A FOLK THEOREM FOR COMPETING MECHANISMS

MICHAEL PETERS
UNIVERSITY OF BRITISH COLUMBIA
AND CRISTIAN TRONCOSO VALVERDE
UNIVERSIDAD DIEGO PORTALES MANUEL RODRÍGUEZ

Abstract. We provide a partial characterization of the set of outcome functions that can be supported as perfect Bayesian equilibrium in the recommendation game described in Yamashita (Econometrica 2010). We show that the set of outcome functions that can be supported is at least as large as the set supportable by a mechanism designer in the sense of Myerson (Myerson 1979). We show how to support random and correlated outcomes as equilibrium outcomes in the recommendation game.

Many outcome functions can typically be supported as equilibria in competing mechanism games. Some of these outcomes look quite 'collusive'. The reason for this is that competing mechanism games often provide players the opportunity to make what they do conditional on what other players do. This allows players to support collusive outcomes by writing contracts that commit them to react whenever an opponent deviates from a putative equilibrium outcome. A complete characterization of supportable outcomes in regular contracting games is provided in Peters (2010). He shows that an equilibrium outcome function is supportable as a perfect Bayesian equilibrium in a regular contracting game only if it is supportable in a particular reciprocal contracting game in which players contracts condition directly on other players' contracts.

In most of the literature on common agency and competing auctions, contracts cannot condition directly on other contracts. It is natural to ask whether this feature could be used to limit the large set of supportable outcomes. Yamashita (2010) suggested a contracting game in which contracts condition on one another indirectly through communication with agents. The logic of his game is straightforward. Each principal commits to a mechanism that simply asks agents what he should do. If the majority of the agents' recommendations agree, the principal commits himself to carry out the recommendation.

October 11, 2012.

To support some cooperative outcome function as an equilibrium, principals offer recommendation mechanisms, and on the equilibrium path, agents unanimously recommend that each principal carry out his part of this cooperative outcome. Should any principal deviate and try to offer something other than a recommendation mechanism, the agents unanimously recommend that the others punish the deviator. The reason the agents are willing to do this is because they expect all the other agents to do it, and believe they will be ignored if they don't do likewise.

In addition to the fact that contracts can't condition directly on one another, Yamashita's recommendation game also requires that players communicate their messages privately. The Yamashita game has the same restrictions on players' ability to write contracts as does the existing literature on common agency and competing auctions. What is perhaps surprising about our result here, is that restricting contracts so that they obey the same communication restrictions as the standard literature provides essentially no restrictions on the set of supportable outcomes.

There are a number of reasons we need to write a paper on this instead of referring to Yamashita. First of all, though he explains perfectly well how competing mechanisms can be used to support multiple outcomes, he doesn't provide an explicit theorem characterizing the things that are supportable. Characterization isn't really the point of his paper. When he describes what a characterization might look like, he describes a 'value' that imposes a lower bound on principals' payoffs from supportable outcomes. This 'value' is the lowest payoff that the principal attains from any mechanism he can offer in any continuation equilibrium against any array of mechanisms of the other players. Since the calculation of these values basically requires the calculation of all equilibrium strategy rules, the 'characterization' is really nothing more than a restatement of the definition of equilibrium. So our primary objective in this paper is to turn this argument into a representation that can be used to compare his result to the rest of the literature.

One of the difficulties that arise in doing this is that Yamashita restricts players to pure strategies and non-random mechanisms. This is sensible for expositional reasons in his paper, but here we want to illustrate formally how to handle randomization. One benefit of our approach is that it shows how principals can use recommendation mechanisms to implement correlated actions.

The second difficulty has to do with private communication. What agents 'recommend' to principals in Yamashita's game is a direct mechanism. In the course of the operation of this direct mechanism a principal communicates privately with each of this agents, which determines the principal's own action. We show how to tie these private communications together in such a way that principals can coordinate their action choices.

Finally, Yamashita limits commitment ability to a group of uninformed principals who deal with informed agents who have no commitment power at all, and who make no direct choices beyond the messages that they send to principals. We show how to extend his approach to problems with informed principals and to situations in which all participants have commitment power.

Our main result is to show that if the game has enough players, every outcome function that is implementable in the sense of Myerson (1979) is also supportable as an equilibrium in this game. For complete information games, the set of joint mixtures over actions for which each player receives at least his minmax payoff it equivalent to the set of outcomes supported as equilibrium in Yamashita's game. In one sense this extends the results in Kalai, Kalai, Lehrer, and Samet (2010) to arbitrary numbers of players. However, this extension is done without using the same kind of commitment device that they did.[1] For the more interesting case of incomplete information, we show that the set of supportable outcomes is at least as large as the set of outcome functions supportable by a centralized mechanism designer. The set of supportable outcomes can be strictly larger because Yamashita's game allows players to recommend punishments that depend on what the deviator chooses to do.[2] This makes no difference in games of complete information where the max min and min max coincide when punishments can be correlated in the way we allow here. However, for the case of incomplete information this generally makes it possible to players to implement more severe punishments than a centralized planner could. We discuss this in more detail below.

1. Fundamentals

There are $n \geq 7$ players. We sometimes write N to represent the set of players. Player i must choose an action a_i from a finite set A_i.

[1] As in Peters (2010), they use commitment devices that condition on other commitment devices.

[2] In the terminology of Peters (2010), Yamashita's game isn't *regular.*

Let $a = \{a_1, \dots, a_n\}$ be an array of actions in $A = A_1 \times \cdots \times A_n$. $A_{-i} = \prod_{j \neq i} A_j$.

Each player i has a privately observed payoff type θ_i drawn from a finite set Θ. Payoffs are given by $u_i : A \times \Theta^n \to \mathbb{R}$. Players have expected utility preferences over actions.

Let P_i, P_{-i}, and P be the set of probability distributions on A_i, A_{-i}, and A respectively. A typical element $p \in P$ is a vector with p_k equal to the probability that the k^{th} element in A occurs, where the set A is indexed in some arbitrary fashion.

Let $q : \Theta^n \to P$ be an outcome function. In what follows we slightly abuse notation by writing $u_i(q, \theta)$ instead of $\sum_{a \in A} q_a u_i(a, \theta)$. We are interested in allocation rules that are incentive compatible and individually rational. Incentive compatibility means

$$\mathbb{E}\left\{u_i\left(q\left(\theta\right), \theta\right) | \theta_i\right\} \geq \mathbb{E}\left\{u_i\left(q\left(\theta_i', \theta_{-i}\right), \theta\right) | \theta_i\right\} \tag{1.1}$$

for each $i \in N$, and $\theta_i' \in \Theta_i$. Individual rationality means that for each player i there is a punishment $p^i : \Theta_{-i} \to P_{-i}$ such that for every θ_i

$$\mathbb{E}\left\{u_i\left(q\left(\theta_i, \theta_{-i}\right), \left(\theta_i, \theta_{-i}\right)\right) | \theta_i\right\} \geq$$

$$\max_{a_i} E\left\{u_i\left(a_i, p^i\left(\theta_{-i}\right), \left(\theta_i, \theta_{-i}\right)\right) | \theta_i\right\}. \tag{1.2}$$

With complete information, an allocation is individually rational if and only if it provides each player with an expected payoff that exceeds his or her *minmax value,* defined for player i as

$$u_i^* \equiv \min_{p_{-i} \in P_{-i}} \max_{a_i \in A_i} u_i\left(a_i, p^i\right). \tag{1.3}$$

Again, with complete information the punishment

$$p_{-i}^* \in \arg \min_{p_{-i} \in P_{-i}} \max_{a_i} u_i\left(a_i, p^i\right)$$

can be used to support all implementable allocations.

Notice that when constructing a punishment, or a minmax value, punishers are allowed to correlate their punishments. This is appropriate for a mechanism designer who can enforce contracts and correlate actions among agents who have agreed to participate.

