left(A^i\right)^{|\tau_{-i}(T_{-i})|}$ define $S\left(\left(A_i^{\tau_{-i}(t_{-i})}\right)_{t^{-i}}\right)$ as follows:

$$\left\{\left(A_{-i}^{\tau_{-i}(t_{-i})}\right)_{t^{-i}} : A_{-i}^{\tau_{-i}(t_{-i})} \subset A^{-i},\ \exists c \text{ s. t. } c\left(\left[\widetilde{c}_{\tau_{-i}(t_{-i})}\right]\right) = A_i^{\tau_{-i}(t_{-i})},\ \widetilde{c}_{\tau_{-i}(t_{-i})}\left([c]\right) = A_{-i}^{\tau_{-i}(t_{-i})}\right\}.$$

Let us explain what it means that $\left(A_{-i}^{\tau_{-i}\left(t^{t_{-i}}\right)}\right)_{t_{-i}} \in S\left(\left(A_i^{\tau_{-i}(t_{-i})}\right)_{t_{-i}}\right)$. By the definition of S, there exists a contract, c, available for player i such that if the type profile of the other players is t_{-i}, then if player i offers c then his restricted action space will be $A_i^{\tau_{-i}\left(t^{-i}\right)}$ and players

$-i$'s restricted actions space will be $A_{-i}^{\tau_{-i}(t^{-i})}$. The following lemma is key to characterize the participation constraint.

Lemma 3. For all $i \in \{1, ..., n\} : \cap_{\left\{A_i^{\tau_{-i}(t_{-i})}\right\}_{t_{-i}}} S\left(\left\{A_i^{\tau_{-i}(t_{-i})}\right\}_{t_{-i}}\right) \neq \{\emptyset\}$.

Lemma 4. For all i: $\cap_{\left(A_i^{\tau_{-i}(t_{-i})}\right)_{t_{-i}}} S\left(\left(A_i^{\tau_{-i}(t_{-i})}\right)_{t_{-i}}\right) \neq \{\emptyset\}$.

Let us explain the statement of this lemma. Consider the case when only player i has a type. Denote the equilibrium contracts of players $-i$ is $\widetilde{c}$. In this case,

$$S(A_i) = \left\{A_{-i} : A_{-i} \subset A^{-i},\ \exists c \text{ such that } c([\widetilde{c}]) = A_i,\ \widetilde{c}([c]) = A_{-i}\right\}.$$

Suppose that $A_{-i} \in \cap_{A_i} S(A_i)$. This means that for all $A_i \subset A^i$, player i can offer a contract such that his restricted action space is A_i and the restricted action space of the other players is A_{-i}. Based on this observation it is easy to show that players $-i$ could modify their contracts such that whenever player i deviates, the restricted action space of players $-i$ is A_{-i}. That is, the restrictions following a deviation do not depend on the contract of the deviator.

In the general case, $\left(A_{-i}^{\tau_{-i}(t_{-i})}\right)_{t_{-i}} \in \cap_{\left(A_i^{\tau_{-i}(t_{-i})}\right)_{t_{-i}}} S\left(\left(A_i^{\tau_{-i}(t_{-i})}\right)_{t_{-i}}\right)$ implies that for all $\left(A_i^{\tau_{-i}(t_{-i})}\right)_{t_{-i}}$, player i can offer a contract such that such that if the type profile of the other players is t_{-i}, then player i's restricted action space will be $A_i^{\tau_{-i}(t_{-i})}$ and players $-i$'s restricted actions space will be $A_{-i}^{\tau_{-i}(t_{-i})}$. That is, player i can achieve any restriction he wants against any element of the intersection.

Before we prove the previous lemma, we sketch the proof for the case where only player i has a type. Suppose, by contradiction, that $\cap_{A_i \subset A^i} S(A_i) = \{\emptyset\}$. Then for all $A_{-i} \subset A^{-i}$ there exists an $A_i \subset A^i$ such that $A_{-i} \notin S(A_i)$. Therefore, one can construct a function, $f : 2^{A^{-i}} \to 2^{A^i}$, such that

$$\forall A_{-i} \subset A^{-i} : A_{-i} \notin S(f(A_{-i})).$$

Define c_x as follows:

$$c_x([c]) = f(\widetilde{c}([< x > (x)])).$$

Since f and $\widetilde{c}$ are definable functions, c_x is a definable function in one free variable. Let γ denote the Godel code of c_x. Then

$$c_\gamma([c]) = f(\widetilde{c}([c_\gamma])).$$

Notice that

$$\widetilde{c}([c_\gamma]) \in S(c_\gamma([\widetilde{c}]))$$

by the definition of S. On the other hand,

$$\widetilde{c}([c_\gamma]) \notin S(f(\widetilde{c}([c_\gamma]))) = S(c_\gamma([\widetilde{c}])),$$

a contradiction to the assumption that $\widetilde{c}$ was definable.

