# COMMUNICATION AND AUTOMATA*

Penélope Hernández and Amparo Urbano**

WP-AD 2001-04

Correspondence: Universitat de Valencia. Avenida de los Naranjos, s/n, Ed. Departamental Oriental, 46022 Valencia. Spain. Tel. +34 96 382 82 07 / Fax: +34 96 382 82 49 / email: Penelope.Hernandez@uv.es and Amparo.Urbano@uv.es

Editor: Instituto Valenciano de Investigaciones Económicas, S.A.
First Edition March 2001.
Depósito Legal: V-1164-2001



---

* We thank seminar participants at the Tenth International Conference on Game Theory (1999), SUNY at Stony Brook. Special thanks go to A. Neyman for very useful comments and suggestions. The authors also thank partial support by DGICYT under project PB95-1074.

** Penélope Hernández & Amparo Urbano: University of Valencia

# COMMUNICATION AND AUTOMATA

Penélope Hernández and Amparo Urbano

## ABSTRACT

The main contribution of this paper is to present a new procedure to reach cooperation through pseudorandom schemes in the finitely repeated Prisoner's Dilemma game, when strategies are implemented by automata. The equilibrium path consists of a communication process followed by a coordinated play. The choice of the set of communication messages is efficient since it is the minimum set with respect to the whole coordination procedure. This allows us to reach efficient outcomes with automata complexity lying between the ones already offered in the literature.

KEYWORDS: Pesudorandom Processes; Complexity; Automata; Cooperation.

# 1 Introduction

Cooperation in the finitely repeated Prisoner's Dilemma cannot be achieved under fully rational players. Bounded rationality assumptions are needed to obtain the efficient payoff at the repeated game equilibrium. Specifically, Neyman (1985, 1998, N hereafter), Rubinstein (1986), Zemel (1989) and Papadimitriou and Yannakakis (1994, PY hereafter), have shown that, when player implement their strategies by means of finite automata, cooperation can be obtained as the equilibrium outcome in the finitely repeated Prisoner's Dilemma.

The key point to achieve most of these authors' result is to ¯ll up players's complexity in such a way that they cannot deviate from the equilibrium path. When dealing with large automata the equilibrium entails mixed strategies, whose main concern, in turn, is the choice of the set of pure strategies that belongs to the support of the mixed strategy equilibrium. The cardinality of this set is rather big; in fact, it is an exponential number on the size of the automata. To solve the choice of this set the equilibrium construction makes use of communication schemes, which play two roles: on one hand, they determine the set of plays; and on the other they specify the one which will be actually played in equilibrium.

Our paper addresses cooperation in the finitely repeated Prisoner's Dilemma, when strategies are implemented by finite automata, by means of a pseudorandom process. Given a fixed number of automata states, the pseudorandom process determines the set of possible plays to achieve the cooperative outcome. The choice of the specific play is undertaken by a communication scheme. Players follow a coordinated play to check that they are actually expending their automata full capacities along the equilibrium path.

It is important to point out that the underlying assumption of bounded rationality (automata framework) is what allows us to translate mixed strategies into pseudorandom procedures. The advantage of such procedures is that in one hand they minimize the costs of communication and on the other coordination between players can be more efficiently reached using specific sequences which are implemented by deterministic algorithms which an automaton can accept.

The equilibrium path consists of a communication process followed by a coordinated play. In the last part of this play, a verification scheme is carried out. The choice of the set of messages is vital in our communication process and thus the corresponding optimal set is related with the efficiency of the coordination procedure. We construct the minimal set of messages such that

it deters deviations by players (it fills up the players' automata capacities) and it guarantees the cooperative payoffs.

The main features of the construction are the following: 1) the choice of the set of messages is efficient since it is the minimum set with respect to the whole coordination process; 2) the messages used to determine a given coordinated play are identified by a pseudorandom process and 3) the communication process is optimal with respect to the coordination scheme, i.e. it is a "short" communication to select a specific play. This is achieved by choosing a "short" relationship among messages from the communication phase and those belonging to the verification play.

The main result is stated in terms of the size of the smallest automaton that implements the cooperative outcome and which depends on both the " ¡ approximation to the efficient outcome and the number T of repetitions. Our upper bound lies between that of PY (1994) and the one of N (1998). Namely, our upper bound includes that of PY (1994), but, in turn, it is included in the one of N (1998). The reason behind this last relationship between Neyman's upper bound and ours is that the Prisoner's Dilemma game does not belong to the class of games where the communication phase dictates the " ¡ approximation to the efficient outcome. In the class of games where this distortion rate is only generated by the communication process our bound is the largest bound to achieve cooperation in a finitely repeated game played by finite automata and it improves all the other bounds already offered in the field.

The paper is organized as follows. Section 2 sets up the model and enumerates several known results on play complexity. The main result is presented in section 3 and in section 4 we offer an informal sketch of the equilibrium path. Pseudorandom procedures are introduced in section 5 where, in particular, we analyze Linear Feedback Shift Registers and present some of their properties. The main result is proved in section 6, where we characterize the automata bounds and the equilibrium strategies are constructed . Finally, section 7 concludes the paper.

# 2 The model

## 2.1 The Prisoner's Dilemma

Let $PD$ be a Prisoner's Dilemma game. $PD=(\{1;2\};(A^i)_{i\in\{1;2\}};(r^i)_{i\in\{1;2\}})$ where $\{1;2\}$ is the set of players. $A^i=\{C;D\}$ is a finite set of actions (or pure strategies) for player $i$, where $C$ stands for cooperation and $D$ for defection. $r^i : A = A^1 \times A^2 \to R$ is the payoff function of player $i$. The payoff matrix is as follows:

| | Defect | Cooperate |
|---|---|---|
| Defect | 1,1 | 4,0 |
| Cooperate | 0,4 | 3,3 |

For any finite set $B$ we denote by $\Delta(B)$ the set of all probability distributions on $B$. An equilibrium of $PD$, is a pair $(\sigma^1;\sigma^2)\in\Delta(A^1)\times\Delta(A^2)$ such that for every $i$ and any strategy of player $i$, $\tau^i\in A^i$; $r^i(\tau^i;\sigma^{-i})\leq r^i(\sigma^1;\sigma^2)$: If $\sigma$ is an equilibrium, the payoff vector $r(\sigma)$ is called an equilibrium payoff. The only equilibrium of $PD$ is $(D;D)$ and denoting by $E(PD)$ the set of all equilibrium payoffs of $PD$, $E(PD)=(1;1)$. Let $u_i(PD)$ be the individual rational payoff to player $i$ in pure strategies, i.e., $u_i(PD)=\min\max r^i(a^i;a^{-i})$ where the max ranges over all pure strategies of player $i$, and the min ranges over all pure strategies of player $3-i$. In our $PD$ game $u_i(PD)=1$.

## 2.2 The finitely repeated game $PD^T$

From $PD$ we define a new game in strategic form $PD^T$ which models a sequence of $T$ plays of $PD$ called stages. At each stage, each player is informed of all actions played before. Thus, the information available to each player before choosing his action at stage $t$ is all past actions of the players in previous stages of the game. Formally, let $H_t;t=1;...;T$, be the cartesian product of $A$ by itself $t-1$ times, i.e.: $H_t=(A)^{t-1}$, with the common set theoretic identification $A_0=\emptyset$, and let $H=\bigcup_{t\geq 0}H_t$. A pure strategy $\sigma^i$ for player $i$ in $PD^T$ is a mapping from $H$ to $A^i;\sigma^i : H \to A^i$. Obviously, $H$ is a disjoint union of $H_t$, $t=1;...;T$ and $\sigma^i_t{:}H_t \to A^i$ is the restriction of $\sigma^i$ to $H_t$. We denote the set of all pure strategies of player $i$ in $PD^T$ by $\Sigma^i(T)$. Any 2-tuple $\sigma=(\sigma^1;\sigma^2)\in\times\Sigma^i(T)$ of pure strategies induces a play $!(\sigma)=(!_1(\sigma);...;!_T(\sigma))$

with $!_t(¾) = (!_t^1(¾); !_t^2(¾))$ defined by $!_1(¾) = (¾^1(®); ¾^2(®)) = ¾(®)$ and by induction $!_t^i(¾) = ¾^i(!_1(¾); :::; !_{t¡1}(¾)) = ¾_t^i(!_1(¾); :::; !_{t¡1}(¾))$:

Let $r_T(¾) = \frac{r(!_1(¾))+:::+r(!_T(¾))}{T}$ be the average payoff of strategy ¾:

Two strategies $¾^i$ and $¿^i$ of player $i$ in $PD^T$ are equivalent if for every 3 ¡ $i$ tuple of pure strategies $¾^{¡i}$; $!_t(¾^i; ¾^{¡i}) = !_t(¿^i; ¾^{¡i})$ for every $1 · t · T$.

An equivalence class of pure strategies is called a reduced strategy.

## 2.3 Finitely repeated games played by ¯nite automata

A finite automaton for player $i$ that implements strategies in repeated games is a tuple $M^i =< Q^i; q_0^i; f^i; g^i >$, where:

- $Q^i$ is the set of states
- $q_0^i$ is the initial state
- $f^i$ is the action function, $f^i : Q^i ! A^i$
- $g^i$ is the transition function from state to state $g^i : Q^i £ A^{¡i} ! Q^i$

The size of a finite automaton is the number of its states.

We define a new game in strategic form $PD^T(m_1; m_2)$ which denotes the $T$ stage repeated version of $PD$, with the average payoff as an evaluation criterion and with all the finite automata of size $m_i$ as the pure strategies of player $i$, $i = 1; 2$. Let $Σ^i(T; m_i)$ be the set of pure strategies of $PD^T(m_1; m_2)$ that are induced by an automaton of size $m_i$.

A finite automaton for player $i$ can be viewed as a prescription for this player to choose his actions in each stage of the repeated game. If at state $q$ the other player chooses the action tuple $a^{¡i}$, then the automaton's next state is $g^i(q; a^{¡i})$ and the action to be taken at stage 1 is $f^i(q^i)$. The action in stage 2 is $f^i(g^i(q^i; a_1^{¡i}))$ where $a_1^{¡i}$ is the action taken by the other players in stage 1. More generally, we define inductively,

$$g^i(q; a_1^{¡i}; :::; a_t^{¡i}) = g^i(g^i(q; a_1^{¡i}; :::; a_{t¡1}^{¡i}); a_t^{¡i}),$$

where $a_j^{i\,i}$ is the action of player $-i$ at stage $j$; the action prescribed by the automaton for Player $i$ at stage $t$ is $f^i(g^i(q^i; a_1^{i\,i}; ...; a_{t-1}^{i\,i}))$:

For every automaton $M^i$ for player $i$, we consider $\sigma^i_{M^i}$ the strategy in $PD^T$ such that $\sigma^i_M(a_1,...,a_{t-1}) = f^i(g^i(q^i; a_1^{i\,i}; ...; a_{t-1}^{i\,i}))$: Moreover, $\sigma^i$ for Player $i$ in $PD^T$ is implementable by the automaton $M^i$ if $\sigma^i$ is equivalent to $\sigma^i_{M^i}$ i.e.: for every $\varepsilon \in \Sigma^2(T)$; $\gamma(\sigma^i; \varepsilon) = \gamma(\sigma^i_M; \varepsilon)$:

A finite sequence of actions $(a_1; ...; a_t)$ and a pure strategy $\sigma^i$ are compatible, if for every $1 < s < t$; $\sigma^i(a_1; ...; a_{s-1}) = a_s^i$: Let $A^n(\sigma^i)$ be the set of all sequences of actions of length $n$ that are compatible with $\sigma$: We can consider for any sequence of actions $(a_1; ...; a_t)$ and $\sigma^i$ the new strategy $(\sigma^i \mid a_1; ...; a_t)$ in $PD^T$ by

$(\sigma^i \mid a_1; ...; a_s)(b_1; ...; b_{s'}) = \sigma^i(a_1; ...; a_s; b_1; ...; b_{s'})$:

The main results in automata complexity have been given by Kalai and Stanford (1988), Kakai (1990) and Neyman (1998, 1997). The number of different reduced strategies that are induced by a given pure strategy $\sigma^i$ of player $i$ in $PD^T(m_1; m_2)$ and all $\sigma^i$-compatible sequences of actions provides a first measure of the complexity of $\sigma$ (Kalai and Stanford, 1988). Neyman (1998) showed that this measure equals the size of the smallest automaton that implements $\sigma$.