2. Recommendation Game

One of the things that makes competing mechanism games challenging is specifying exactly what message spaces and mechanisms are feasible for players. Since our objective here is to study the implications of private communication, we use a very narrow interpretation of what the set of feasible mechanisms is.

Our game takes place in two stages. Players can make commitments that are based on public messages in the first stage, and on private communication that occurs during both stages. In the first stage, every player publicly announces such a commitment, declares a second public message, then sends private messages to the other players. In the second stage all messages are private. As is almost universally assumed in mechanism design, messages are verifiable, even though they are private.

Our mechanisms are unusual in two basic ways. First, they are *not* direct mechanisms. Players don't directly convey their type information. Of course they are going to do so indirectly. Second, players cannot make arbitrary commitments based on the messages they receive. We are going to give players the ability to refuse to participate in a principal's mechanism.

Since play necessarily involves indirect mechanisms, we need to describe the message spaces.

Encryption. Players are going to encrypt their type information before they publish it. To see how, begin with a publicly known algorithm $\mathcal{E}$, a measurable set K, along with a non-informative prior probability measure on K. The elements of K are referred to as *encryption keys.* The algorithm $\mathcal{E}$ converts elements of $(\Theta \times [0,1]) \times K \times K$ into a set $\tilde{\mathcal{M}}$. The elements of $\tilde{\mathcal{M}}$ are referred to as *encrypted messages.* Finally the elements in $[0,1]$ are referred to as *correlating messages.* Their role is explained below.

We assume without further comment that the encryption algorithm works in the most desirable way. For example it has the property that for every $\tilde{m} \in \tilde{\mathcal{M}}$, every $(\theta, x) \in \Theta \times [0,1]$ and every $k \in K$, there is a $k' \in K$ such that $\mathcal{E}((\theta, x), k, k') = \tilde{m}$. In words, no information is provided about the encrypted value $\tilde{m}$ from only a single encryption key.

On the other hand, a player who has both the encryption keys should be able to decrypt, so we assume that for every $\tilde{m} \in \tilde{\mathcal{M}}$ and $(k, k') \in K \times K$, either there is a unique pair (θ, x) such that $\tilde{m} = \mathcal{E}((\theta, x), k, k)$, or there is no pair for which this equality holds.[3]

Indirect Mechanisms. The public message $\tilde{m} \in \tilde{\mathcal{M}}$ that each player announces in the first period is essentially an encrypted version of his

[3]For example, suppose one wants to encrypt the real number z. Let the keys be in $K_1 = K_2 = \mathbb{R}$, and suppose the algorithm γ is given by xyz. The number $\tilde{m}$ is the encrypted value of z. Given the keys x_0 and y_0, the decrypted value of z is $\frac{\tilde{z}}{x_0 y_0}$. If you only know x_0, then z could still be any real number.

type. To decrypt this information, players need to know the keys that were used to create $\tilde{m}$. The keys are conveyed privately. In the first period, each player sends a key in K to each of his opponents. In the second period, each player tells each player what keys were reported to him in the first period. Since each player received one key from each of his $n-1$ opponents in the first period, he conveys $n-1$ keys to each of his opponents in the second period. This means that at the end of the second period, each player has received $(n-1)$ reports, each of which contains $(n-1)$ keys, along with the $n-1$ keys that were reported to him in the first period. In other words, he has a total of $n(n-1)$ keys that he can use to try to decrypt the public messages.

Let Γ_i be the set of indirect mechanisms consisting of measurable mappings from $\tilde{\mathcal{M}}^n \times K^{n(n-1)}$ into A_i. Standard competing mechanism games like competing auctions or common agency would simply allow players to commit directly to these mechanisms. What these mechanisms are missing is some way for players to communicate their market information to one another as in Epstein and Peters (1999). The basic insight in Yamashita (2010) about how to do this is to add one additional message from each of the players who might participate in a mechanism consisting of a *recommendation* to the principal about which indirect mechanism in Γ_i the mechanism designer should use. We are now able to describe these.

Recommendation Mechanisms. Now we want to expand the message space that player i can condition on to $\Gamma_i^{n-1} \times \tilde{\mathcal{M}}^n \times K^{n(n-1)}$, so that it includes not just the public messages in $\tilde{\mathcal{M}}^n$ and key reports in $K^{n(n-1)}$, but also a set of recommendations that the others make about how player i should use these mechanisms. An alternative way to think about these messages is that they represent the mechanisms that the other players expect player i to use. This is the suggestion in Yamashita (2010).

To get the characterization we want, we are going to impose a restriction on what mechanisms are enforceable. In particular, we assume that mechanisms are only enforceable when players jointly agree to participate. Joint agreement occurs when all but possibly one of the other players sends the same recommendation.

To define this formally, let $\mathcal{U}(\gamma_i)$ be the number of distinct elements in the vector γ_{-i}. The set of feasible mechanisms is then defined to be

$$\mathcal{R}_i \equiv$$

$$\Big\{ r : \Gamma_i^{n-1} \times \tilde{\mathcal{M}}^n \times K^{n(n-1)} \to A_i ; \mathcal{U}(\gamma_{-i}) > 2 \implies$$

$$r\left(\gamma_{-i},\tilde{m},\tilde{k}\right)=r\left(\gamma_{-i},\tilde{m}',\tilde{k}'\right)\forall\left(\tilde{m},\tilde{k}\right),\left(\tilde{m}',\tilde{k}'\right)\in\tilde{\mathcal{M}}^{n}\times K^{n(n-1)}\right\}.$$

No player can commit to actions that depend on the public and private messages unless other players agree to participate by sending the 'same' recommendations. Notice that this does not require that players follow recommendations. For example, if all the others recommend that player i simply take action a_i, then i can commit to any mechanism that depends on public and private messages. No player can unilaterally veto i's ability to commit since commitments are still valid when there are two distinct recommendations. However, players can jointly refuse to participate in i's mechanism by sending random recommendations.

The point of Yamashita (2010) is to explain how these recommendations can be used to fully capture the idea that agents have market information to convey to the principal. The way they do this is to make joint recommendations based on whether or not they have seen some player deviate from some putative equilibrium.

The *Recommendation Game* is a competing mechanism game in which players simultaneously announce mechanisms in $\mathcal{R}$ along with their encrypted types. At the same time, they then send private messages revealing information about their encryption keys to the others. Then, as in any competing mechanism game, the other players send private messages which determine each players actions. We are interested in the perfect Bayesian equilibrium of the recommendation game. A perfect Bayesian equilibrium is a set of sequentially rational strategy rules and beliefs that satisfy Bayes rule when possible.[4]

3. Theorem

At this point we can state our main theorem:

Theorem 1. *If there are 7 or more players, then an allocation rule can be supported as a perfect Bayesian equilibrium in the recommendation game if it is incentive compatible and individually rational.*

It is important to point out what this theorem adds to the logic in Yamashita. Most obviously it covers random, and even correlated outcomes that could not be captured because of the pure strategy non-random mechanism assumptions in Yamashita. Secondly, it extends the characterization from the uninformed principal informed agents framework to an environment in which there are informed principals. It covers common agency provided there are seven or more principals. Common agency is ruled out by Yamashita's approach since he relies on

[4]This is sometimes called a weak perfect Bayesian equilibrium.

multiple agents to coordinate report. It also admits problems in which bargaining power is evenly distributed among players. At the most fundamental level, it provides a characterization in the form of a set of inequalities, which Yamashita's paper does not do, as we explained above.

Secondly, the characterization result supports the usual Bayesian individual rationality condition in which punishments do not have to be incentive compatible. This is true even though the solution concept used to describe equilibrium requires sequential rationality - as does Yamashita. The way this is accomplished is by making two modifications to Yamashita's game. One is to have each player announce their type before they learn whether or not they will be engaged in a punishment by having them publish encrypted type reports in the first period. The second is to allow players to jointly veto a mechanism by sending different recommendations to a player. This prevents a deviating player from committing to a mechanism that depends on the types of the non-deviators. The deviator has no better option than to choose his action as best he can given the anticipated punishment. Effectively, subgame perfection imposes no restrictions on the set of implementable punishments.