Proof. Suppose by contradiction that $\cap_{\left\{A_i^{\tau_{-i}(t_{-i})}\right\}_{t_{-i}}} S\left(\left\{A_i^{\tau_{-i}(t_{-i})}\right\}_{t_{-i}}\right) = \{\emptyset\}$. Then, for all $\left\{A_{-i}^{\tau_{-i}(t_{-i})}\right\}_{t_{-i}} \subset \left(A^{-i}\right)^{|\tau_{-i}(T_{-i})|}$ there exists an $\left\{A_i^{\tau_{-i}(t_{-i})}\right\}_{t_{-i}} \subset \left(A^{i}\right)^{|\tau_{-i}(T_{-i})|}$ such that $\left\{A_{-i}^{\tau_{-i}(t_{-i})}\right\}_{t_{-i}} \notin S\left(\left\{A_i^{\tau_{-i}(t_{-i})}\right\}_{t_{-i}}\right)$. Let us fix a function $f : 2^{\left(A^{-i}\right)^{|T_{-i}|}} \to 2^{\left(A^{i}\right)^{|T_{-i}|}}$ such that

$$\forall \left\{A_{-i}^{\tau_{-i}(t_{-i})}\right\}_{t_{-i}} \subset \left(A^{-i}\right)^{|\tau_{-i}(T_{-i})|} : \left\{A_{-i}^{\tau_{-i}(t_{-i})}\right\}_{t_{-i}} \notin S\left(f\left(\left\{A_{-i}^{t_{-i}}\right\}_{t_{-i}}\right)\right).$$

Let $f_{\tau_{-i}(t_{-i})}$ denote the projection of f corresponding to t_{-i}. That is, $f = \left\{f_{\tau_{-i}(t_{-i})}\right\}_{t_{-i}}$. Define c_x as follows:

$$c_x(c) = \begin{cases} f_{t'_{-i}}\left(\left\{\widetilde{c}_{\tau_{-i}(t_{-i})}\left([< x > (x)]\right)\right\}_{t_{-i}}\right) & \text{if } \exists t'_{-i} \in T^{-i} \text{ st. } c = \widetilde{c}_{\tau_{-i}\left(t'_{-i}\right)}, \\ A^i & \text{otherwise.} \end{cases}$$

Since f and $\widetilde{c}_{\tau_{-i}(t_{-i})}$ are definable functions, c_x is a definable function in one free variable. Let γ denote its Godel code. Then

$$c_\gamma(c) = \begin{cases} f_{\tau_{-i}\left(t'_{-i}\right)}\left(\left\{\widetilde{c}_{\tau_{-i}(t_{-i})}\left([c_\gamma]\right)\right\}_{t_{-i}}\right) & \text{if } \exists t'_{-i} \in T^{-i} \text{ st. } c = \widetilde{c}_{\tau_{-i}\left(t'_{-i}\right)}, \\ A^i & \text{otherwise.} \end{cases}.$$

Notice that

$$\left\{\widetilde{c}_{\tau_{-i}(t_{-i})}\left([c_\gamma]\right)\right\}_{t_{-i}} \in S\left(\left\{c_\gamma\left(\left[\widetilde{c}_{\tau_{-i}(t_{-i})}\right]\right)\right\}_{t_{-i}}\right) \tag{5.6}$$

by the definition of S. On the other hand,

$$\begin{aligned} \left\{c_\gamma\left(\left[\widetilde{c}_{\tau_{-i}(t_{-i})}\right]\right)\right\}_{t_{-i}} &= \left\{f_{t_{-i}}\left(\left\{\widetilde{c}_{\tau_{-i}\left(t'_{-i}\right)}\left([c_\gamma]\right)\right\}_{t'_{-i}}\right)\right\}_{t_{-i}} \\ &= f\left(\left\{\widetilde{c}_{\tau_{-i}(t_{-i})}\left([c_\gamma]\right)\right\}_{t_{-i}}\right), \end{aligned}$$

and therefore,

$$\left\{\widetilde{c}_{\tau_{-i}(t_{-i})}\left([c_\gamma]\right)\right\}_{t_{-i}} \notin S\left(\left\{c_\gamma\left(\left[\widetilde{c}_{\tau_{-i}(t_{-i})}\right]\right)\right\}_{t_{-i}}\right) \tag{5.7}$$

by the definition of f. Notice that (5.6) and (5.7) contradict to each others, and hence, $\left\{\widetilde{c}_{\tau_{-i}(t_{-i})}\right\}_{t_{-i}}$ were not definable functions. ■