## 2.4 Play complexity

As noted above the complexity of a strategy $\sigma^i$ can be defined by the size of the smallest automaton that implements it. Next we define the complexity for each player on plays and on sets of plays of the repeated game. Let $\gamma$ be a play. We define the $i-$th player complexity of a play $\gamma$, $\mathrm{comp}^i(\gamma)$; as the smallest complexity of a strategy $\sigma^i$ of player $i$ which is compatible with $\gamma$:

$\mathrm{comp}^i(\gamma) = \inf\{\mathrm{comp}^i(\sigma) : \sigma \in \Sigma^i$ is compatible with $\gamma\}$:

Let $Q$ be a set of plays. A pure strategy $\sigma^i$ of player $i$ is conformable to $Q$ if it is compatible with any $\gamma \in Q$: The complexity of player $i$ of a set of plays $Q$ is defined by the smallest complexity of a strategy $\sigma^i$ of Player $i$ that is comformable to $Q$.

$\mathrm{comp}^i(Q) = \inf\{\mathrm{comp}^i(\sigma) : \sigma \in \Sigma^i$ is comformable to $Q\}$

We consider the complexity of some particular plays which will be used in the proof of the main result. Before stating the main result it is useful to enumerate several results on complexity. We follow Neyman's approach (1998), however they are included here for completeness.

The next lemma states a lower bound of the complexity of a sequence of actions of length $t$:

**Lemma 1** Let $a = (a_1; :::; a_t) \in A^t$: Then $comp^i(a) \cdot t$:

Let $a = (a_1; :::; a_t) \in A^t$ and $b = (b_1; :::; b_s) \in A^s$; and denote by $a+b = (a_1; :::; a_t; b_1; :::; b_s) \in A^{t+s}$ the concatenation of two histories. The next lemma states the complexity bound of such a concatenation.

**Lemma 2** Let $a = (a_1; :::; a_t) \in A^t$ and $b = (b_1; :::; b_s) \in A^s$: Then $comp^i(a+b) \geq \max(comp^i(a); comp^i($

For $a = (a_1; :::; a_t) \in A^t$ and a positive integer $d$, define $d \natural a$ by induction on $d : 1 \natural a = a$ and $(d+1) \natural a = d \natural a + a$:

The complexity of a sequence of actions that changes in the last stage is stated next.

**Lemma 3** Let $a = (a_1; :::; a_t) \in A^t$ with $a_1 = a_2 = ::: = a_{t-1}$ and $a^i_{t-1} \neq a^i_t$: Then $comp^i(a) = t$:

Let $a = (a_1; :::; a_t) \in A^t$ and $b = (b_1; :::; b_s) \in A^s$; and $s$ with $\min(t; s) \geq s - 1$ then define $a =_s b$ if $a_r = b_r$ for every $r < s$:

A lower bound for the complexity of a play that consists of a cycle $(t \natural a + b)$ repeated $d$ times and there is a deviation of player $i$ from the cycle is the following,

**Lemma 4** Let $a = (a_1; :::; a_k) \in A^k$ and $b = (b_1; :::; b_n) \in A^n$ with $a^i_1 \neq b^i_1; t \geq 0$ and $d \geq 1$: Assume that $! = (!_1; :::; !_s) \in A^s$ with $(d-1)(tk+n) + tk + 1 < s \cdot (d+1)(tk+n)$ and
$d \natural (t \natural a + b) =_s !$ and $((d+1) \natural (t \natural a + b))^i_s \neq !^i_s$: Then $comp^i(!) \geq d(t+1)$:

Finally, let $f : A^1 \to A^2$ be a 1-1 function and let $a = (a_1; :::; a_n) \in A^n$ be a play with $a^2_t = f(a^1_t)$ for every $1 \cdot t \cdot n$, then $a$ is called a coordinated play. We need a complexity lower bound for a play that consists of a coordinated periodic play. This is given next.

**Lemma 5** Let $a = (a_1; :::; a_n) \in A^n$ be a coordinated play, $b \in A$ with $b^1 \neq a^1_1$; and $d \in N$: Then $comp^i(d \natural a + b) \geq (d-1)n + 1$:

# 3 The main result

The main result addresses cooperation in the finitely repeated Prisoner's Dilemma for subexponential size (of the number of repetitions) of the smallest automaton which implements it. The equilibrium is characterized by conditions on the size of automata. Let $m_1$ and $m_2$ denote the automata size of players 1 and 2, respectively. The equilibrium constraints on the automata sizes ensure both that a payoff in a sufficiently small neighborhood of the efficient outcome (3,3) is generated by strategies that are implemented by automata of sizes which are less than $m_1$ and $m_2$; and that the equilibrium path is supported by pure strategy punishments. Moreover, these conditions, in turn, depend on both the $\varepsilon$ approximation to the efficient outcome and the number $T$ of repetitions.

We construct an equilibrium which entails mixed strategies for both players. Let player 2 be the one with the bigger automaton size. Player 2 follows a mixed strategy to generate enough randomization to sustain the proposed equilibrium: payoffs $\varepsilon$ close to the cooperative payoffs. This deters player 1's deviations by forcing him to fill up his automaton complexity. Player 1 also follows a mixed strategy which supports the proposed coordination by precluding player 2's deviations.

Our approach is new in the literature. We apply a deterministic algorithm to produce pseudorandom sequences. These sequences look uniformly distributed on a finite set and they are used to construct the support of the equilibrium mixed strategy of player 2. Pseudorandom sequences are selected by two criteria: firstly their complexity and secondly their optimal codification. The first criterion bounds the complexity of the equilibrium path by selecting the subset of Player 2's pure strategies. The second criterion allows to construct a "short" communication phase to select a specific play which implies stronger security in Player 1's strategy in terms of precluding a larger number of Player 2's deviations.

Thus, we present an alternative construction to those of N (1998) and PY (1994). First, our pseudorandom process allow us to determine a unique equilibrium play in contrast with PY (1994)'s approach, where for each communication sequence ( each "business card") they have to design a different "fixed up" phase to preclude deviations from the equilibrium path, i.e., all the play after a communication message is different from all the one after a distinct message. Second, our communication phase is optimal with respect to the verification play in contrast with N (1998)'s communication phase, whose length is about twice the optimal length.

The equilibrium condition in terms of the upper bound of the smallest automaton imple-

menting the cooperative outcome and which depends on both the $\varepsilon$-approximation to the efficient outcome and the number $T$ of repetitions, is about $T^3$ and it lies between that of PY (1994) and the one of N (1998). Namely, our upper bound includes that of PY (1994), which is about $T^2$, but, in turn, it is included in the one of N (1998), which is $\exp(\varepsilon^3 T)$. However, although N (1998)'s bound is not improved, our construction (optimal communication scheme) is powerful enough to be considered as an alternative approach for a wider class of games. For instance, when dealing with games where the communication phase dictates the $\varepsilon$-approximation to the efficient outcome our approach performs better than the other constructions in the literature. Unfortunately, the Prisoner's Dilemma does not belong to this class of games.

The next theorem states our main finding:

**Theorem 1** *For every $\varepsilon > 0$; sufficiently small and for $T$ and $m_0$ large enough if $m_0 < m_1 < \varepsilon^2 T^3$ and $m_2 > T$, then there exists an equilibrium for $PD^T(m_1; m_2)$ in which the expected average payoff to each player is at least 3-$\varepsilon$.*

# 4 The scheme of the play

In this section we present the scheme of the play to reach cooperation in the finitely repeated Prisoner's Dilemma game. The plays along the equilibrium path are divided into a communication phase followed by a play phase.

Players follow first a phase of communication where the smart player, say player 2, sends a signal. This signal specifies one of the finitely many plays of the repeated game to be played in the play phase and it uses two actions that we label 0 and 1. Since player 2 proposes the plays, signals have to be independent of the associated payoffs to each of them.

Player 2 plays a mixed strategy during this phase and Player 1 responds properly to any message. The action of Player 1 is independent of the signal that player 2 is sending. Thus, the specification of the set of messages and the correspondence with the set of plays is crucial in our construction, since we associate each message from the communication phase with a unique play in the play phase.

After the communication phase the equilibrium play enters in the play phase. This phase consists of a cycle of action pairs which is a coordinated play[1]. The cycle is repeated along the play until $T$. The length of the cycle does not depend on the signal sent by player 2. Each one of the cycles has associated payoff approximately equal to the cooperative payoff. Thus, in any one of the proposed plays, player 1 has no incentive to deviate prior to the very last stages of the finitely repeated game. The cycle has two parts: the regular play and the verification play. The former has most of the action pairs as "cooperate, cooperate". The last part of the coordinated play is called the verification play.

Player 2 designs a verification play to check that player 1 has spent all his states following the play. It consists of a coordinated play with the identity as the function between $A^1$ and $A^2$ i.e., both players play the same action. In words, both players follow a monitoring phase such that the sequence of actions can be understood as a coordination process that determines each pure strategy. Nevertheless, players lose some efficiency because they will play half of the time the action pair $(0;0)$ with payoffs (1, 1), i.e.: the "minmax value", instead of the action pair $(1;1)$ with payoffs (3, 3). The sequence of actions played in this phase corresponds with the output of a LFSR whose input is the "seed" which composes the message or signal sent in the

[1] $a = (a_1;:::;a_n)$ is a coordinated play if there exists a 1-1 function $f : A^1 \to A^2$ such that $a_t^2 = f(a_t^1)$ for every $1 \cdot t \cdot n$:

communication phase. Thus, there exists a one to one relationship between each verification play and each message from the communication phase.

Therefore, the verification scheme is constructed such that it satisfies three properties. First, it is balanced (the number of 1's is equal to the number of 0's) to deter player 2's deviations by selecting the best payoff sequences. Second, this phase generates the rate of perturbation, "; with respect to the cooperative payoff. Finally, player 2 fills up player 1's capacity by generating enough pure strategies so that the number of remaining states is sufficiently small. In this way, player 1's deviations from the proposed play by counting up until the last stage of the game are avoided. For instance, player 1 could be able to select just one proposed play and deviate in the last stage of this play while repeating the cycle in all other proposed plays. Similarly, player 1 could increase his own payoff by neglecting a subset of plays. Also the repetition of the cycle precludes sophisticated deviations by player 1.