Of course, Yamashita's point is not to provide a characterization in the first place. It is simply to show how recommendation mechanisms work. Our model goes beyond this. We start with the set of incentive compatible individually rational allocation rules, then show how to implement all of them.[5]

The proof involves a number of detailed constructions, so we sketch the way it works here. The novel parts involve randomizing and correlating the actions of players who make independent commitments, explaining why players can implement type contingent punishments that aren't incentive compatible, and explaining how the very 'indrect' mechanisms that we have described are converted into something that looks more like a standard direct mechanism.

Begin with the encrypted types that players publish in the first period. We want other players to be able to use this type information when they choose their own action. As we have described, we allow players to privately send encryption keys to some of the other players in the first period. The whole point of this exercise is to have players declare their types before they know whether or not there has been a

[5]Even if we don't know what these allocation rules are, it seems a far easier problem to calculate them for some environment than it does to find all mechanisms which have pure strategy continuation equilibrium.

deviation. If players believe they will remain on the equilibrium path, they want to convey their types truthfully to all the other players simply because the outcome function that they expect to be implemented along the equilibrium path is incentive compatible. Once they realize there has been a deviation and they will be punishing, they might want to lie about their types, since we haven't required the punishment to be incentive compatible. However, if the non-deviators have followed their equilibrium strategy, they will already have published their true type and sent out the encryption keys the other players need to decode it.

Once the second period comes along, the players recognize whether or not there has been a deviation that pulls them off the equilibrium path. Then we simply extend the Yamashita idea not just to recommendations, but also to reports about encryption keys. Since the other players are expected to report keys truthfully and make a common recommendation, players might as well do the same since the mechanisms we construct explicitly ignore unilateral deviations in reports.

The key to making this work is then to have at least three players making recommendations and reports about relevant information. Since players don't want to reveal their type information to any of the other players in the first period, they send out two sets of encryption keys which can only be used together to decrypt the types. Each player must then report each of his two encryption keys to three players. This is why we have 7 players - the sender and 6 receivers.

Once we have convinced players to send out the right encryption keys in the first period, we design a mechanism that uses the decrypted values of players types and correlating reports to choose actions. We correlate the actions using a mixing device from Kalai, Kalai, Lehrer, and Samet (2010) which shows how to use uniformly distributed correlating messages to do this. To show that players are happy to choose their correlating messages uniformly, we extend the proof of this fact from Kalai, Kalai, Lehrer, and Samet (2010) to the situation where there is an arbitrary number of players.

4. Proof: Some Preliminary Ideas.

Our proof combines a number of ideas. We borrow methods from computer science to implement correlated and random outcomes. We then develop a sequential communication mechanism that effectively converts private communication into a public correlating device. We explain each of these methods before we proceed to the proof of the main theorem.

4.1. **Implementing random outcomes with non-random contracts.** The set of all profiles of actions, A, is a finite set. Suppose that it has κ elements in all. Give each profile $a \in A$ an arbitrary index l, so that the set of profiles is indexed from 1 to κ. Let π be a vector of κ probabilities that sum to one, with π_a being interpreted as the probability assigned to profile a. Let $\tilde{x}$ be a random variable uniformly distributed on $[0,1]$. The *randomizing function* $\alpha^{\pi}(\cdot, A)$ is defined by

$$\alpha^{\pi}(\tilde{x}, A) = \left\{ a_l \in A : l = \min_{l' \in \{1,\ldots,\kappa\}} \sum_{i=1}^{l'} \pi_i \geq \tilde{x} \right\}. \tag{4.1}$$

For any profile a_l, this randomizing function takes value a_l with probability π_{a_l}.

Let $\alpha_i^{\pi}\left(\tilde{t}, A\right)$ be the projection of α onto A_i - in words, α_i^{π} takes value $a_i \in A_i$ whenever the i^{th} coordinate of $\alpha^{\pi}\left(\tilde{t}, A\right)$ is equal to a_i. From the definition, it is immediate that

$$\left\{\alpha_1^{\pi}\left(\tilde{t}, A\right), \ldots, \alpha_n^{\pi}\left(\tilde{t}, A\right)\right\} = \alpha^{\pi}\left(\tilde{t}, A\right). \tag{4.2}$$

In other words, if we want players to implement a profile of actions a with probability π_a, then we can do it by showing them a common correlating device $\tilde{t}$ and having them each take action $\alpha_1^{\pi}\left(\tilde{t}, A\right)$,

This idea is taken from Kalai, Kalai, Lehrer, and Samet (2010) who explain the idea in two player games. This is why the correlating messages appear along with type in our formulation. The correlating messages are ultimately used to carry out the randomization as in (4.1). To get this randomization to work. We next have to find some way to ensure that players are happy sending uniformly distributed correlating messages. We turn now to this problem.

4.2. **A property of uniform distributions.** For any non-negative real number x, $\lfloor x \rfloor$ means the *fractional part* of x (sometimes the terminology is x mod 1). Let $\tilde{x}_1, \ldots, \tilde{x}_n$ be a collection of n independent random variables, where each $\tilde{x}_i$ is uniformly distributed on $[0,1]$. For $n \geq 2$, fix $\tilde{x}_i = \overline{x}$ for some i. Then $\lfloor \overline{x} + \sum_{j \neq i} \tilde{x}_j \rfloor$ is a random variable. This random variable turns out to be uniformly distributed on $[0,1]$ independent of $\bar{x}$. [6] Since this argument proves very useful below, we give a simple proof in the Appendix section 8.1.

[6] This appears to be conventional wisdom in statistics. The theorem is referred to in Deng and E.Olusegun (1990). A proof that the sum mod 1 of a pair of random variables on $[0,1]$ is uniform as long as at least one of the random variables is uniform is given in Deng, Lin, Wang, and Yuan (1997), Theorem 3.1 (see especially the comment after the theorem).

An immediate corollary is the following result, which will be used extensively below:

Lemma 2. *Suppose that $\{\tilde{x}_1, \dots, \tilde{x}_n\}$ are all independently and uniformly distributed on $[0,1]$ and π a vector of probabilities assigned to each element of A. Then the probability distribution*

$$\alpha^{\pi}\left(\lfloor \tilde{x}_i + \sum_{j \neq i} \tilde{x}_j \rfloor\right)$$

is independent of $\tilde{x}_i$.

The way we use this is to imagine that we want to implement a randomization over profiles of actions in A which assigns probability π_k to the profile a_k. Then we can do it by asking each player to name a number in $[0,1]$, then implementing the action $\alpha^{\pi}\left(\lfloor \tilde{x}_i + \sum_{j \neq i} \tilde{x}_j \rfloor\right)$. If every player thinks the others are choosing numbers using a uniform distribution, then no player thinks he can influence the probability distribution over outcomes by doing anything else.

4.3. **Confirmation Process.** Now we turn to the hard part of this problem, which is to ensure that when players communicate privately, they all end up with the same information. In particular, we want every player to condition on the same value for every other player's type. Also to implement randomized outcomes as above, we need to do something to ensure that every player communicates the same randomizing device to every other player.

This is accomplished in part by forcing players to commit themselves by publicly announcing messages in the first period. However, the last thing we want is for players to announce their types publicly. In an environment like that, most players would simply wait around to see the type reports then best reply against them. It would be hard for the players to sustain any kind of cooperation in that case.

This is the point of the encryption. The complicated part is to get the players to reveal their keys to all the others. Since they convey these keys privately, there is nothing to guarantee that they reveal the correct keys, or that they send the same keys to all players.

What we want to show is that in our game, there will always be a perfect Bayesian equilibrium in which each player reveals *the same* encryption key to each player in a group consisting of at least 3 of the other players. Ultimately, this will be enough for us to show that there is always and equilibrium in which each player recovers all of the encryption keys and that players convey their encryption keys truthfully.