Before we proceed with the proof of the "only if part" of the theorem, for all $i \in \{1, ..., n\}$ fix $\left\{B_{-i}^{\tau\left(t^{-i}\right)}\right\}_{t^{-i}}$ such that

$$\left\{B_{-i}^{\tau\left(t^{-i}\right)}\right\}_{t^{-i}} \in \cap_{\left\{A_i^{\tau_{-i}(t_{-i})}\right\}_{t_{-i}}} S\left(\left\{A_i^{\tau_{-i}(t_{-i})}\right\}_{t_{-i}}\right).$$

.

Proof of the "only if" part of Theorem 5. Fix an equilibrium in the contracting game and let $\widetilde{c}_{t_i}$ denote the contract offered by player i with type t_i. Recall the definition of τ_i: $\tau_i\left(t^i\right) = \left\{t^{i\prime} \in T^i : \widetilde{c}_{t_i} = \widetilde{c}_{t'_i}\right\}$. We shall also use the notation $c_{\tau_i(t_i)}$ for c_{t_i}. In addition, let $a^i\left(t^i, t^{-i}\right)$

denote the equilibrium strategy of player i with type t^i at the second stage if player j offered contract c_{t^j}. Obviously, a^i is measurable with respect to τ_{-i}. It must be the case that

$$a^i\left(t^i, t^{-i}\right) \in c_{t^i}\left(\left[c_{t^1}\right], ..., \left[c_{t^n}\right]\right). \tag{5.8}$$

In addition, let $a_j^i\left(t^i, t^{-ij}, c_j\right)$ denote the strategy of player i with type t^i if player q $(q \neq j)$ offered contract c_{t^q} and player j offered c_j at the first stage. Of course, a_j^i must be measurable with respect to τ_{-ij} and

$$a_j^i\left(t^i, t^{-ij}, c_j\right) \in c_{t^i}\left(\left[c_{t^1}\right], ..., \left[c_j\right], ..., \left[c_{t^n}\right]\right). \tag{5.9}$$

The functions $\left\{c_{t^i}, a^i, a_j^i\right\}$ fully describe the equilibrium in the contracting game.

In what follows we construct a public-message mechanism which supports the same allocation. The message space of player i is $\left\{\tau_i\left(t^i\right) : t^i \in T^i\right\}$. The restriction on player i's action space if each player participates and player j reports $\tau_j\left(t_j\right)$ is

$$r^i\left(t\right) = c_{t^i}\left(\left[c_{t^1}\right], ..., \left[c_{t^n}\right]\right). \tag{5.10}$$

Notice that r^i is measurable with respect to τ. The restriction on player i's action space if all players but player j participate and player q reports $\tau_q\left(t_q\right)$ is

$$r_j^i\left(t^{-j}\right) = B_{-j}^{\tau_{-j}\left(t^{-j}\right)}. \tag{5.11}$$

Notice that r_j^i is measurable with respect to τ_{-j}.

It remained to specify the players' strategies at the final stage. If each player participates and player j reported $\tau_j\left(t^j\right)$ then define player i's action, $s^i\left(t^i, t^{-i}\right) = a^i\left(t^i, t^{-i}\right)$. By (5.8) and (5.10) these strategies are consistent with the restrictions, that is, $s^i\left(t^i, t^{-i}\right) \in r^i\left(t\right)$.

Suppose now that player j deviates and submits a function $f_j \in F_j$ at the first stage of the mechanism. By the definition of $\left\{B_{-j}^{\tau_{-j}\left(t^j\right)}\right\}$ there exists a contract, c_{f_j}, such that

$$c_{f_j}\left(\left[c_{f_j}\right], \left[c_{t^{-j}}\right]\right) = f_j\left(\tau_{-j}\left(t^j\right)\right), \tag{5.12}$$

and

$$c_{t^{-j}}\left(\left[c_{t^{-j}}\right], \left[c_{f_j}\right]\right) = B_{-j}^{\tau_{-j}\left(t^{-j}\right)}. \tag{5.13}$$

Therefore, define the strategy of player i with type t^i at the last stage as $s_j^i\left(t^i, t^{-ij}, f_j\right) = a_j^i\left(t^i, t^{-ij}, c_{f_j}\right)$ if player q $(q \neq j)$ reported $\tau_q\left(t^q\right)$ at the first stage and player j submitted f_j. (This means that player j will be treated in the mechanism as if he deviated and offered c_{f_j} in the contracting game.) Notice that by (5.9), (5.11), and (5.13) these strategies are consistent with the restrictions, that is, $s\left(t^i, t^{-ij}, f_j\right) \in r_j^i\left(t^{-j}\right)$.