A clarification is now at hand. Player 1 has to process the message sent during the communication phase. Given our automaton framework we minimize the information processing of this player by using the same states to process the signal and to follow the regular part of the different cycles. We refer to them as the "reused states" of player 1. However, this introduces an additional difficulty since these states of the automaton of player 1 admit both actions 0 and 1. This entails that there are deviations of player 2 that might be unpunished. If player 2 knew exactly the states that admit both actions, he could take advantage over them in future stages of the game. These deviations can only be undertaken by player 2 in the play phase, since the sequences from the communication phase are balanced and thus player 2 is indifferent among them. To avoid this problem player 1 uses a mixed strategy whose support consists of the minimal subset of pure strategies which are conformable with the proposed plays and such that it generates enough uncertainty to hide the location of the reused states. Player 1's mixed strategy is constructed by a uniform distribution in this minimal subset.

Note that every player's behavior plays a different role in the game. The signaling activity of player 2 has two purposes: how to coordinate and how to fill up player 1's capacity. And these are the goals of the mixed strategy of player 2. On the contrary, player 1's role consists of supporting the coordination proposed by player 2 by means of a mixed strategy. To this end, player 1 builds a mechanism against no detectable deviations of player 2.

# 5 E ± cient pseudorandom procedures: security and complexity

## 5.1 Pseudorandom sequences and automata

Reaching efficient outcomes in the finitely Prisoner's Dilemma when players' automata sizes are sufficiently large involves the play of mixed strategy equilibria. The role of mixed strategies with finite automata is twofold. On the one hand, they allow players to reach more efficient payoffs. On the other hand, they serve the purpose of deterring deviations. To select a mixed strategy over the set of finite automata is to choose a subset of automata over a support with an exponential number of automata[2]. Coordination in this process is crucial since players have to share the same information. Therefore the problem is how to select the same subset. Obviously, there is not a unique efficient subset. Thus, reaching an efficient payoff can be viewed as equivalent to selecting a set of finite automata.

In our model, this problem can be understood as finding out an algorithm for player 2 such that the coordination can be more efficiently reached using specific sequences. Moreover, in the automata framework these sequences have to verify a complexity requirement. Namely, each sequence has to be implemented in a fixed number of states. Thus, we use sequences with pseudorandom behavior because their complexity is fixed. In other words, the length of the sequences and the associated complexity coincide.

In contrast with previous analysis in the literature (N,1998; PY, 1994) we look for random sequences implemented by deterministic algorithms that an automaton can accept. Obviously, a sequence produced by a deterministic automaton is not random. However, we generate deterministic sequences that pass various tests for randomness; such sequences are called pseudorandom sequences and they are a subset of all the sequences of a given length.

Pseudorandom generators[3] have the remarkable property of being efficient "amplifiers/expanders" of randomness, at a low cost. Using very little randomness -in form of a randomly chosen seed- they produce very long sequences which look random by efficient algorithms. Under the au-

[2]The number of automata with size at most m is of the order of an exponential function of m logm: See Neyman(1997).

[3]Pseudorandomness has been extensively used in Modern Cryptography. The reason is that the implementation of all cryptographic tasks requires a lot of "high quality random bits" at a low cost. See Gossner (2000).

tomata framework it is sufficient to guarantee that the automaton size is equal to or bigger than the length of the sequence.

We apply a pseudorandom process in the verification play. This process produces uniformly distributed sequences in a binary alphabet which ensure that their associated payoffs are the same for each of them.

More specifically, the use of pseudorandom sequences in the verification play guarantees on one hand the coordination of players in the set of pure strategies which are the support of player 2's mixed strategy and on the other the independence of player 2 from the proposed plays. Pseudorandom sequences also identify the complexity of the coordination procedure. Player 2's mixed strategy is determined by the output of the pseudorandom generator of the verification play. Since the support of player 2's mixed strategy is related with player 1's complexity, then both players's equilibrium complexity is determined by the output of the pseudorandom generator.

Moreover, under our construction, the input of the pseudorandom in the verification play is the seed which composes the signal from the communication phase. The length of the communication phase is about (of order of magnitude) the logarithm of that of the verification play. The relationship between the two lengths can be understood as a codification issue and thus we can consider the set of signals as the codification of the verification play. In this sense our codification is almost optimal (by Information Theory, Shannon, 1948) and it produces a short communication path to select a specific play. This property is important since on one hand it minimizes the number of reused states of Player 1 which, in turn, minimizes deviations by player 2 and increases the security of the equilibrium path; and on the other a short communication allows for a faster approximation to $\mathsf{x}$.

Thus, given the complexity of player 1, the pseudorandom generators determine a given security which in our context means to preclude both players' deviations.

From an implementation point of view, a commonly employed method of generating pseudorandom sequences is based on the use of suitable linear recurrence relationships in a finite field of two elements ($\mathsf{F}_2$, henceforth). Sequences in finite fields whose terms depend in a simple manner on their predecessors are of importance for a variety of applications. Such sequences are easy to generate by recursive procedures, which is certainly an advantageous feature from a computational viewpoint, and they also tend to have some useful structural properties.

The next section presents formally the Linear Feedback Shift Register procedure, LFSR.

The LFSR can be produced by machines with a simple hardware. Sequences generated by an LFSR are $n$-periodic. This means that the procedure is implementable by an automaton since sequences have a finite period, in contrast with other generators of pseudorandom sequences which need a sophisticated hardware.

## 5.2 Linear Feedback Shift Registers

An $n$-stage Linear Feedback Shift Register (LFSR) is a procedure to generate pseudorandom sequences. It consists of a shift register $R_0 = (r_n; r_{n-1}; \ldots; r_1)$, the seed, and a tap sequence $T = (t_n; t_{n-1}; \ldots; t_1)$ where each $r_i$ and $t_i$ is a binary digit. Let $H$ be the square matrix, where the first row is the vector $T$, the $n-1$ minor is the identity matrix and the last column is a vector of $0$'s. By induction, let $R_i = HR_{i-1}$, for $i = 1; \ldots 2^n - 1$, where

$$H = \begin{array}{c} 0 \\ \\ \\ \\ \\ \\ \\ \\ @ \end{array} \begin{array}{c} \begin{array}{ccccccc} t_n & t_{n-1} & t_{n-2} & \cdots & t_3 & t_2 & t_1 \end{array} \\ \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 0 & 1 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 0 & 0 \\ 0 & 0 & 0 & \cdots & 0 & 1 & 0 \end{bmatrix} \end{array} \begin{array}{c} 1 \\ \\ \\ \\ \\ \\ \\ \\ A \end{array}$$

Consider the output matrix, whose columns are the different $R_i$, with $i = 0; 1; \ldots n$. The output of the LFSR is the last row of such a matrix.

An $n$-stage LFSR generates a pseudorandom bit string. The randomness properties of this sequence depend on the characteristic polynomial $T(x) \in F_2$ associated with the tap sequence $T$. Specifically, the polynomial

$T(x) = t_1 x^n + t_{n-1} x^{n-1} + \ldots + t_n x + 1$

is formed by the elements of the tap sequence plus the constant 1.

A sequence generated by an LFSR with primitive polynomial[4] is called an "$n$-sequence". The length of the sequence does not depend on the initial values, i.e. the "seed". Also the sequence has a period of $2^n - 1$ nonzero bit sequences before repeating itself, which means that the period is maximal. Moreover the length of $R^0$ is the logarithm of that of the output, which

[4] A primitive polynomial of degree n is an irreducible polynomial that divides $x^{2^n-1} + 1$ but not $x^d + 1$ for any d that divides $2^n + 1$.

can be understood as an optimal codification property. Sequences generated by a different associated polynomial such as an irreducible polynomial or a factorizable polynomial, have different properties. For instance, the period of the output produced by an irreducible[5] polynomial is a divisor of $2^n - 1$; which means that the period may not be maximal; and the length of the output generated by a factorizable polynomial depends on the seed.

The next example shows how to generate pseudorandom sequences by an LFSR with primitive polynomial. Let $T(x) = x^4 + x + 1$ be a primitive polynomial of degree 4 on a Galois field $GF(2)$. Let $R_0 = (0;1;1;0)$ be the seed. The output of the LFSR associated with this polynomial is produced in the following way,

$$\text{Let } R_1 = HR_0 = \begin{matrix} 0 & & & & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & & 0 & & 0 & \\ 1 & 0 & 0 & 0 & & 1 & & 0 & \\ 0 & 1 & 0 & 0 & & 1 & & 1 & \\ 0 & 0 & 1 & 0 & & 0 & & 1 & \end{matrix}$$

$$\text{Let } R_2 = HR_1 = \begin{matrix} 0 & & & & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & & 0 & & 1 & \\ 1 & 0 & 0 & 0 & & 0 & & 0 & \\ 0 & 1 & 0 & 0 & & 1 & & 0 & \\ 0 & 0 & 1 & 0 & & 1 & & 1 & \end{matrix}$$

We calculate $R_i$ for $i = 0;:::;14$: The output vector is represented by a matrix with 4 rows and 15 columns (from $R_0$ to $R_{14}$ ). Notice that columns are different from each other.

$$\begin{matrix} 0 & & & & & & & & & & & & & & & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & 1 & \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 1 & \\ 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & \end{matrix}$$

The output of the LFSR associated to $T(x)$ corresponds with the last row of the above matrix: i.e.,

$$0\ 1\ 1\ 0\ 0\ 1\ 0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 1$$

Notice that $R_{15} = HR_{14} = (0;1;1;0) = R_0$: Hence, the period of the LFSR associated with the primitive polynomial $T(x)$ is 15, which is maximal since $15 = 2^4 - 1$:

[5] A polynomial $T(x) \in K[x]$ is irreducible if there are no polynomials $R(x)$, and $S(x) \in K[x]$, and with $R(x) \neq I$ and $S(x) \neq I$, such that $T(x) = R(x)S(x)$:

## 5.3 Properties of LFSR with primitive polynomial:

The main properties of pseudorandom sequences generated by LFSR with primitive polynomial are that their output length is independent of the seed, they have maximal period and their codification is optimal, and thus they provide with the necessary conditions[6] to mimic uniform random sequences. These pseudorandom sequences can be expected to have some statistical properties satisfied by those sequences of independent random variables which attain each value 0 and 1 with probability $\frac{1}{2}$. Thus, the number of blocks of either ones or zeros has to be the same. This means that the relative frequency of each bit will approach $\frac{1}{2}$ in the long run, or in other words, that sequences are balanced. Also, the relative frequency of two successive 0's (or of two successive 1's) will approach $\frac{1}{4}$ in the long run and so on. More generally[7], for any given block of $m$ bits the relative frequency of this block among all the blocks of $m$ successive bits in the sequence will approach $2^{-m}$ in the long run. This property is known as the distribution test and the serial test.

The sequences produced by our LFSR with primitive polynomial satisfy the above requirements. To see that let $b = (b_1; :::b_m) \in F_2^m$, where $F_2$ is a finite field with two elements. Let $Z(b)$ be the number of $n$, $1 \cdot n \cdot r - 1$; where $r$ is the sequences' period, i. e., $r = 2^m - 1$, and such that $s_{n+i-1} = b_i$ for $1 \cdot i \cdot m$: The case $m = 1$ corresponds with the distribution test. The case $m \geq 2$ corresponds with the serial test for blocks of length $m$. The following result shows that $Z(b)$ is equal to the number $r2^{-m}$ provided that $m$ is not large.