We can illustrate the procedure we are going to use with an example. Suppose that player i has three different types, and wants to implement one of three different actions $\{a_1, a_2, a_3\}$. A player of type t_i strictly prefers action a_i to the other two no matter what his opponents do, but he does better with these actions if his opponents don't know his type. He wants to commit himself to a type contingent action (and wants the players to know he is committed) but doesn't want to reveal his type. The way he would do it is to publicly announce his encrypted type $\tilde{m}$, using a pair of keys $(k, k')$to encrypt it. He would then privately reveal k to a group $G^+$of his opponents consisting of at least 3 players. He would privately reveal the second key k' to another group G^- consisting of three completely different players. He can report any key he likes to players who aren't in either G^+ or G^-. Even if he reveals the keys truthfully, none of the players has enough information to decrypt $\tilde{m}$ from the single key report they have heard

He should then commit himself to a mechanism that chooses an action based reports the other players provide him in the second period about the encryption key he reported to them in the first period. The commitment should work the following way: If at least two of the three players in group G^+ report the same key $\tilde{k}$ and at least two of the three players in G^- report the same key $\tilde{k}'$, then he should commit himself to the action $a_{\tilde{t}}$ where $\tilde{t} = \gamma^{-1}\left(\tilde{m}, \tilde{k}, \tilde{k}'\right)$ if $\gamma^{-1}\left(\tilde{m}, \tilde{k}, \tilde{k}'\right) \in \{t_1, t_2, t_3\}$ and to action a_1, say, otherwise (recall that γ is the publicly know encryption algorithm). It should be clear from this example that there is an equilibrium where each of his opponents reports the key they received accurately in the second period because they believe that player i reported the same key to each of the other two players, and because they believe the other two players are both going to report truthfully. The same argument goes for the second group.

Since the two keys together identify the type and lead to the right commitment, player i has no incentive to lie about his type or to transmit different keys to different players. This is the method we are going to use to ensure players transmit their encryption keys accurately - they are going to commit themselves to mechanisms that give them the right incentives to do so.

Recall that in the first period, players privately send keys to their opponents. For each player i, let G_i^+ and G_i^- be arbitrary disjoint subsets of i's opponents consisting of exactly 3 players each (which explains why we need 7 players to make this argument work). The members of these two sets will hear real messages from player i, the other players will hear a message chosen completely randomly. These sets are part of the description of an equilibrium, but for the moment assume they are common knowledge among the players.

The point of the whole exercise is to decrypt the messages the players announced in the first period in order to discover their type and correlating messages. Since each player receives a total of $n(n-1)$ messages that are supposed to convey information about the encryption keys, he faces the problem of distilling this information down to exactly $2n$ (or n pairs) of keys that he can use to decrypt public messages. Call this process τ and observe that it is a mapping from $K^{n(n-1)}$ into K^{2n}.

Now adopt the following notation: let $\tilde{k}_j^1$ represent the private message that player i receives from player j in the first round. Let $\tilde{k}_{jm}^2$ represent the second round report that player j makes to i about the key he received from player m in the first round. Observe that the message $\tilde{k}_{ji}^2$ would be j's report about what i told him in the first round. This message is important in what follows despite the fact that i already knows what report he made to j in the first round. To see why, just consider the simple example given above where the mechanism designer has three actions and three types. By committing himself to act in a sensible way only when his agents report the same information, he effectively commits himself to send each of them the same information. Since the principal acts when agents messages confirm each other, we refer to this as a confirmation process.

Let $\tilde{k}_{-i}^1$ represent the vector of reports that i receives from the other players in the first period, and $\tilde{k}_{-i}^2$ the vector of reports that i receives in the second period. A confirmation process converts the vectors $\left(\tilde{k}_{-i}^1, \tilde{k}_{-i}^2\right)$ into exactly $2n$ encryption keys. For a pair of vectors $\tilde{k}_{-i}^1$ and $\tilde{k}_{-i}^2$ of first and second period reports to i, and a subset G_j^+ of 3 players other than j, define

$$MAJ_j\left(\left(\tilde{k}_{-i}^1, \tilde{k}_{-i}^2\right), G_j^+\right) =$$

$$(4.3)\quad \begin{cases} k & \tilde{k}_{mj}^2 = k \forall m \in G_j^+ \vee \exists! m \in G_j^+ ; \tilde{k}_{mj}^2 \neq k \wedge i \notin G_j^+ \\ k & \exists m \in G_j^+ : \tilde{k}_{mj}^2 = \tilde{k}_j^1 \vee \tilde{k}_{mj}^2 = k \forall m \in G_j^+ ; m \neq i \wedge i \in G_j^+ \\ \overline{k} & \text{otherwise.} \end{cases}$$

The notation $\exists!$ in this expression means "there exists a unique...", $\vee$ means logical 'or', while $\wedge$ means logical 'and'.

We can define $MAJ_j\left(\left(\tilde{k}_{-i}^1, \tilde{k}_{-i}^2\right), G_j^-\right)$ in the same way. The function is complicated enough, it is probably just as easy to define it in words. The set G_j^+ consists of three players that learn the first of j's encryption keys. The function MAJ is supposed to record what these three players reported to player i. How i interprets the messages depends on whether he is one of the members of G_j^+ or not. If G_i^+ contains only players other than i, then MAJ looks at their three reports about j's key, and replies with that report if all three players reports are the same, with the majority report if only two out of the three reports agree, and with an arbitrary report $\overline{k}$ otherwise.

In the event that i is in the set G_j^+, then MAJ looks at the report that i received from j in the first period, and the reports that the other two players in G_j^+made to him in the second period. MAJ responds with the majority report, or with $\overline{k}$ if there isn't a common message among the three. The notation MAJ is obviously meant to signal 'majority report'.

Definition 3. The mapping τ for player i is referred to as a *confirmation process* if there is a collection of sets $\left\{G_j^+, G_j^-\right\}_{j=1,\dots,n}$ such that

(4.4)

$$\tau_j\left(\tilde{k}_{-i}^1, \tilde{k}_{-i}^2\right) = \left(MAJ_j\left(\left(\tilde{k}_{-i}^1, \tilde{k}_{-i}^2\right), G_j^+\right), MAJ_j\left(\left(\tilde{k}_{-i}^1, \tilde{k}_{-i}^2\right), G_j^-\right)\right).$$

A confirmation process describes the way that a particular player i processes information from other players. The first thing he does is to choose the groups G_j^+ and G_j^- for himself and each of his opponents. He then responds only to messages he receives from those groups, and only if the messages he receives from within a group agree with one another. If they do agree, he ends up with a pair of encryption keys for each player. Once he has these pairs of encryption keys, he can use them to decrypt the public messages $\tilde{m}$ that were made in the first period. This leads very naturally to a *confirmation mechanism* which makes commitments based on the decrypted type and correlating reports that emerge from this process.

Let $\left(\theta^\tau\left(\tilde{m}, \tilde{k}_{-i}^1, \tilde{k}_{-i}^2\right), x^\tau\left(\tilde{m}, \tilde{k}_{-i}^1, \tilde{k}_{-i}^2\right)\right)$ be the vector of decrypted types and type reports to be used by player i when using the confirmation process τ. The j^{th} component of this vector is given by

$$\left(\theta_j^\tau\left(\tilde{m}, \tilde{k}_{-i}^1, \tilde{k}_{-i}^2\right), x_j^\tau\left(\tilde{m}, \tilde{k}_{-i}^1, \tilde{k}_{-i}^2\right)\right) \equiv$$

(4.5)

$$\begin{cases} \mathcal{E}^{-1}\left(\tilde{m}_j, \tau_j\left(\tilde{k}_{-i}^1, \tilde{k}_{-i}^2\right)\right) & \text{if } \mathcal{E}^{-1}\left(\tilde{m}_j, \tau_j\left(\tilde{k}_{-i}^1, \tilde{k}_{-i}^2\right)\right) \in \Theta \times [0,1] \\ \left(\underline{\theta}_j, \underline{x}_j\right) & \text{otherwise.} \end{cases}$$

Definition 4. A mechanism $\gamma_i : \mathcal{M}^n \times K^{n(n-1)} \to A_i$ for player i is a *confirmation mechanism* if there is a mapping $\gamma_i^0 : \Theta^n \times [0,1]^n \to A_i$

and a confirmation process τ such that

$$\gamma_i\left(\tilde{m},\tilde{k}^1_{-i},\tilde{k}^2_{-i}\right)=\gamma_i^0\left(\theta^\tau\left(\tilde{m},\tilde{k}^1_{-i},\tilde{k}^2_{-i}\right),x^\tau\left(\tilde{m},\tilde{k}^1_{-i},\tilde{k}^2_{-i}\right)\right). \tag{4.6}$$

A strategy for this game specifies an action in information sets in which player i has to send out encryption keys in the first round, and information sets in which player i has to report the keys he received from others in the first round.