Since $s^i = a^i$ the strategies implement the same allocation in the mechanism and in the contracting game. We only have to show that the constraints (5.1), (5.2), and (5.3) are satisfied. Suppose that (5.1) is not satisfied for t^i, t^{-i} and there is an $a^i \in r^i\left(s\right)$ such that

$$E_{t^{-i}}\left(u_i\left(a^i, s^{-i}\left(t\right), t\right) : t^i, \tau^{-i}\left(t^{-i}\right)\right) > E_{t^{-i}}\left(u_i\left(s\left(t\right), t\right) : t^i, \tau^{-i}\left(t^{-i}\right)\right).$$

But then, by (5.10) player i with type t^i also could have taken a^i at the second stage if player j offered c_{t^j} at the first stage. Since $a^i = s^i$, the previous inequality implies that such a deviation would be profitable. This contradicts the assumption that a^i was an equilibrium strategy in the contracting game. Suppose now that (5.2) does not hold for $t^i \in T^i$, that is, there exists a $t^{i\prime} \in T^i$ such that

$$\begin{aligned} & E_{t^{-i}}\left(u_i\left(s\left(t,\tau\right),t\right):t^i\right) \\ < \quad & E_{t^{-i}}\left(\max_{a\in r^i\left(t^{i\prime},t^{-i}\right)} E_{t^{-i}}\left(u_i\left(a,s^{-i}\left(t^{i\prime},t^{-i}\right),t\right):t^i,\tau_{-i}\left(t^{-i}\right)\right):t^i\right). \end{aligned}$$

Since $a^i = s^i$, the left-hand-side of this equality is the interim payoff of player i with type t^i in the contracting game. In addition, by (5.10), the right-hand-side is the maximum payoff player i with type t^i can achieve in the contracting game if he offered $c_{t^{i\prime}}$ instead of c_{t^i}. Since such a deviation cannot be profitable, the previous inequality cannot hold. Finally, (5.3) is satisfied because player i's payoff with type t^i who submits f_i at the first stage of the mechanism is the same as his payoff in the contracting game if he offers the contract c_{f_i} by (5.11), (5.12), (5.13), and by the definition of s^i_j. ■

The next example provides an environment and an allocation that can be implemented with contractible contracts, then provides an allocation that is implementable under the usual revelation principle, but not with contractible contracts.

Example. Suppose that there are two players 1 and 2. Player 1 has four possible types, $T^1 = \left\{t_1^a, t_2^a, t_1^b, t_2^b\right\}$. The probability of each type is one fourth. Player 2's type space is degenerate. Actions of player 1 are $\{A_1, A_2\}$. Actions of player 2 are $\left\{a_1, a_2, b_1, b_1, g_{t_1^a}, g_{t_2^a}, g_{t_1^b}, g_{t_2^b}\right\}$. Payoffs are defined as follows.

$$\begin{aligned} u_j\left(t_i^y, A_k, x\right) &= u_j\left(t_i^y, A_k, x\right) = -1, \text{ if } k \neq i,\ x \neq g_t, x \notin \{y_1, y_2\} \ \ (i,j=1,2), \\ u_j\left(t_i^a, A_i, a_i\right) &= u_j\left(t_i^b, A_i, b_i\right) = 10,\ \ (i,j=1,2), \\ u_j\left(t_i^a, A_i, a_l\right) &= u_j\left(t_i^b, A_i, b_l\right) = 9 \text{ if } l \neq i,\ \ (i,j=1,2), \\ u_1\left(t, A, g_t\right) &= 0,\ u_2\left(t, A, g_t\right) = 15,\ \ i = 1,2,\ \ t \in T^1, \\ u_j\left(t, A, g_{t'}\right) &= 0 \text{ if } t' \neq t,\ \ (i,j=1,2). \end{aligned}$$

The idea of this game is the following. Player one wants match the index of his type with his action. That is, he wants to take A_i if his type is t_i^a or t_i^b $(i = 1, 2)$. The payoffs of the players are high, ten, if player 2 can match player 1's type with his action perfectly. (That is, if he takes action $a_i\left(b_i\right)$ if $t = t_i^a\left(t_i^b\right)$.) Their payoffs are nine if player 2 matches the type of player one imperfectly, that is, he takes action $a_l\left(b_l\right)$ if the type is $t_k^a\left(t_k^b\right)$ $(l \neq k)$. The twist of the game is the following. If player 2 knows the type of player 1, he can guess it, that is, he can take an action from $\left\{g_t : t \in T^1\right\}$. If he guesses right, he gets a payoff of 15 and player 1 gets zero. If he guesses wrong both players receive a payoff of zero.