Proposition 1:(Nidl and Niederreiter, 1983)
If $1 \cdot m \cdot k$ and $b \in F_2^m$, then for any kth-order maximal period sequence in $F_2$ we have

$$Z(b) = \begin{cases} 2^{k-m} - 1 \text{ for } b = 0 \\ 2^{k-m} \text{ for } b \neq 0 \end{cases}$$

[6] These properties are called "the Golomb's randomness postulates". They were one of the first attempts to establish some necessary conditions for a periodic pseudorandom sequence to look random.

[7] For a more complete analysis, consult Nidl and Niederreiter, 1983, Chapter 8.

We will use the output of an LFSR with primitive polynomial to construct the verification play. Proposition 1 guarantees that the pseudorandom sequences used in our construction behave as uniformly distributed sequences and thus they preclude deviations by players along the equilibrium path.

We assume that for each length $2^k$ of the LFSR sequences, players consider all the possible primitive polynomial of degree $k$. Notice that the number of primitive polynomial of degree $n$ is $\frac{\acute{A}(2^n - 1)}{n}$ where $\acute{A}$ is the Euler function and that $\frac{\acute{A}(2^n - 1)}{n}$ is equal[8] to $\frac{2^n - 1}{n \log n}$.

[8] See Hardly and Wright: An Introduction to the Theory of Numbers, Chapter XVIII.

# 6 Proof of the main result

## 6.1 Notation

We use the following notations . Label the players actions by 0 and 1, with 0 corresponding with "defect" and 1 with "cooperate". The strategy $b^2$ of player 2 in the one shot game $PD$ is a best reply of player 2 to the strategy $a^1$ of player 1. The pure strategy $a^i$ is denoted by 1. Let $D^i$ be the punishing strategy of player $i$, i.e., player $i$'s strategy that holds player $3 - i$ down to $u^{3-i}(PD)$. Note that $D^i \neq a^i$, and denote the pure strategy $D^i$ by 0.

We consider the case of "smart" players. For $m_i < T/4$; $i = 1; 2$ see the paper by Neyman (1998). Then, assume that $\min(m_1; m_2) > T/4$. We build up a mixed strategy equilibrium. To this end, the next sections construct the set of messages and the players' equilibrium strategies. To check that they are indeed an equilibrium we show that there do not exist profitable deviations.

## 6.2 De¯nition of the play

We define first the play to construct an equilibrium point $(\sigma^*; \tau^*)$ of $PD^T(m_1; m_2)$, with $m_0 \cdot m_1 \cdot T^3$ , for $T$ large enough, and with stage game as in section 2. The associated equilibrium payoff vector is $(y^1; y^2)$ where $|y^i - r^i(1; 1)| < \varepsilon$, or $y^i = 3 - \varepsilon$:

The mixed equilibrium strategy of player 2, $\tau^*$, chooses randomly a pure strategy $\tau^\omega$ where $\omega$ is an element of the message space $Q$. The message space $Q$ is the set of sequences of zeros and ones of length $4k - 2$, where $k$ depends on both the length of the game, $T$ and on player 1' automaton size $m_1$.

Each pure strategy $\sigma$ of player 1 and the pure strategy $\tau^\omega$ of player 2 induce a play $\rho(\sigma; \tau^\omega) = (\rho_1(\sigma; \tau^\omega); ...; \rho_T(\sigma; \tau^\omega))$ that depends on $\omega$, and therefore we may denote it by $\rho(\omega) = (\rho_1(\omega); ...; \rho_T(\omega))$ and call it the proposed play. The payoff associated to $\rho(\sigma; \tau^\omega)$ does not depend on the selected message, which we call the seed, $\omega$.

Player 2 communicates his choice of $\omega$ in $Q$ at the beginning of the play to player 1, who processes this information. The action of player 1 in the communication phase is independent of $\omega$ and player 2 communicates the proposed play $\rho(\omega)$ in the first $4k - 2$ stages of the game. After the communication phase, the proposed play enters a cycle of length $l$ (excluding the last stages of the game). Both players cooperate during the first $l - 2^k$ stages of the cycle, i.e.,

$!_t(\sigma) = (1;1)$ whenever $t < T$ and $t \pmod{l} < l - 2^k$. The remaining $2^k$ stages correspond with the verification play. Since the play here is the output of a LFSR with primitive polynomial the length of this play is maximal with respect to the "seed" (of length $k$). Let $\mu^\alpha(\sigma)$ denote the output of this LFSR, then after the initial $l - 2^k$ stages of the cycle, the players play the string $((\mu_1^\alpha(\sigma);\mu_1^\alpha(\sigma));:::;((\mu_{2^k}^\alpha(\sigma);\mu_{2^k}^\alpha(\sigma)))$, i.e.,for $t > l$ with $l - 2^k + 4k - 2 < t \pmod{l} \cdot l + 4k - 2$, $!_t(\sigma) = (\mu_{t(\mod l)}^\alpha(\sigma);\mu_{t(\mod l)}^\alpha(\sigma))$:

The strategy of player 1 detects with positive probability any deviation by player 2: some of his deviations are immediately detected with positive probability, and others will lead to a detection with positive probability in a future stage. The strategy of player 1 triggers to punishing (playing $D^i$ ) forever once he detects a deviation. We turn now to the formal construction of the proposed play.

## 6.3 The set of messages

We start with the construction of the set $Q$, and the integers $k$ and $l$. Let $k = k(m_1;l)$, be the smallest integer such that $2^{2k}l > m_1 - l$. We will see that the number of pure strategies for player 2 is at most $2^k$ and by Lemma 3 the complexity of each pure strategy of player 1 is at least $l$, filling up, in this way, player 1's complexity. It follows that $2^{2k-2} \cdot [\frac{m_1-l}{l}] < 2^{2k}$.

Recall that $l$ is the length of the cycle. For every $T$ (the length of the game), the cycle has to be repeated a large number of times, $L$. To ensure that at the end of the repeated game player 1 is in the regular play (not in the verification play) we choose $l = [\frac{T}{L+\pm}]$ where $\frac{1}{2} < \pm < 1$ and $T - L[\frac{T}{L+\pm}] < l - 2^k$. To deter deviations $L$ has to be greater than[9] 3. Thus, assume that $L = 4$.

To build the set of messages, consider the set of sequences for the verification play. Recall that each verification sequence is the output of a LFSR , where this output is identified by both the seed and the associated primitive polynomial . Each seed is a sequence of zeros and ones of length $k$. As the coefficients of the primitive polynomial are elements over a finite alphabet (in our case $f0;1g$), we can represent each polynomial as a sequence of zeros and ones whose length is the polynomial's degree plus 1. If $T(x) = t_1x^k + t_{k-1}x^{k-1} + ::: + t_kx + 1$ is a primitive polynomial then $t_1 = 1$ and there exists $1 < j \cdot k$ such that $t_j \neq 0$: Hence, it is enough to consider $k - 1$ coefficients to identify a primitive polynomial. The sequence obtained by the

[9] See the proof of lemma 9 below.

concatenation of a seed and the coefficients of the polynomial determines uniquely the output of the LFSR associated to this polynomial with this seed. To keep the same preferences among the messages, the balancedness property has to be verified and then we need at most $2k - 1$ stages to balance the above sequences.

We produce a message $\sigma = (\sigma_1, \ldots, \sigma_{4k-2})$ from each verification sequence in the following way. Since each verification sequence belongs to the output associated to the seed $s = (s_1, \ldots, s_k) \in \{0,1\}^k$ and the primitive polynomial $T(x) = t_1 x^k + t_{k-1} x^{k-1} + \ldots + t_k x + 1$, the message consists of the concatenation of the seed plus the coefficients of the polynomial plus a string of ones and zeros of length $2k - 1$ such that the number of zeros is equal to the number of ones. Then

$$Q = \{\sigma = (\sigma_1, \ldots, \sigma_{4k-2}) = (s_1, \ldots, s_k, t_2, \ldots, t_k, 1, \ldots, 1, 0, \ldots, 0) \in \{0,1\}^{4k-2} \mid \sum_{i=1}^{4k-2} \sigma_i = 2k - 1\}$$

Notice that an optimal codification of the verification sequences would produce sequences (messages) of length $2k$. Our communication sequences are almost optimal, since their length is of the same order of magnitude ($< 4k$).

### The associated play to a given message

To every $\sigma$ we associate a play $\gamma(\sigma)$ of $PD^T$, i.e., a sequence $\gamma(\sigma) = (\gamma_1(\sigma), \ldots, \gamma_T(\sigma))$ with $\gamma_t(\sigma) = (\gamma_t^1(\sigma), \gamma_t^2(\sigma))$ in $A$ as follows: $\gamma_T(\sigma) = (1, b^2)$ and for $t < T$,

$$\gamma_t(\sigma) = \begin{cases} (1, \sigma_t) & \text{if } 0 \le t \le 4k-2 \\ (1,1) & \text{if } 4k-2 < t \bmod l \le l - 2^k + 4k - 2 \\ (\mu_i^*(\sigma), \mu_i^*(\sigma)) & \text{if } t \bmod l = l - 2^k + 4k - 2 + i < l \\ (0,0) & \text{if } t \bmod l = 4k-2. \end{cases}$$

The first row corresponds to the communication phase when player 2 sends the message $\sigma$ and player 1 plays 1. The second row refers to the regular play of the cycle where both players play 1, "cooperate". The verification play is represented by the third row. The last one is the end of the verification phase where both players play 0, "defect".

Setting

$$\mu^0(\sigma) = ((1, \sigma_1), \ldots, (1, \sigma_{4k-2}))$$

and recalling that

$$\mu^*(\sigma) = ((\mu_1(\sigma), \mu_1(\sigma), \ldots, (\mu_{2^k}(\sigma), \mu_{2^k}(\sigma))$$

Let $d = T - 2^k - l$, then (the notation is defined in section 2.4)

$!(²) = \mu^{0}(²) + L((l\ ¡\ 2^{k})\ ¤\ (1;1) + \mu^{¤}(²)) + (d\ ¡\ 1)(1;1) + (1;b^{2})$:

Players communicate and then follow a cycle repeated until $T$ where both players play a coordinated play which has two parts: a regular play and a verification play. Player 2 plays the best response $b^{2}$ in the last stage of the play.

## 6.4 Complexity bounds

Our approach shows how to achieve cooperation in the $PD^{T}(m_1; m_2)$ by a mixed strategy equilibrium in terms of player 1's complexity bounds -as functions of the number of repetitions- and the "-approximation to $x$. We study the behavior of these bounds to characterize them in our model. Complexity bounds provide the constraints to induce the mixed strategy equilibrium.

Let $f$ be a real function.

De¯nitions:

$f$ grows polynomially is denoted by $f = O(p)$ for some polynomial $p$ i:e: : $f = n^{O(1)}$:

$f$ grows exponentially is denoted by $f = \Omega(2^{n^{²}})$ for some $² > 0$; i:e : $f > 2^{n^{²}}$

$f$ grows subexponentially is denoted by $f = o(2^{n^{²}})$; i:e: : $8² > 0\ \frac{f}{2^{n^{²}}} < ²$ for all large enough $n$:

First, we study the lower complexity bound of player 1 which in turn will determine the upper bound.