Definition 5. A strategy rule for player i is said to be *confirming* with respect to the confirmation process τ if i sends the same encryption key to each member of G_i^+, sends the same encryption key to each member of G_i^-, and truthfully reports the keys that were reported to him.

Lemma 6. *Let τ be a confirmation process for player i, and suppose that all players are using strategies that are confirming with respect to τ. Then for any $j\neq i$, $\tau\left(\tilde{k}^1_{-i},\tilde{k}^2_{-i}\right)$ is independent of what j reports. On the other hand, if (k,k') is any pair of encryption keys, there are messages that i can send such that*

$$\tau_i\left(\tilde{k}^1_{-i},\tilde{k}^2_{-i}\right)=(k,k').$$

The logic of this Lemma is straightforward - any report that one player sends to another will be ignored unless it is confirmed by at least one other player. If the others are all using confirming reporting strategies, then player j anticipates that the message he received from player l in the first round was was the same message that l conveyed to each of the other players. If he reports it truthfully to i in the second round, then one of two things will happen: if j is in G_l^+ or G_l^-, then two other players will report the same message to i. If j reports the message truthfully, he will simply confirm those other messages. If he reports something else, his report will be discarded. If j is not in G_l^+ or G_l^-, then his message would be ignored anyway.

For i, if he reports the same message to each of the players in G_i^+ and each of the players in G_i^-, then those players are expected to mimic that message back to him if they are using confirming reporting strategies. Since they do this no matter what message i sends them, the result follows.

There is one last preliminary idea needed regarding confirmation processes. A confirmation process for player i includes a list of groups G_j^+ and G_j^- for each of the other players. If player j also uses a confirmation process, then j could, in principle, specify a distinct set of groups. We will say that the confirmation process used by a pair of players is *consistent* if each process uses the same groups G_j^+ and G_j^- for each player.

4.4. **Consensus Mechanisms.** Now we want to extend the confirmation idea described above once more. The idea that principals should ask their agents for recommendations about how to process information is due to Yamashita (2010). His idea was to have the principal commit himself to carry out the recommendations of the agents provided an outright majority of the agents make the same recommendation. We simply adapt this idea here. In our context there may or may not be agents, so players ask other players for recommendations. If the principal's mechanism commits him to carry out the recommendation when all the other players, or all but one of the other players agree, then we say that the principal's mechanisms is a *consensus mechanism.*[7]

Formally, a mechanism $r_i : (\Gamma_i)^{(n-1)} \times \tilde{\mathcal{M}}^n \times K^{n(n-1)} \to A_i$ is a consensus mechanism if

$$r_i\left(\gamma_{-i}, \tilde{m}, \left(\tilde{k}^1_{-i}, \tilde{k}^2_{-i}\right)\right) =$$

$$\begin{cases} \gamma'\left(\tilde{m}, \left(\tilde{k}^{\tilde{1}}_{-i}, \tilde{k}^2_{-i}\right)\right) & \text{if } \{\exists! j : \gamma_j \neq \gamma_k \equiv \gamma' \forall k \neq j\} \vee \{\gamma_k = \gamma_j \equiv \gamma' \forall j, k\} \\ \overline{a}_i & \text{otherwise.} \end{cases}$$

Each of the other players recommends a mechanism γ that converts a public profile of encrypted types and correlating messages, along with all the private communication, into an action. The consensus mechanism implements the mechanism γ if all the recommendations, or all but one of the recommendations, agree. Otherwise, the consensus mechanism implements an arbitrary outcome.

Now the proof of our folk theorem can be done constructively. On our equilibrium path all players will offer a consensus mechanism independent of their type. If all players do this, then each of them will recommend a confirmation mechanism to each of the other players. The details of the confirmation mechanism will depend on the allocation rule we are trying to support. If some player deviates and offers something other than a consensus mechanism, then the other players will recommend to each other a confirmation process than penalizes the deviator.

[7]This is the generalization of a *menu mechanism* in common agency. The principal offers the agent a menu of indirect mechanisms and commits himself implement whatever they choose. In a multiple agency context, the additional restriction is simply that agents have to agree about which mechanism they want.

It should be apparent why this construction will work. Players can see whether or not everyone has offered a confirmation process after mechanisms are announced. They then believe that they know what recommendations the others will make. The nature of a consensus process is such that unilateral disagreement is ignored, so going along with the majority is a weak best reply.

Once players decide to participate and recommend the confirmation process they think that everyone else is going to recommend, they need to send out encryption keys and report the keys that have been reported to them. If they believe that the others are using confirming strategies, then they can't improve on a strategy that sends the same encryption keys to each of the players in the confirming groups G_i^+ and G_i^- and reports truthfully any keys they learned in the first period.

5. The Proof of the Main Theorem

Proof. On the equilibrium path each player, no matter what his type, should offer a consensus mechanism, and choose a pair of keys (k, k') by independently drawing each key from K using some non-informative prior distribution on K.[8] The players should report keys keys 'truthfully' in both periods. To describe exactly what truthful reporting means, and to describe the recommendations players are supposed to make, we need to describe the confirmation mechanisms that players are going to recommend. These depend on the outcome function we are trying to support.

Let $q(\theta)$ be the randomization that is to be supported when types are θ. Since the allocation rule is individually rational, there is a collection of punishments that ensure participation by each player when a mechanism designer tries to implement q. Let $\{p_i(\theta_{-i})\}_{i \in N}$ be the type contingent randomization that is to be carried out by the players other than i when i is being punished.

Let τ be a confirmation process for player i, which specifies the confirmation sets G_i^+ and G_i^-. Recall that the process τ produces n pairs of keys that can be used to decrypt the public message $\tilde{m}$, producing a vector in $T^n \times [0,1]^n$ of types and correlating messages as in (4.5). Write (θ^τ, x^τ) for short to represent these decrypted bits of information (implicitly bearing in mind that they are derived from reports $\left(\tilde{m}, \tilde{k}_{-i}^1, \tilde{k}_{-i}^2\right)$. The equilibrium path recommendation by the

[8]For example, for each measurable subset B of K the probability with which the key is drawn from B is equal to the measure of B divided by the measure of K.

other players to player i is given by

$$\gamma_i\left(\tilde{m}, \tilde{k}^1_{-i}, \tilde{k}^2_{-i}\right) = \alpha_i^{q(\theta^\tau)}\left(\lfloor \sum_{j\in N} x_j^\tau \rfloor, A\right) \tag{5.1}$$

where α_i^q is the projection of the randomizing function for mixture $q\left(\tau^\theta\left(s_{-i}, t_{-i}\right)\right)$ on the set A onto the set A_i. The randomizing function is defined by (4.1) above.

When player l unilaterally deviates in the first period and offers something other than a recommendation mechanism, the others (including the deviator) will recommend

$$\gamma_i^l\left(\tilde{m}, \tilde{k}^1_{-i}, \tilde{k}^2_{-i}\right) = \alpha_i^{p_i(\theta^\tau)}\left(\lfloor \sum_{j\in N} x_j^\tau \rfloor, A_{-l}\right) \tag{5.2}$$

to each non-deviating player i. Note that the confirmation process used by each player in (5.2) is the same as the confirmation process they use in (5.1).