Allocation 1: Consider the $\alpha(t_i^a) = (A_i, a_1)$ and $\alpha(t_i^b) = (A_i, b_1)$ $(i = 1, 2)$. This allocation can be implemented by a mechanism with public messages and therefore with contractible contracts. To show this, we simply construct the partition of T^1, τ_1,the restrictions, $\{r^i, r_j^i\}$, and the strategies, $\{s^i, s_j^i\}$ that are used to satisfy (5.1), (5.2), and (5.3). Observe first, that the interim payoff for player 2 in this mechanism is 9.5 since he has probability 1/2 of matching his action with 1's type. Player 1 of type t_1^a or t_2^a has interim payoff 10, while the other two types have interim payoff 9.

Set the information partition to $\tau_1(t_1^a) = \tau_1(t_2^a) = \{t_1^a, t_2^a\}$ and $\tau_1(t_1^b) = \tau_1(t_2^b) = \{t_1^b, t_2^b\}$. Obviously $\alpha_2(t)$ is measurable with respect to τ_1. Define the restrictions as follows:

$$\begin{aligned} r^1(t) &= \{A_1, A_2\} \text{ for all } t,\ r^2(t_1^x) = r^2(t_2^x) = \{x_1\} \text{ for } x = a, b, \\ r_2^1(t) &= A_1 \text{ for all } t,\ r_1^2 = g_{t_1^a} \text{ for all } t. \end{aligned}$$

These restrictions uniquely determin the strategies of player 2, and player 1's strategy if player 2 does not participate. These restrictions obviously satisfy (5.1), (5.2), and (5.3).

Allocation 2: A more desirable allocation would be given by $\alpha(t_i^a) = (A_i, a_i)$ and $\alpha(t_i^b) = (A_i, b_i)$ $(i = 1, 2)$. This one provides both players 10 independent of 1's type. Since player 2's action must be measurable with respect to his information about player 1's type, player 2 must learn player 1's type to implement this allocation, by having player 1 announce his type truthfully. However, if he does this, player 2 benefits by refusing to participate and guessing player 1's type since this raises his payoff to 15 for all values of player 1's type. It is possible to implement this allocation with private messages, since the mechanism designer, after privately learning 1's type, simply instructs player 2 what to play when he participates, and provides 2 with no information at all if he refuses to participate. Of course by the theorem above, this allocation cannot be implemented with contractible contracts.

References

[1] Kyle Bagwell and Robert W. Staiger. Reciprocity, non-discrimination and preferential agreements in the multilateral trading system. *European Journal of Political Economy*, 17(2):281–325, June 2001.

[2] L. Epstein and M. Peters. A revelation principle for competing mechanisms. *Journal of Economic Theory*, 88(1):119–161, September 1999.

[3] C Fershtman and K.L. Judd. Equilibrium incentives in oligopoly. *American Economic Review*, 77:927–940, 1987.

[4] Seungjin Han. Menu theorems for bilateral contracting. *Journal of Economic Theory*, 127(1):157–178, November 2006. available at http://ideas.repec.org/a/eee/jetheo/v131y2006i1p157-178.html.

[5] Kalai, A.T., Kalai, E., Lehrer, E., and D. Samet. A Commitment Folk Theorem. manuscript, Tel-Aviv University, April, 2007.

[5] Michael Katz. Observable contracts as commitments: Interdependent contracts and moral hazard. *Journal of Economics and Management Strategy*, 15(3):685–706, September 2006.

[6] David Martimort and Lars Stole. Communications spaces, equilibria sets and the revelation principle under common agency. University of Chicago unublished manuscript, 1998.

[7] Michael Peters. Common agency and the revelation principle. *Econometrica*, 69(5):1349–1372, September 2001.

[8] S.C. Salop. *Practises that Credibly Facilitate Ologopoly Coordination*. Cambridge, MIT Press, 1986.

[9] Leo Simon and W. Zame. Discontinuous games and endogenous sharing rules. *Econometrica*, 58(4):861–872, 1990.

[10] Moshe Tennenholz. Program Equilibrium. *Games and Economic Behavior*,49(2):363–373,2004.

[11] Takuro Yamashita. A revelation principle and a folk theorem without repition in games with multiple principles and agents. manuscript, Stanford University, February 2007.

Very Preliminary Version - July 31, 2008