The lower complexity bound is related to the length of both the communication phase and the verification play. The length of the first is $4k ¡ 2$, meanwhile that of the second is $2^{k}$: Also, given " and the cooperative payoff, $k$ is such that it verifies the restriction:

$ky ¡ r(1;1)k \cdot "$

Then, $ky ¡ r(1;1)k = \left\| \frac{1}{T} \sum_{t=1}^{T} r_t(!(²)) ¡ (3;3) \right\| =$

$\left\| \frac{1}{T}((2k ¡ 1)r(1;0) + L\frac{2k}{2}r(0;0) + r(1;b^{2}) + (T ¡ L\frac{2k}{2} ¡ (2k ¡ 1) ¡ 1)r(1;1)) ¡ \frac{T(3;3)}{T} \right\|$

Since by assumption $L = 4$, then to check the above expression is equivalent to checking:

$\left\| (2k ¡ 1 + 1)(r(1;0) ¡ r(1;1)) + 2^{k+1}(r(1;1) ¡ r(0;0)) \right\| \cdot "T$

We obtain that $2^{k+2} + 4k \cdot "T$.

The $k$ which solves the above inequality is:

$k \cdot \frac{¡ \text{LambertW}^{3}(\ln 2) 2^{\frac{1}{4}"T}}{\ln 2} + \frac{"T}{4}$

Then the number of states used in the verification play is about

$$2^k = 2^{\left\lfloor \frac{\text{LambertW}\left((\ln 2)e^{(\ln 2)\frac{\varepsilon T}{4}}\right)}{\ln 2} + \frac{\varepsilon T}{4} \right\rfloor}.$$

Next, the upper bound of player 1's complexity satisfies the constraint

$$2^{2k+2} \cdot \left[\frac{m_1 + l}{l}\right] < 2^{2k}$$

Then, $m_1 < \frac{T}{4}(\exp^{3}(\text{LambertW}((\ln 2)\, 2^{\frac{1}{4}\varepsilon T}))^{3} - 2^{\frac{1}{4}\varepsilon T})^2$

Simplifying the above expression, $m_1 < \frac{1}{4}\frac{T}{\ln^2 2}\text{LambertW}^2((\ln 2)\, 2^{\frac{1}{4}\varepsilon T})^{3}$

To study the behavior of $m_1$ we compute the limit of the upper bound with respect to $T$; $T^2$ and $T^3$; when $T$ tends to infinity.

$$\lim_{T \to \infty} \frac{\frac{1}{4}\frac{T}{\ln^2 2}\text{LambertW}^2((\ln 2)2^{\frac{1}{4}\varepsilon T})^{3}}{T^{3}} = \text{signum}^2(\varepsilon)\ 1 = +1$$

$$\lim_{T \to \infty} \frac{\frac{1}{4}\frac{T}{\ln^2 2}\text{LambertW}^2((\ln 2)2^{\frac{1}{4}\varepsilon T})^{3}}{T^2} = \text{signum}^2(\varepsilon)\ 1 = +1$$

$$\lim_{T \to \infty} \frac{\frac{1}{4}\frac{T}{\ln^2 2}\text{LambertW}^2((\ln 2)2^{\frac{1}{4}\varepsilon T})^{3}}{T^3} = 1{:}562\,5 \pounds 10^{-2}\varepsilon^2 > 0$$

Then, the upper bound of $m_1$ is lower than $\varepsilon^2 T^3$.

## 6.5 Properties of the associated play.

For $T$ large enough, the payoff reached by the proposed play is $\varepsilon$ close to the cooperative payoff and is independent of the seed. Let $R(!(\varepsilon)) = \frac{1}{T}\sum_{t=1}^{T} r(!_t(\varepsilon))$.

**Lemma 6** *The payoff vector $\sum_{t=1}^{T} r(!_t(\varepsilon))$ is independent of $\varepsilon$, and for sufficiently large values of $T$,*

$$\|R(!(\varepsilon)) - r(1;1)\| < \varepsilon.$$

*Proof:*

For every $\varepsilon \in Q$ the number of ones and zeros coincides in both the communication and the verification phases. Thus, payoffs to any player in these phases are

$$\sum_{i=1}^{4k^2} r(\mu^0(\varepsilon)) = (2k - 1)r(1;1) + (2k - 1)r(1;0)$$

$$\sum_{i=1}^{2k} r(\mu^*(\varepsilon)) = 2^{k-1}r(1;1) + 2^{k-1}r(0;0)$$

Therefore, $\sum_{t=1}^{T} r(!_t(\sigma)) = (2k - 1)r(1;0) + L2^{k-1}r(0;0) + r(1;b^2) + (T - L2^{k-1} - 2k + 1 - 1)r(1;1)$

The right hand of the above equality does not depend on $\sigma$: Moreover for $T$ large enough $\frac{(L2^k+2k-1+1)}{T}$ is small and then $\frac{1}{T}\sum_{t=1}^{T} r(!_t(\sigma)$ converges to $r(1;1)$. ¤

We analyze next the play complexity for both players, or $comp^1(Q)$ and $comp^2(!(\sigma))$ where $Q$ can be understood as the set of plays $!(\sigma)$ for $\sigma \in Q$: It is trivial that $comp^2(!(\sigma)) = T + 1$: Lemma 2 establishes a lower bound for player 1's complexity on the set of plays $!(\sigma)$ for every $\sigma \in Q$: Notice that players follow a coordinated play after the communication phase. The complexity for player 1 of the coordinated play is exactly the period of the path that coincides with the length of the cycle $l$ for every $\sigma \in Q$: Hence, a lower bound of player 1's complexity is the number of the different coordinated plays. The next lemma shows that this number coincides with $|Q|/l$:

**Lemma 7** For every $(\sigma;t);(\sigma^0;t^0) \in Q \times \{1;...;l\}$ with $(\sigma;t) \neq (\sigma^0;t^0)$ $(!_t(\sigma);...!_{t+l-1}(\sigma)) \neq (!_{t^0}(\sigma^0);...!_{t^0+l-1}(\sigma^0))$:

**Proof:**

Both players follow a coordinated play after the communication phase until the $T - 1$ stage. Hence, it is enough to prove that for any pair $(\sigma;t) \neq (\sigma^0;t)$ there exists $0 \leq s \leq l$ with $(!_t(\sigma);...;!_{t+s}(\sigma)) \neq (!_{t^0}(\sigma^0);...;!_{t^0+s}(\sigma^0))$; or that there exists $0 \leq s \leq l$ with $!_{t+s}(\sigma) \neq !_{t^0+s}(\sigma^0)$:

Suppose that $t = t^0$ and thus $\sigma \neq \sigma^0$.We know that the mapping associated[10] to the LFSR is 1-1. This means that $\mu^¤(\sigma) \neq \mu^¤(\sigma^0)$. Therefore there exists $0 \leq s^0 < 2^k$ with $\mu^¤_{s^0}(\sigma) \neq \mu^¤_{s^0}(\sigma^0)$: Let $s = t + s^0$ such that $0 \leq s \leq l$: We conclude that $!_{t+s}(\sigma) \neq !_{t+s}(\sigma^0)$:

Next, suppose that $t \neq t^0$. We can always choose one $s$ such that the $!_t(\sigma)$ is in the regular part and $!_{t^0+s} = !_{l+4k-2}$. Then $!_{t+s}(\sigma) = (1;1)$ and $!_{t^0+2^k}(\sigma^0) = (0;0)$: More specifically, suppose that $t < t^0$: If $t^0 - t > l - 4k - 2$; setting $s = l - t + 4k - 2^0$; then $t^0 + s = l + 4k - 2$ then $!_{t^0+s}(\sigma^0) = !_{l+4k-2}(\sigma^0) = (0;0)$ and $!_{t+s}(\sigma) = (1;1)$: If $t^0 - t \leq l - 4k - 2$; setting $s = 4k - 2 + l - t$; then $!_{t+s}(\sigma) = !_{l+4k-2}(\sigma) = (0;0)$ and $!_{t^0+s}(\sigma^0) = (1;1)$. Note that this choice is independent of $\sigma;\sigma^0 \in Q$. ¤

[10] The LFSR can be viewed as a mapping between two s-dimensional vector spaces.

## 6.6 Construction of player 2's equilibrium strategy

We construct now the equilibrium strategy of player 2. It consists of a mixed strategy with $|Q|$ pure strategies. To every $\alpha \in Q$, a proposed play $!(\alpha)$ is associated with a pure strategy in the support of $\mu^{\alpha}$; the equilibrium mixed strategy. Player 2 follows the proposed play and punishes forever as soon as he detects a deviation. Thus, for any $\alpha \in Q$ , $\mu^{\alpha} = (\mu_t^{\alpha})_{t=1}^{T}$ is the pure strategy of player 2 defined by,

$$\mu_t^{\alpha}(s_1, \ldots, s_{t-1}) = \begin{cases} !_t^2(\alpha) & \text{if} \quad (s_1, \ldots, s_{t-1}) = (!_1(\alpha), \ldots, !_{t-1}(\alpha)), \\ 0 & \text{otherwise} \end{cases}$$

The mixed strategy $\mu^{\alpha}$ informs player 1 about the chosen message $\alpha$ . As already noted, no message gives any advantage to player 2, since both the communication and verification phases consist of balanced sequences. Also recall that the associated payoff to each message coincides for all of them. Thus, there is no strategic behavior from player 2 in the communication phase. Then, he follows the proposed cycle until the end of the game where there exists a verification play related to the chosen message. The pure strategy $\mu^{\alpha} \in \Sigma^2(T, T+1)$, i.e.,$\mu^{\alpha}$ is implemented by an automaton $< \{1, \ldots, T, T+1\}, 1, f_{\alpha}^2, g_{\alpha}^2 >$ of size $T+1$ where:

- $\{1, \ldots, T, T+1\}$ is the set of states.
- 1 is the initial state.
- The action function $f_{\alpha}^2$ defined by $f_{\alpha}^2(t) = !_t^2(\alpha)$ if $t \leq T$; $f_{\alpha}^2(T+1) = D^2$.
- The transition function $g_{\alpha}^2$; defined by $g_{\alpha}^2(t, a) = t+1$ if $a = !_t^1(\alpha)$ and $t \leq T$, and $g_{\alpha}^2(t, a) = T+1$ otherwise, i.e., if $a \neq !_t^1(\alpha)$, or if $t = T+1$.

## 6.7 Construction of player 1's equilibrium strategy

The equilibrium strategy of player 1 consists of a mixed strategy. Player 1 has to answer correctly to any seed sent by player 2. Thus, each pure strategy belonging to the support of the mixed equilibrium strategy must be conformable with the set of plays $\{!(\alpha) : \alpha \in Q\}$: Player 1 has to process the information sent by player 2 in the communication phase. To this end, he uses the same states to be used in the regular play. Recall that this states admit both actions

and then deviations by player 2 are not detectable. To conceal the location of these reused states player 1 uses a mixed strategy. The difference among the pure strategies in its support is the location of the communication phase in the regular play. The mixed strategy of player 1 is a uniform distribution over a minimal subset of pure strategy $¾ \in \Sigma^1(m_1)$ where $¾$ is conformable with $f!(²) : ² \in Qg$: This set is the minimal set which produces enough uncertainty for player 2 about the actual location of the reused states.

Let us construct an automaton that implements a pure strategy $¾ \in \Sigma^1(m_1)$ conformable with $f!(²) : ² \in Qg$: Firstly, we design an automaton that implements the cycle and then we define the transition function in the communication phase.