We can now specify the strategies for the players.

STRATEGIES: In period 1, each player should offer a consensus mechanism, choose a number x in $[0,1]$ using a uniform distribution, select a pair of keys (k, k') by independently applying a non-informative prior distribution to K, then publicly announce the message $\tilde{m} = \mathcal{E}\left(\left(\theta_i, x\right), k, k'\right)$. He should truthfully report the keys to his confirmation groups G_i^+ and G_i^-. He should choose reports to players outside his confirmation groups using a non-informative prior on K.

Since all players offer the same mechanism in the first period, no matter what their types, all on path information sets have each player offering a consensus mechanism. In those information sets, player i should recommend γ_j as given by (5.1) to each other player j, then truthfully report the keys they received in the first period to each of the other players.

In any information set in which a single player, say player k, has deviated and offered some mechanism other than a consensus mechanism, player i (including the deviator himself) should recommend the punishment mechanism γ_j^k as given by (5.2) to each player $j \neq k$, truthfully report the keys they received from the other players to each player other than k, and choose key reports and recommendations for player k that are independently drawn using non-informative prior distributions.

In any information set in which more than three distinct mechanisms are offered, each player should independently randomize over all messages using a non-informative prior distribution.

BELIEFS: In every information set off the equilibrium path, players should maintain their prior beliefs about types. In the information set in which a single player has deviated and offered something other than a consensus mechanism, the non-deviators should believe that the deviator truthfully reports the keys that other players reported to him.

Now we proceed to prove that the strategies and beliefs specified above constitute a perfect Bayesian equilibrium.

ON-PATH: To begin, note that we are trying to show that there is an equilibrium that supports the outcome function q as a perfect Bayesian equilibrium. In this equilibrium players expect the others to recommend a confirmation mechanism as given by (5.1) to each of the other players. If player i does this as well, then he expects each player will use the mechanism described by (5.1) to convert key reports and public messages into actions. Since he is expecting players to encrypt their true types, choose their correlating message using a uniform distribution, randomly select keys independently using a non-informative prior, and report all keys truthfully, he expects by (5.1) and (4.2) that the randomization $q(\theta)$ will occur when players true types are θ. So his payoff by following the equilibrium strategy is

$$\mathbb{E}\left\{u_i\left(q\left(\theta_i,\theta_{-i}\right),\left(\theta_i,\theta_{-i}\right)\right)|\theta_i\right\}.$$

There are a variety of deviations that are possible. Notice first of all, that whatever he does in the first period, he is reporting what he heard from other players in the second period. He expects the others to report exactly the same messages that he heard. Then by Lemma 6, he believes that his own second period messages will be ignored. So reporting truthfully is a best reply.

If he sends out different keys to different players in his confirmation group, he expects them to truthfully report these different keys to the others. His keys will be treated as bad keys, and by (4.5) his type will be interpreted as $(\underline{\theta}_i, \underline{x}_i)$. By Lemma 2, his payoff is then

$$\mathbb{E}\left\{u_i\left(q\left(\underline{\theta}_i,\theta_{-i}\right),\left(\theta_i,\theta_{-i}\right)\right)|\theta_i\right\}$$

which is lower than his equilibrium payoff because of the fact that $q(\cdot)$ is incentive compatible.

If he sends players false keys, or encrypts the wrong type and correlating message, then again, by Lemma 2 (which shows that the distribution of the fractional part of the sum of correlating messages is independent of i's message when the others messages are chosen uniformly), the best he can do is given by the payoff

$$\mathbb{E}\left\{u_i\left(q\left(\theta_i',\theta_{-i}\right),\left(\theta_i,\theta_{-i}\right)\right)|\theta_i\right\}$$

for some θ_i'. Any such deviation is unprofitable since q is incentive compatible.

OFF-PATH: This brings us to the deviations that take the game off path in the second period. If there are three or more mechanisms offered in the first period, then reporting randomly is a best reply to the expectation that everyone else will report randomly. In that case, contracts cannot be enforced, actions are already committed, and there is nothing else a player expects to be able to do.

The only unilateral deviation occurs when a single player l offers something other than a consensus mechanism. Then player l expects the others to make random recommendations to him, making it impossible for him to commit. Each non-deviating player j is expected to recommend (5.2) to any other non-deviating player i. Since these recommendations commit i to $\gamma_i^l\left(\tilde{m}, \tilde{k}_{-i}^1, \tilde{k}_{-i}^2\right)$ as given by (5.2). Then by (4.2), the deviator's payoff is

$$E\left\{u_l\left(a_l, p^i\left(\theta_{-l}\right), \left(\theta_l, \theta_{-i}\right)\right)|\theta_l\right\}$$

where a_l is the action specified by his contract for the case in which player l does not receive a common recommendation. This is lower than his equilibrium payoff because of the individual rationality condition (1.2).

The deviator cannot improve this payoff by altering his messages for two reasons. First, the punishment the others impose is independent of his type, and by Lemma 2, his correlating message. So it doesn't matter what keys he sends in the first period, or what his encrypted message is. In the second period, the others are expected to report the keys they heard truthfully. By Lemma (6), his second period reports are ignored.

Finally, when players other than the deviator make their reports in the second period, they expect all the others, including the deviator, to recommend the punishment mechanism and to reveal the keys they received in the first period truthfully. Their deviations are ignored as a consequence, so that truthful reporting is sequentially rational. □

6. Remarks.

The approach above shares many of the methods of the literature on communication in games, as in Gerardi (2004),Forges (1986), or Barany (1992). Gerardi (2004), for example, uses the majority rule approach to ensure that players all send the 'correct' message in his communication protocols. This is exactly the idea behind a consensus mechanism. He also uses the randomization idea in (4.1), albeit restricted to two players.[9] The important difference between our paper and all this literature is the fact that we are doing mechanism design - players can make commitments based on messages. So the allocation rules we support aren't typically communication equilibrium (or correlated equilibrium with complete information).

As an example, consider a prisoner's dilemma played between two players 1 and 2. To make the environment fit our settings, add two disinterested players 3 and 4 who take no actions of their own. The actions are C for cooperate and D for defect. The only communication equilibrium in this game has both players A and B playing D, since the action C is strictly dominated. The outcome where both 1 and 2 play C can be supported as an equilibrium with recommendation mechanisms. A recommendation mechanism commits the player to the action the other three players recommend provided 2 of the three recommendations agree. To keep things simple, suppose the only other mechanisms that players are allowed to offer are the ones that ignore all messages and commit to either C or D. The strategies are for each player 1 and 2 to offer a recommendation mechanism then recommend C if the other player offers a recommendation mechanism, Players 3 and 4 recommend C if 1 and 2 both offer recommendation mechanisms, and recommend D if one player offers a recommendation mechanism and the other doesn't. It should be apparent in this construction that deviating from this equilibrium changes the action of the other player from C to D. So these strategies constitute an equilibrium.

What is important in this exercise is that players 1 and 2 have a way to *commit* themselves to an action which can never be part of a communication equilibrium.

There is a literature on mechanism design in communication networks (J. Renault and Tomala (2010) or Renou and Tomala (2012)) which considers sequential communication schemes like the one we described in Section 4.3. In this literature, a centralized mechanism designer can communicate with only a subset of all the agents. However, the agents can communicate among themselves according to some exogenously fixed communication protocol. The papers cited above provide communication protocols which allow agents to communicate their type information secretly to the principal. The essence of their result

[9] He has two players publicly announce numbers in the interval $[0, 1]$ then uses the fractional part as a public correlating device. As there are only two players and their messages are public, he doesn't need our Remark 7, which shows that the property for the two player case also holds with more players and private communication. In our model, there are no public messages at all, beyond the mechanisms that players announce at the beginning of the game.

is to show that, provided the communications network is right, there is a way for agents to encode their own information along with the information they have received from others, and pass it along in such a way that only the the mechanism designer can decode it.

In order to ensure that players pass along encoded information truthfully, their papers use a method that resembles our confirmation process. A protocol that transmits player 1's type (assumed here to be a positive number) to the mechanism designer is repeated, say, 3 times. Player 1 chooses at random one of the three repetitions and transmits his type on that repetition. On each of the other two repetitions he transmits the number 0 as his type. If the mechanism designer decodes 2 zeros and one positive number, he responds as if the type is a positive number. If he decodes any other sequence, he implements a punishment. The purpose of this is to ensure that the other players transmit messages from player 1 truthfully. They don't know which of the three messages from player 1 contain his type report. So they have a 1 in three chance of changing the outcome in a way that they might like, and a 2/3 chance of inducing the punishment when they lie. Assuming that there is a punishment that is strictly worse than any outcome the mechanism designer might otherwise implement, no matter the type of any other player, then repeating the protocol enough times will ensure that players other than player 1 transmit messages truthfully.

The details of the argument differ, but the spirit is the same as the confirmation process - if other players are hearing the same message that you are, then there are sometimes ways to to check whether they are transmitting the information truthfully. Our communication process is structured to do this, so we don't need a 'worst outcome' that a mechanism designer can use to enforce truth-telling. In our framework, deviating messages are simply ignored. Of course, the context of our result is quite different since we don't have a mechanism designer in the first place - we deal with decentralized competition.

Nonetheless, the method they describe illustrates how the results presented here might be extended to games with fewer than seven players. Our communication mechanism requires agreement among all but one of the players who are participating in a mechanism. If there are only two players, and the messages they send are different, then the player who is interpreting them does not know which message is the correct one, and which is a deviation. Each of our players has to have at least three others sending him messages for our method to work. The method above illustrates how a player might detect deviations with messages from only two players provided the sequential communication mechanism goes on for long enough. It may also be possible to extend our results if there is a public correlating device using methods like those in Forges and Vida (2011) who show that communications equilibrium outcomes can be implemented with long cheap talk in games with only two players using a public device.

The use of a second round of communication to provide a mechanism designer with additional information is similar to the argument in Mezzetti (2004), who shows how a mechanism designer can improve outcomes by using a second round of information in which players provide information about their values. When players' payoffs are interdependent, each player's value contains information about everyone else's type in much the same way the first round reports do here. Of course, the method we use to get players to reveal this information is quite different than it is in that reference.

Folk theorem like results for competing mechanism games have been provided by Tennenholtz (2004), Kalai, Kalai, Lehrer, and Samet (2010) and Peters and Szentes (2012). The essential difference between these papers and our result here is that they assume contracts condition directly on the contracts of other players. The paper by Peters and Szentes (2012) deals with incomplete information games. It fully characterizes the outcome functions that can be supported as contract equilibrium. However, it assumes that players never communicate privately. Any type information that a player wants to convey must be publicly conveyed through his contract offer. This can limit the effectiveness of punishments since a deviating player will inevitably know the types of the other players when he deviates. It is difficult to give a formal description of the difference between the two papers because Peters and Szentes (2012) rule out randomization. To illustrate the relationship between the outcome function and the information that a deviator would then have during the punishment phase, we would need to develop considerable additional formalism. Roughly speaking, their characterization provides an individual rationality constraint that looks like 1.2 except for the fact that the deviator's beliefs when he chooses his best action would depend on the types of the punishing players. They provide an example of an outcome function that is supportable in the sense described here, which cannot be supported as an equilibrium in their game because of the fact that firms equilibrium contract offers leak information about their types. So the set of outcome functions supportable as Bayesian equilibrium in the Peters and Szentes (2012) model is strictly smaller than the set supported here.

The paper by Peters (2010) provides a characterization of outcome functions supportable as perfect Bayesian equilibrium in regular contracting games. It revisits the question in Epstein and Peters (1999) and provides a set of indirect mechanisms that can be used to mimic equilibrium outcomes in any competing mechanism game - effectively providing a revelation principle for competing mechanisms. It uses the encryption idea that is used here in order to support outcome functions that are Bayesian incentive compatible and individually rational as subgame perfect equilibrium. There are two differences. First, it assumes that contracts can condition directly on one another, somewhat in the manner of Peters and Szentes (2012). Secondly, it assumes that the process by which type information is revealed or not revealed at the second stage is carried out completely automatically. The contribution of this paper is to illustrate that the outcome functions that are supported in Peters (2010) can also be supported when players use more traditional mechanisms in which outcomes are conditioned entirely on messages privately sent by agents.

However, the competing mechanism game described in this paper is not equivalent to the game described in Peters (2010). In particular, the game described here, like the game described by Yamashita, is not 'regular' in the sense of Peters (2010). What that means in particular is that the recommendation game described here (as well as the game described by Yamashita) have players making recommendations after they see deviations. That means that they can tailor the recommendation to what the deviator actually chooses to do. The deviator will be max-mined instead of being min-maxed, generally a more severe punishment. So there will be outcome functions here that are supported as equilibria, which cannot be supported by the game in Peters (2010), where the punishment players can impose on a deviator must be independent of the deviation. A full characterization of the outcome functions supported as subgame perfect equilibrium for non-regular games (like Yamashita's) is still unknown.

7. Refinements in Competing Mechanism Games

The game here differs from Yamashita's game in an important way. In particular, in our game, commitments aren't enforceable unless players jointly agree to participate in them by sending consistent recommendations. The purpose behind this unusual modeling device is to prevent players from offering mechanisms which don't support any kind of continuation equilibrium. The reason such mechanisms present a problem is that subgame perfection requires Bayesian continuation equilibrium in all continuation games, including games in which some player deviates to a problematic mechanism.

To understand the problem, consider the following game with complete information. There are four players in this game instead of seven because with complete information, there is no need to exchange encryption keys. We only need enough players to support the recommendations.

Suppose that player 1 has three possible actions, $\{a, b, c\}$. None of the other players controls any actions at all. Player 1 offers a mechanism, and the solution concept requires that after seeing the mechanism, continuation play constitutes a Nash equilibrium (subgame perfection). Obviously, player 1 simply chooses his favorite action in any Bayesian equilibrium. However, player 1 could deviate and offer a mechanism which invites players 2 and 3 to send a message in $[0, 1]$. He commits to translate the messages m_2 and m_3 into actions the following way:

$$\gamma(m_2, m_3) = \begin{cases} a & \text{if } m_2 < m_3 < m_2 + \frac{1}{2}, \\ b & \text{if } m_2 = m_3 \text{ or } m_3 = m_2 + \frac{1}{2}, \\ c & \text{otherwise.} \end{cases}$$

Now imagine payoffs for player 2 are$u(a) = -1$, $u(b) = 0$, and $u(c) = 1$. Player 3's payoff is $-u$. This is simply the Sion Wolfe Sion and Wolfe (1957) example of a game that has no equilibrium in either pure or mixed strategies. This is a feasible mechanism in our framework, and a reasonable looking mechanism in any framework. So in this simple setting, there can be no subgame perfect equilibrium even though the game is trivial.

One approach to this problem is to restrict the set of mechanisms that players are allowed to recommend to principals (by requiring that mechanisms only use finite message spaces for example so that continuation equilibrium always exists). An alternative approach would be to use a refinement other than subgame perfection.[10] For example, Peters and Szentes (2012) suggest a refinement that looks more like sequential rationalizability. Fortnow (2009) adds a computation cost to mechanisms so that a mechanism that doesn't support an equilibrium outcome becomes infinitely costly to offer.

Here all we do is to allow agents to jointly veto a mechanism that doesn't have an equilibrium by sending different recommendations. Vetoing a mechanism like that is always sequentially rational if you believe all the others will veto it anyway.

[10]Sequential equilibrium is not well suited to the game discussed here because the messages spaces aren't finite.