### 6.7.1 The Automaton of player 1

**The cycle:** The cycle is the deterministic part of player 1's automaton, i.e., it does not depend on the pure strategy selected by player 1. The equilibrium strategy of player 1 has to follow a cycle whose length is $l$. In the first $l - 2^k$ stages of the cycle, player 1 will play the same action, "cooperate", if player 2 does it. In addition, he has to count until $l - 2^k + 1$ because he needs to know when to start the verification play.

The mixed equilibrium strategy of player 1, $¾^¤ \in \Delta(\Sigma(m_1; T))$; is a mixture of pure strategies, each one being implemented by an automaton with state space

$$M^1 = f®g \cup Q \times f1; :::; lg$$

The action function of the automaton is given by,

$$f^1(®) = D^1;$$

and

$$f^i(²; j) = \begin{cases} 1 & \text{if } 0 < j \leq l - s \\ \mu_j(²) & \text{if } l - s < j \leq l \\ 0 & \text{if } j = l \end{cases}$$

We visualize the states of the automaton of the form $(²; j)$ as arranged in a rectangular array with $|Q|$ rows and $l$ columns. The rows are indexed by the different elements $²$ in $Q$ and the columns are indexed by 1,...,$l$. Thus the action function assigns the action $a^1$ to each state in the first $l - 2^k$ columns , and in the remaining $2^k$ columns an action that depends on the

row (verification phase). We may think that every row corresponds with each pure strategy of player 2. The number of columns is the length of the cycle.

Figure 1 illustrates the automaton of player 1 for $k = 3$; and $|Q| = (2^3 - 1)\frac{Á(2^3 - 1)}{3} = 7\frac{6}{2} = 14$: Suppose that the regular play has 12 columns and that the verification play has $2^3 = 8$ associated columns. The filled disks (²) represent states of the automaton whose action function is 1, "cooperate", when player 2 plays 1 as well. The small disks (±) represent states that play the action 0, "defect", when player 2 plays 0. The big disks ( ° ) mean the final states of the verification phase where both players have to play "defect" at the same time. The transition function in this state goes to the first state in the same row. The horizontal arrows indicate the transition of the automaton when player 1 follows a coordinated play.

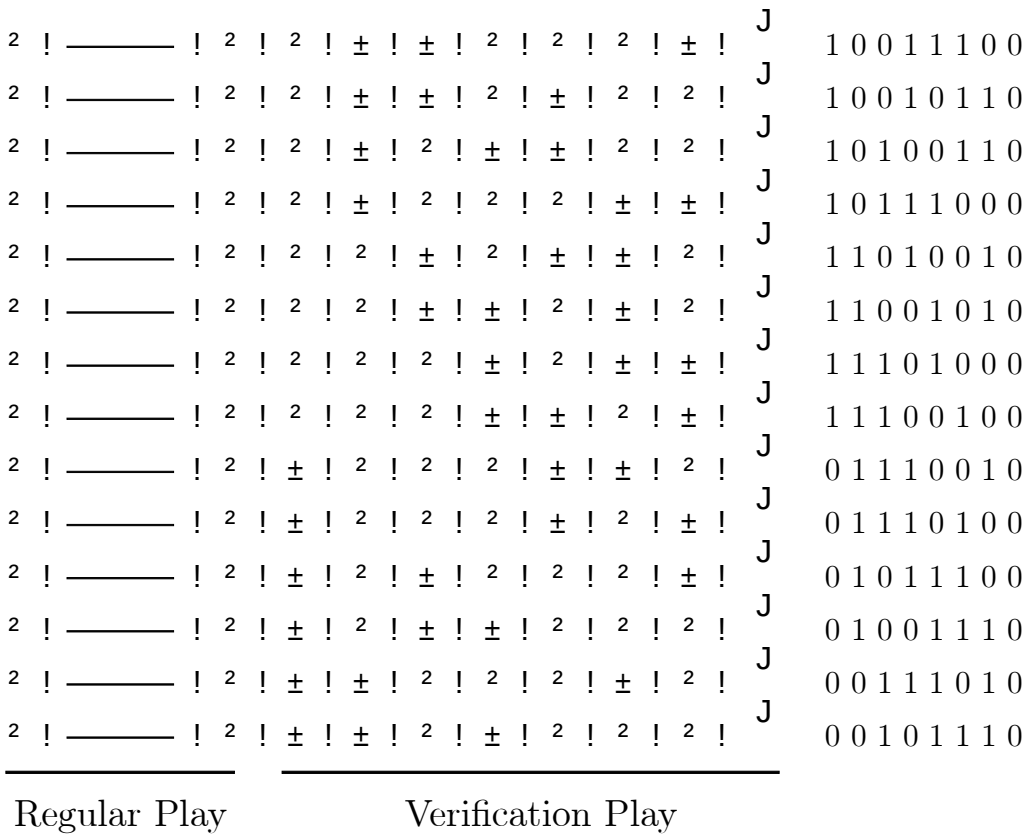


Figure 1.

The transitions of the automaton will be defined such that for each fixed $² \in Q$, if player 2's strategy is ¿², the state of the automaton at stage $t = j \bmod(l + 4k - 2)$ with $0 \leq j \leq l$, is

$(z; j)$. This leads to the following transitions:

$$g^1((z; j); 1) = \begin{cases} (z; j+1) & \text{if} \quad 0 < j \cdot l - 2^k \\ (z; j+1) & \text{if} \quad \mu_j(z) = 1 \quad \text{and } l - 2^k \cdot j \cdot l \end{cases}$$

$$g^1((z; j); 0) = \begin{cases} (z; j+1) & \text{if} \quad l - 2^k < j < l \quad \text{and } \mu_j(z) = 0 \\ (z; 1) & \text{if} \quad j = l \end{cases}$$

Player 1 remains in the same row and goes to the next column in case player 2 plays correctly in the verification phase. At the end of the verification play player 1 goes to the first column in this row, i.e., he starts another repetition of the cycle if player 2 plays 0 in this stage.

The states of the automaton of the form $l - 2^k < j < l$ implement a coordinated play. Any deviation from this play at these states results in punishing forever.

$g^1((z; j); e) = ®$ if $l - 2^k < j \cdot l$ and $\mu_j(z) \neq e$
$g^1(f®g; ¤) = ®$:

The communication phase: In the communication phase player 1 has to process the information sent by player 2. To this end, he uses the same states to be used in the regular play. We design the transition function for the first $4k - 2$ stages such that player 1 follows a specific play after the communication phase and he conceals his reused states by means of changing their locations in his pure strategies. In other words, each pure strategy in the support of player 1's mixed strategy is designed such that he selects the right row along the communication phase and he does not reveal which states admit both actions.

The transition function of the automaton of player 1 in the communication phase depends on the pure strategies in the support of ¾¤. Each pure strategy is given by two random integers, $p \in \{1; l - 2^k - 6k + 3\}$ and $n \in \{1; 2\}$. The first one determines the initial state of the automaton. We denote this initial state by $(1; p)$. Thus, $p$ is the column where player 1 processes the signal sent by player 2. To select the range of $p$ it is necessary to know the location of the last state in the communication phase. To this end, we choose this range such that it ensures that the processing scheme entirely lies on the regular play.

The random number $n \in [2; k]$ determines the jumps in the columns (along the same row) that player 1 will follow in the communication phase when player 2 sends a 0 after the first $2k - 1$ stages.

The transition function of the communication phase consists of three parts: the first one corresponds to the first $2k - 1$ stages; the second to stages $2k$ to $4k - 3$, and finally the third part corresponds to the last stage of the communication.

Thus, to select the right row during the first $2k - 1$ stages[11], the transition function is moving between the different rows in such a way that in stage $4k - 1$ the state of the automaton is in the row that corresponds to the sequence of actions of player 2 in the first $2k - 1$ stages of the game. This is achieved through the following partial transition function. If $\sigma = (\sigma_1, \ldots, \sigma_{4k-2})$

$$g^1((\sigma, j), 0) = ((\sigma_1, \ldots, \sigma_{j-p}, 0, 1, \ldots, 1, \ldots, 0), j + 1) \quad \text{if} \quad p \cdot j \cdot p + k$$

Figure 2 illustrates the communication phase associated to the verification play in the above example. The star ($?$) is the initial state. The diamonds ($\diamond$) represent those states in the regular play that are used to process the information sent by player 2 in the communication phase, and thus admit both actions 0 and 1 from player 2. The big states with a dot are the states in the regular play that player 1 uses to determine the end of the communication phase. These states also admit both actions, 0 and 1.

[11] Notice that player 1 is processing the information sent by player 2 by trying to recognize the different sequences. By the properties of the LFSR, player 1 needs $k$ stages to identify the seed and $k - 1$ stages to identify the primitive polynomial.

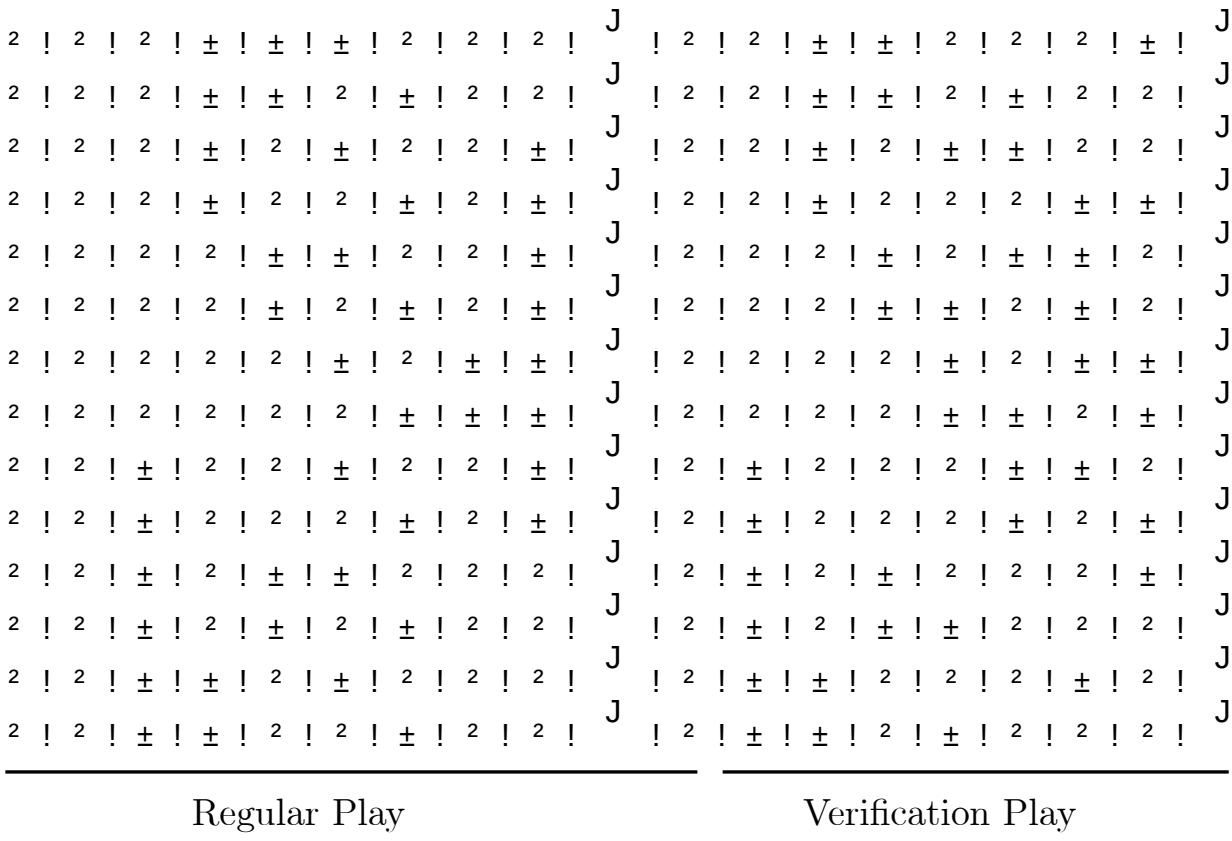


Figure 2.