Conclusion

The basic contribution of this paper can be understood in one of two ways. First, it shows how to modify the recommendation game described by Yamashita in order to show that the recommendation game supports all the outcomes supportable by a centralized mechanism designer. Alternatively, it shows how to support the set of outcome functions described in Peters (2010) with traditional mechanisms in which commitments are based on private communication with agents.

8. Appendix

8.1. **Uniform Distributions and independence.**

Remark 7. $\lfloor \bar{x} + \sum_{j \neq i} \tilde{x}_j \rfloor$ is uniformly distributed on $[0,1]$ independently of $\bar{x}$ provides each $\tilde{x}_j$ is uniformly distributed on $[0,1]$.

Proof. Suppose that $n = 2$. Then $\sum_{j \neq i} \tilde{x}_j = \tilde{x}_j$, and $\lfloor \bar{x} + \tilde{x}_j \rfloor$ is obviously uniform. Let both $\tilde{x}_1$ and $\tilde{x}_2$ be uniform on $[0,1]$. Then the probability density function of $\tilde{z} = \tilde{x}_1 + \tilde{x}_2$ is[11]

$$f(z) = \begin{cases} z & 0 \leq z \leq 1 \\ 2 - x & \text{otherwise.} \end{cases}$$

The probability that $\lfloor \tilde{z} \rfloor \leq w$ is then given by

$$\int_0^w z dz + \int_1^{1+w} (2 - z) dz = w.$$

So $\lfloor \tilde{x}_1 + \tilde{x}_2 \rfloor$ is uniformly distributed. So when $n = 3$, $\lfloor \bar{x} + \sum_{j \neq i} \tilde{x}_j \rfloor$ is uniformly distributed. Then the argument follows by induction. If for $n-1$ players $\lfloor \bar{x} + \sum_{k \neq j} \tilde{x}_k \rfloor$ is uniformly distributed, then for n players

$$\lfloor \bar{x} + \tilde{x}_j + \sum_{k \neq i,j} \tilde{x}_k \rfloor =$$

$$\lfloor \tilde{x}_j + \lfloor \bar{x} + \sum_{k \neq i,j} \tilde{x}_k \rfloor \rfloor$$

and uniformity follows from the result for $n = 3$. □

[11] Hall (1927).

8.2. **Proof of Lemma 6.**

Lemma 8. *Suppose $n \geq 4$. Consider any subgame and set of strategy rules such that some player j believes that player i is using a confirmation process. Suppose further that all the players other than j are using strategy rules that involve a consistent revelation strategy. Then whatever the realizations (s_{-j}, t_{-j}) of the others' reports, $\tau_k^i(s_{-i}, t_{-i})$ is independent of what j reports if $k \neq j$, while there are reports that j can send to i such that $\tau_j^i(s_{-j}, t_{-j})$ takes any value in S.*

Proof. Fix the first round reports s_{-j} of the players other than j. We write in the obvious way s_{-jk} for the sub vector consisting of reports in s_{-j} by players other than k. Suppose that j's strategy is consistent and he sends the message s' to each of the other players in the first round. Then since every other player is using a consistent strategy, the value that i uses for player j will be based on first round messages (s', s_{-ij}), second round message (s', s_{-kj}) from each player $k \neq j$ since each such player is using a consistent reporting strategy, and second round message s_{-j} from player j which doesn't depend on s'. Since the first round message from player j agrees with the second round reports of each of the other players, we conclude by (4.4) that

$$\tau_j^i\left((s', s_{-ij}), \prod_{k \neq i,j}(s', s_{-kj}), s_{-j}\right) = s'.$$

Notice that this verifies the last part of the theorem - j can induce any value for τ_j^i in S.

Player j can deviate from this consistent strategy by sending different messages to the other players on the first round. He could also send different messages to i on the second round, but τ_j^i doesn't depend on i's second round messages, so we defer discussion of this second kind of deviation. Let s_k be the message he sends to player k on the first round, and $\tilde{s}_{-j}$ the vector of $n-1$ messages he sends to i on the second round. In this case there are two possibilities. If the players $k \neq j$ all report $s'_k = s'$, or if all but one of the others reports $s' = s'_i$ then by (4.4),

$$\tau_j^i\left((s'_i, s_{-ij}), \prod_{k \neq i,j}(s'_k, s_{-kj}), \tilde{s}_{-j}\right) = s',$$

which is an outcome j could have obtained by using a consistent reporting strategy and reporting s' in the first round to everyone. Otherwise,

$$\tau_j^i\left((s'_i, s_{-ij}), \prod_{k \neq i,j}(s'_k, s_{-kj}), \tilde{s}_{-j}\right) = \underline{s},$$

which is an outcome he could also accomplish with a consistent strategy by sending the message $\underline{s}$ to each player then reporting accurately to i in the second round.

To complete the proof of the theorem, observe that since player k is using a consistent reporting strategy, he will make the same first round report s_k to each of the other players. With the possible exception of player j, each of the others will then report s_k to player i. Since at least two second round reports will agree with k's first round report, we have

$$\tau_k^i\left(\left(s_i', s_{-ij}\right), \prod_{k' \neq i,j}\left(s_{k'}', s_{-k'j}\right), \tilde{s}_{-j}\right) = s_k$$

independent of $\tilde{s}_{-j}$. □

References

Barany, I. (1992): "Fair Distribution Protocols or How the Players Replace Fortune," *Mathematics of Operations Research*, 17, 327–340.

Deng, L.-Y., and G. E.Olusegun (1990): "Generation of Uniform Variates from Several Nearly Uniformly Distributed Variables," *Communications in Statistics-Simulation and Computation*, 19(1).

Deng, L.-Y., D. K. Lin, J. Wang, and Y. Yuan (1997): "Statistical Justification of Combination Generators," *Statistica Sinica*, 7, 993–1003.

Epstein, L., and M. Peters (1999): "A Revelation Principle for Competing Mechanisms," *Journal of Economic Theory*, 88(1), 119–160.

Forges, F. (1986): "An Approach to Communication Equilibria," *Econometrica*, 54, 1375–1385.

Forges, F., and P. Vida (2011): "Implementation of Communication Equilibria by Correlated Cheap Talk: The Two Player case," Discussion paper, CESifo Working Paper No. 3360, to appear in Theoretical Economics.

Fortnow, L. (2009): "Program Equilibria and Discounted Computation Time," Working paper, University of Chicago.

Gerardi, D. (2004): "Unmediated communication in games with complete and incomplete information," *Journal of Economic Theory*, 114(1), 104–131.

Hall, P. (1927): "The Distribution of Means for Samples of Size N drawn from a Population in which the Variate takes Values between 0 and 1, All Such Values Being Equally Probable," *Biometrika*, 19(3), 24–245.

J. Renault, L. R., and T. Tomala (2010): "Secret Information Transmission," Discussion paper, HEC Paris.

Kalai, A., E. Kalai, E. Lehrer, and D. Samet (2010): "A Commitment Folk Theorem," *Games and Economic Behavior*, 69(1), 127–137.

Mezzetti, C. (2004): "Mechanism Design with Interdependent Valuations: Efficiency," *Econometrica*, 72, 1617–1626.

Myerson, R. (1979): "Incentive Compatibility and the Bargaining Problem," *Econometrica*, 47, 61–73.

Peters, M. (2010): "On the Revelation Principle and Reciprocal Mechanisms in Competing Mechanism Games," Discussion paper, University of British Columbia.

Peters, M., and B. Szentes (2012): "Definable and Contractible Contracts," *Econometrica*, 80(1), 363–411.

Renou, L., and T. Tomala (2012): "Mechanism Design and Communication Networks," *Theoretical Economics*, 7, 489–533.

Sion, M., and P. Wolfe (1957): *On a Game without a Value*Princeton:Princeton University Press.

Tennenholtz, M. (2004): "Program Equilibrium," *Games and Economic Behavior*, 49(2), 363–373.

Yamashita, T. (2010): "Mechanism Games with Multiple Principals and Three or More Agents," *Econometrica*, 78(2), 791–801.