In second place, we design the transition function[12] when player 2 is sending the last part of the seed but the last stage, i.e., for $t : 4k - 2 > t > 2k - 1$. Here, the randomness of the jumps allow player 1 to hide the reused states.

As we noted above, player 1's automaton is a matrix with $l$ columns and $2^k(\frac{\acute{A}(2^k - 1)}{k})$ rows, i.e., $2^k(\frac{2^k - 1}{k \log k})$ rows. The communication phase starts in the $p$ column that player 1 has chosen randomly. Hence, the states used to process the seed are located in a submatrix with $4k - 2$ rows and a number of columns which depends on $n$.

Now, the associated transition function for $t : 4k - 3 > t > 2k - 1$ is defined by:

$$g^1((^2; j); 0) = (^2; j + n) \text{ if } p + 2k - 1 < j < p + 4k - 2 \text{ and } ^2_j = 0$$

[12]Notice that we do not use a distribution over transition functions, but we produce enough uncertainty on the ¯nal states of the transition function to deter deviations.

Finally, the last state in the communication phase is not in the same column for every row. It depends on $\sigma; n; p$, i.e. on where the communication starts, on the distribution of ones in $\sigma$ and on the number of jumps.

This state is defined as follows for every message:

Let $\varphi$ be a function

$$\varphi : Q \to [p + k; p + 6k - 3]$$
$$\sigma \to \varphi(\sigma) = p + 2k - 1 + (2k - 1 - \sum_{i=1}^{2k-1} \sigma_i) + n \sum_{i=1}^{2k-1} \sigma_i$$

Let $\varphi$ be the $\max \varphi(\sigma)$ where $\sigma \in Q$:

Notice that $p$ will be chosen[13] such that $[p; p + \varphi] \cap \{1; l - 2^k\} \neq \emptyset$:

The initial state is $(1; p)$ where 1 means the signal whose first $2k - 1$ components are ones and $p$ is a random value such that $[p; p + \varphi] \cap \{1; l - 2^k\} \neq \emptyset$ that player 1 chooses.

Now it is possible to define the transition function for the final state for every row: $g^1((\sigma; \varphi(\sigma)); 0) = (\sigma; 1)$:

This is equivalent to: $g^1((1; p); \sigma) = (\sigma; 1)$:

In all other cases the value of $g^1$ equals $\Phi$.

Figure 3 illustrates[14] the scheme that player 1 follows to process the message sent by player 2. This is an example for $k = 3$; $p = 5$ and $n = 2$: The star ($?$) represents the initial state of the automaton of player 1.

[13] Notice that this set is not empty, since $T - L\frac{T}{L+\pm} < l - s$:

[14] We consider the primitive polynomial $f(x) = x^4 + x + 1$ on GF(2):

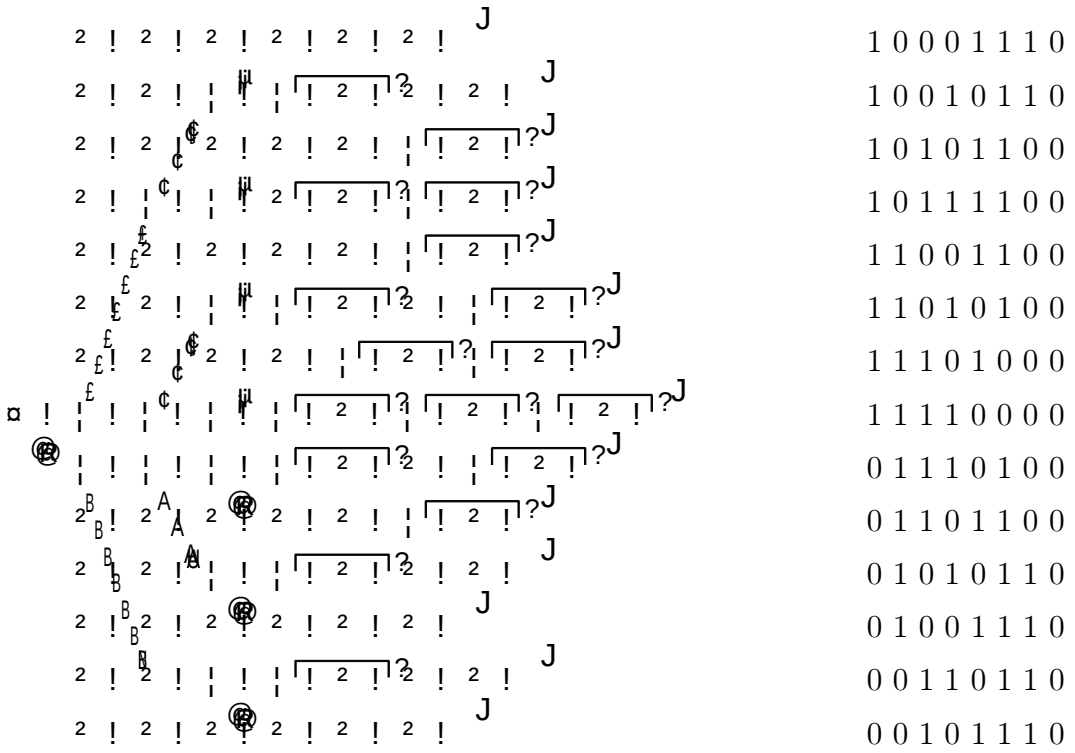


Figure 3: Communication phase.

## 6.8 Equilibrium conditions:

We check here that the constructed strategies are indeed an equilibrium. We show first that any profitable deviation by player 1, cannot be implemented by a finite automata of complexity $m_1$: Then, we study the complexity of a strategy of player 1 which yields a higher payoff when playing against ¿¤, i.e. $comp^1(¾)$ where $r^1_T(¾;¿¤)$ ¸ $P_{t=1}^{T} \frac{r^1(!(²))}{T}$: Secondly, we show that with probability close to 1 there is no profitable deviation by player 2.

Let ¾ be a strategy of player 1 and ² 2 Q, with $r^1_T(¾;¿^²)$ ¸ $P_{t=1}^{T} \frac{r^1(!(²))}{T}$: Then, $!_t(¾;¿^²) = !(²)$ for any $t \cdot \frac{T}{z}$ where z is a fixed number that depends on the action pair (1,1), with payoffs $(3;3)$, and on the other payoffs of the stage game PD (4 and 1 in our model). Note that $z < 1;5$. Therefore, for any strategy ¾ of player 1, $r^1_T(¾;¿^²) \cdot P_{t=1}^{T} \frac{r^1(!(²))}{T} + \frac{C}{T}$ where C ¸ T(4 ¡ 3 + ").

Let $\sigma$ be a pure strategy for player 1 with $r_T^1(\sigma;\cdot^{\alpha}) \geq \sum_{t=1}^{T} \frac{r^1(!(\varepsilon))}{T}$ and such that $\sigma$ is implemented by an automaton of size $m_1$.

In order to characterize the size of the automaton which implements a profitable deviation, consider the following partition of the set of seeds

Let

$$Q(1;\sigma) = \left\{ \varepsilon \in Q \text{ such that } r_T^1(\sigma;\cdot^{\varepsilon}) > \sum_{t=1}^{T} \frac{r^1(!(\varepsilon))}{T} \right\}$$
$$Q(2;\sigma) = \left\{ \varepsilon \in Q \text{ such that } r_T^1(\sigma;\cdot^{\varepsilon}) = \sum_{t=1}^{T} \frac{r^1(!(\varepsilon))}{T} \right\}$$
$$Q(3;\sigma) = \left\{ \varepsilon \in Q \text{ such that } r_T^1(\sigma;\cdot^{\varepsilon}) < \sum_{t=1}^{T} \frac{r^1(!(\varepsilon))}{T} \right\}$$

To study the complexity of $\sigma$ we must know that of $!(\varepsilon)$ for every $\varepsilon \in Q$, hence we analyze the complexity for every set of the partition of $Q$. Define $Q_1 = \{!(\sigma;\cdot^{\varepsilon}) : \varepsilon \in Q(1;\sigma)\}$; $Q_2 = \{!(\sigma;\cdot^{\varepsilon}) : \varepsilon \in Q(2;\sigma)\}$; and $Q_3 = \{!(\sigma;\cdot^{\varepsilon}) : \varepsilon \in Q(3;\sigma)\}$: Notice that $comp^1(Q_2) \geq |Q(2;\sigma)|$ by lemma 7.

As $\sigma$ verifies that $r_T^1(\sigma;\cdot^{\alpha}) \geq \sum_{t=1}^{T} \frac{r^1(!(\varepsilon))}{T}$ then $r_T^1(\sigma;\cdot^{\alpha}) \geq \frac{1}{|Q|} \sum_{\varepsilon \in Q} \sum_{t=1}^{T} \frac{r^1(!(\varepsilon))}{T}$: Hence,

$$r_T^1(\sigma;\cdot^{\alpha}) = \sum_{\varepsilon \in Q} \frac{1}{|Q|} r_T^1(\sigma;\cdot^{\varepsilon}) =$$
$$\frac{1}{|Q|}\left[\Sigma_{\varepsilon \in Q(1;\sigma)} r_T^1(\sigma;\cdot^{\varepsilon}) + \Sigma_{\varepsilon \in Q(2;\sigma)} r_T^1(\sigma;\cdot^{\varepsilon}) + \Sigma_{\varepsilon \in Q(3;\sigma)} r_T^1(\sigma;\cdot^{\varepsilon})\right]$$

Now, since any strategy $\sigma$ of player 1, $r_T^1(\sigma;\cdot^{\varepsilon}) \cdot \sum_{t=1}^{T} \frac{r^1(!(\varepsilon))}{T} + \frac{C}{T}$ and by the definition of $Q(3;\sigma)$; then

$$|Q(1;\sigma)| \left[ \sum_{t=1}^{T} \frac{r^1(!(\varepsilon))}{T} + \frac{C}{T} \right] + |Q(3;\sigma)| \left[ \sum_{t=1}^{T} \frac{r^1(!(\varepsilon))}{T} \right] \geq$$
$$|Q(1;\sigma) + Q(3;\sigma)| \sum_{t=1}^{T} \frac{r^1(!(\varepsilon))}{T}$$

Thus $\frac{C}{T}|Q(1;\sigma)| \geq |Q(3;\sigma)|$ and for $T$ large enough $|Q(1;\sigma)| \geq 2|Q(3;\sigma)|$

In the next lemma we study the least complexity of a strategy of player 1 which can give him more that $\sum_{t=1}^{T} \frac{r^1(!_t(\varepsilon))}{T}$.

**Lemma 8** *The complexity of $\sigma$ such that $r_T^1(\sigma;\cdot^{\alpha}) \geq \sum_{t=1}^{T} \frac{r^1(!(\varepsilon))}{T}$ is*
$comp^1(\sigma) \geq 3|Q(1;\sigma)| + |Q(2;\sigma)|$

*Proof:*

By the definition of complexity, $comp^1(\sigma) = comp^1\{!(\sigma;\cdot^{\varepsilon}) : \varepsilon \in Q\} \geq$
$comp^1\{!(\sigma;\cdot^{\varepsilon}) : \varepsilon \in Q(1;\sigma) \cup Q(2;\sigma)\} =$

$\mathrm{comp}^1(Q_1) + \mathrm{comp}^1(Q_2)$:

Notice that $\mathrm{comp}^1(Q_2) \geq l\,|Q(2;\sigma)|$ by lemma 7. Let us bound the complexity of $Q_1$:

By the definition of $Q(1;\sigma)$, for every $\tau \in Q(1;\sigma)$, $r^1_T(\sigma;\tau) > R^1(!(\tau))$: Therefore there exists a deviation from the proposed play at the end of the game i.e., for every $t \leq 2k+4l$; $!_t(\sigma;\tau') = !_t(\tau)$: Now by lemma 4, a deviation after $4k-2+4l$ stages entails a complexity of player 1 of at least $3l$. By the definition of complexity with finite automata it is sufficient to prove that for every pair $(\tau;t);(\tau';t')$ with $(\tau;t) \neq (\tau';t')$ and $t \geq t'$ in

$$Q(1;\sigma) \times \{4k-2+1;...;4k-2+3l\}$$

there exists $s < T-t$ such that

$(!^2_t(\tau);...;!^2_{t+s}(\tau)) = (!^2_{t'}(\tau');...;!^2_{t'+s}(\tau'))$

and

$\sigma(!_1(\tau);...;!_{t+s}(\tau)) \neq \sigma(!_1(\tau');...;!_{t'+s}(\tau'))$

We just consider the case where $t \neq t'$. (The other case satisfies the condition of lemma 7).

We suppose now that $t > t'$; $t = t' \bmod (l)$ and $\tau \in Q(1;\sigma)$:

Let $s$ be the largest positive integer such that

$!_t(\tau);...;!_{t+s}(\tau) = !_{t'}(\sigma;\tau');...;!_{t'+s}(\sigma;\tau')$:

As $r^1_T(\sigma;\tau) \geq \sum_{t=1}^{T} \frac{r^1(!(\tau))}{T} \geq s < T-t$ and

$\sigma(!_t(\tau);...;!_{t+s}(\tau)) \neq \sigma(!_{t'}(\tau');...;!_{t'+s}(\tau'))$: ¤

Lemma 9 below asserts that for any pure strategy of player 1 that play against the mixed strategy $\tau$¤ of player 2, the payoff is the average on $Q$.

**Lemma 9** For any strategy $\sigma \in \Sigma^1(m_1)$

$r^1_T(\sigma;\tau\text{¤}) \leq \frac{1}{|Q|}\sum_{Q} R^1(!(\tau))$

**Proof:**

Suppose that $r^1_T(\sigma;\tau\text{¤}) \geq \sum_{t=1}^{T} \frac{r^1(!(\tau))}{T}$:

Consider the partition of $Q = Q(1;\sigma) \cup Q(2;\sigma) \cup Q(3;\sigma)$:

First, if $|Q(3;\sigma)| = \alpha$ then $|Q| = |Q(1;\sigma)| + |Q(2;\sigma)|$: By the above lemma the complexity of $\sigma$ is greater than or equal to $3l\,|Q(1;\sigma)| + l\,|Q(2;\sigma)|$:

As $m_1 \geq 3|jQ(1;¾)j + |jQ(2;¾)j = |jQj + 2|jQ(1;¾)j$ and since $2^{k-1} \cdot \frac{m_1-|}{|} < 2^k$; then

$m_1 \geq m_1 - 2| + 2|jQ(1;¾)j \;,\; jQ(1;¾)j = ;$

We conclude that $r^1_T(¾;¿^¤) \cdot \sum_{t=1}^{T} \frac{r^1(!(²))}{T}$:

Next, if $jQ(3;¾)j \neq ;$; as already noted, we can suppose that for $T$ large enough $jQ(1;¾)j \geq 2\,jQ(3;¾)j$. Then,

$m_1 \geq 3|jQ(1;¾)j + |jQ(2;¾)j =$
$= \frac{|}{2}jQ(1;¾)j + \frac{3|}{2}jQ(1;¾)j + |jQ(2;¾)j >$
$> |jQj + \frac{5}{2}|jQ(1;¾)j > m_1$: By the same argument, we get a contradiction. ¤

To conclude that $(¾^¤;¿^¤)$ is equilibrium, the next lemma states the equilibrium condition for player 2. It is enough to prove the condition with one strategy in the support of the mixed strategy of player 2.

**Lemma 10** For any strategy $¿ \in \Sigma^2$ and every $² \in Q$
$\sum_{t=1}^{T} r^2(¾^¤;¿) \cdot \sum_{t=1}^{T} r^2(¾^¤;¿^²)$:

*Proof:*

Let $¿$ be a pure strategy of player 2 such that for some $² \in Q$, $!_t(¾^¤;¿) = !_t(²)$ for every $1 \cdot t \cdot 4k - 2$ and $r^2(¾^¤;¿) \geq r^2(¾;¿^²)$.

Let $s'$ be the smallest integer that $2^k < s' \cdot T$ with $!_s(¾^¤;¿) \neq !_{s'}(²)$ and $!_t(¾^¤;¿) = !_t(²)$ $1 < t < s'$.

Let $\overline{v} = \max\{v(²) \text{ such that } ² \in Q\}$

Observe that $T - 4k - 2 - L|$ is of the order $\pm|$ and for sufficiently large values of $T$; these stages $!_t(²) = (1;1)$ for $T > t > p + \overline{v} + 4k - 2 + L|$. Hence, any strategy $¾$ in the support of $¾^¤$does not tolerate any deviation from the proposed play for $T > t > p + \overline{v} + 4k - 2 + L|$ and $r^2(1;b^2) \geq r^2(1;1) > u^2(P\,D)$.

Therefore, if $T > s > 4k - 2 + L| + p + \overline{v}$ then the strategy $¾^¤$ does not tolerate any deviation. Hence, $r^2(¾^¤;¿) < r^2(¾^¤;¿^²)$, and if $s = T$ then $r^2(¾^¤;¿) < r^2(¾^¤;¿^²)$.

If $2^k < s < 2^k + L| + p + \overline{v}$ then with probability close to one the strategy of player 1, $¾^¤$; detects the deviation of player 2 in future stages and player 1 punishes forever. Notice that player 2 gains more if he deviates at the end of the game. The other deviations are punishable with probability close to one:

If player 2 deviates in the communication phase, i.e.: if $(!_1^2(¾¤;¿);:::;!_{2^k}^2(¾¤;¿))$ is not in $Q$, then with probability at least $\frac{1}{2}$ player 1 will detect the deviation before $l$ stages. Player 2 loses $\frac{T¡l}{T}(x^2 ¡ u^2)$ against what he may win, $\frac{2^k¡1}{T}$.

Therefore $¿^¤$ is a best reply against $¾¤$. ¤

# 7 Concluding Remarks

The design of optimal communication schemes is important when communication is inter-play and games are repeated a finite number of times. When players implement their strategies by means of finite automata these communication schemes may also determine the number of possible plays to achieve a cooperative outcome.

We have presented an equilibrium construction with an optimal communication scheme. Our equilibrium condition in terms of the upper bound of the smallest automaton implementing the cooperative outcome and which depends on both the $"_{¡}$ approximation to the efficient outcome and the number $T$ of repetitions, lies between that of PY (1994) and the one of N (1998). Namely, our upper bound includes that of PY (1994), but, in turn, it is included in the one of N (1998). The reason behind this last relationship between Neyman's upper bound and ours is that the Prisoner's Dilemma game does not belong to the class of games where the communication phase dictates the $"_{¡}$ approximation to the efficient outcome.

In the class of games where this distortion rate is only generated by the communication process, the optimality of such a process is vital for the equilibrium conditions. For instance, and without loss of generality, suppose a 2 player game where the targeted cooperative payoff belongs to the convex hull of the two actions $a_1^1$ and $a_2^1$ of player 1 and $a_1^2$ and $a_2^2$, of player 2, i.e., $x = ,_1 r(a_1^1; a_1^2) + ,_2(a_2^1; a_2^2)$, with $,_i > 0;\ i = 1; 2$ and $\sum_{i=1;2} ,_i = 1$: Label $a_1^1$ and $a_1^2$ by 0 and $a_2^1$ and $a_2^2$ by 1, then $x = ,_1 r(0; 0) + ,_2(1; 1)$: Following our approach, and for $T$ large enough, the equilibrium play consists of a communication phase, which is composed of the actions pairs $(0; 0)$ and $(0; 1)$, and a cycle play, where the action pairs $(0; 0)$ and $(1; 1)$ are played in both the regular and the verification plays. In this case, the "-approximation to $x$ only depends on the $(2k ¡ 1)$ stages of the communication phase where the action pair $(0; 1)$ is played.

Known results in the automata complexity's literature (see Neyman, 1998) state that the upper bounds of the automata have to be subexponential to achieve cooperation in finite repetition of the underlying game. In the above problem, the upper bound of player 1 is lower than $\exp(\frac{"T}{c})$, where $c$ is a parameter which depends on the specific game's payoffs. Our bound is the largest bound to achieve cooperation in a finitely repeated game played by finite automata and it improves all the other bounds already offered in the field. This improvement is due to our optimal codification of the communication phase.

# 8 REFERENCES


Gossner, O. (2000): "Sharing a long secret in a few public words", Working Paper 2000-15, THEMA, UMER CNRS.

Hardly, G.H. and Wright, E.M (1980).: An Introduction to the Theory of Numbers., Oxford University Press, Oxford.

Kalai, E. (1990): "Bounded rationality and strategic complexity in repeated games",.T. Ichiishi, A. Neyman, Y. Tauman, eds., Game Theory and Applications, Academic Press, New York, 131-157.

Kalai, E, and Standford,W (1988): "Finite Rationality and Interpersonal Complexity in Repeated Games", Econometrica 56, 2, 397-410.

Nidl and Niederreiter.(1983): Finite Fields. Encyclopedia of Mathematics and its Applications, Gian-Carlo Rota, editor. Cambridge. Massachusetts.

Neyman, A. (1985): "Bounded complexity justifies cooperation in the finitely repeated prisoner's dilemma", Economics Letters 19, 227-229.

——— (1997): "Cooperation, repetition, and automata", S. Hart, A. Mas Colell, eds., Cooperation: Game-Theoretic Approaches, NATO ASI Series F, 155. Springer-Verlag, 233-255.

——— (1998): "Finitely Repeated Games with Finite Automata", Mathematics of Operations Research 23, 3, 513-552.

Rubinstein,A (1986): "Finite Automata play the repeated prisoners' dilemma", Journal of Economic Theory 39, 83-96.

Shannon, C.E. (1948): "A Mathematical Theory of Communication", Bell System Tech. J. 27, 379-423 and 623-656.

Zemel, E. (1989): "Small talk and cooperation: A note on bounded rationality", Journal of Economic Theory 49, 1, 1-9.