A Model of Persuasion with Boundedly Rational Agents

Jacob Glazer
Tel Aviv University and
Boston University

and

Ariel Rubinstein
Tel Aviv University and
New York University

Abstract

A new model of mechanism design with a boundedly rational agent is studied. A speaker presents a request to a listener who would like to accept the request only if certain conditions are met by the speaker's true profile. This persuasion situation is modeled as a leader-follower relationship. The listener first announces and commits to a persuasion rule, i.e., a set of conditions to be satisfied by the profile in order for him to be persuaded. Then, the speaker presents a profile, though not necessarily the true one. The speaker is boundedly rational in the sense that his ability to come up with a persuasive profile is limited and depends on the true profile and on the persuasion rule and the way in which it is framed. We fully characterize the circumstances under which the listener's goal can be achieved.

The second author acknowledges financial support from ERC grant 269143.
We wish to thank Noga Alon, Ayala Arad, Sambuddha Ghosh, Bart Lipman, Michael Richter, Rani Spiegler and Jaber Zarezadeh. Our special thanks to Chuck Wilson for his very useful comments and suggestions, especially regarding Proposition 7.

1. Introduction

"I went to a bar and was told it was full. I asked the bar hostess by what time one should arrive in order to get in. She said by 12 PM and that once the bar is full you can only get in if you are meeting a friend who is already inside. So I lied and said that my friend was already inside. Without having been told, I would not have known which of the possible lies to tell in order to get in." (M.R. describing an actual experience at a Tel Aviv bar.)

In this episode, M.R. was trying to persuade the bar's hostess to let him in. The hostess revealed the conditions for her to be persuaded though she had no way of verifying whether M.R. satisfies those conditions. Thus, her statement also guided M.R. how to lie effectively in order to gain entrance to the bar.

Consider another example: A search committee would like to identify those candidates who exhibit consistency in their preferences, in the sense that when asked to choose between plans of action their preferences satisfy transitivity. The committee members view a consistency of this form to be a desirable attribute for the job. Therefore, the candidates are given the following test: An hypothetical scenario is described to them which involves three possible plans of actions, denoted as a, b and c. Each candidate is then asked to answer three questions of the form "Which plan do you prefer, x or y?" The candidate responds to each question by saying either "I prefer x to y" (denoted as $x \succ y$) or "I prefer y to x" (denoted as $y \succ x$). Assume that the committee is required to inform the candidates of the conditions that their answers must fulfill in order to pass the test. Suppose that the committee announces the following set of conditions (hereafter referred to as a *codex*):

R1: If $a \succ b$ and $b \succ c$, then $a \succ c$.

R2: If $b \succ a$ and $c \succ b$, then $c \succ a$.

R3: If $a \succ b$ and $a \succ c$, then $c \succ b$.

R4: If $c \succ a$ and $c \succ b$, then $a \succ b$.

Notice that the codex is satisfied only by the two (transitive) orderings in which b is positioned in the last place.

If a candidate is fully rational he can come up with answers to the three questions that satisfy all four conditions and thus pass the test, regardless of what his true preferences are. However, this is no longer the case if the candidate's ability to come up with a set of answers

that satisfies the codex is limited and depends on the individual's true preferences, which is the assumption of our analysis.

Consider three candidates named Alice, Bob and Carol, who are all eager to get the job and are willing to lie about their preferences in order to succeed.

Alice holds the ordering $a \succ c \succ b$ and thus satisfies all four conditions. She can pass the test by simply telling the truth.

Bob holds the ordering $a \succ b \succ c$. His ordering does not satisfy the codex since it satisfies the antecedent of R3 but violates R3's consequent. Bob can pass the test by telling the truth about his preferences between *a* and *b* and between *a* and *c* (thus satisfying the antecedent of R3) and lying about his preferences between *b* and *c* (such that the consequent of R3 is also satisfied). In other words, R3 not only informs Bob that his true preferences will be rejected but also *guides* him in how to lie in order to pass the test (i.e., by declaring the ordering $a \succ c \succ b$).

Carol holds the cyclical preferences $a \succ b \succ c \succ a$. The only antecedent she satisfies is that of R1; however, she violates R1's consequent. If she uses R1 as a guide in formulating her answers she will declare the ordering $a \succ b \succ c$ and will fail the test.

In short, all Alice has to do in order to pass the test is tell the truth. Bob and Carol, on the other hand, will fail if they tell the truth. According to our main assumption and given the codex described above, Bob can lie successfully but Carol cannot. The codex guides Bob, who holds an ordering in which *b* is in the middle, to switch the positions of *b* and *c* and, thus satisfy the codex. Carol, whose preferences are cyclical, is not guided to an ordering in which *b* is in last place. Our assumption, presented formally in the next section, is that when faced with such a codex, individuals are able to come up with successful answers if and only if either their true preferences satisfy the codex (as in the case of Alice) or they are guided by the codex to a set of answers that satisfy all the conditions (as in the case of Bob). Under this assumption, only individuals with transitive preferences will be able to pass the test, either by telling the truth or by lying successfully.

The above two scenarios are examples of persuasion situations. A persuasion situation involves a speaker and a listener. The speaker attempts to persuade the listener to take a certain action or to adopt a certain position. The interests of the two parties are not necessarily identical and depend on the speaker's "profile", i.e., a set of relevant non-verfiable attributes

(or facts) known only to the speaker. The speaker would like the listener to choose his desired action regardless of his true profile, whereas the listener wishes to be persuaded only if the speaker's profile satisfies certain conditions (i.e., belongs to a certain set). In his attempt to persuade the listener, the speaker presents a "profile", which is not necessarily the true one. However, cheating effectively (i.e., presenting a persuasive false profile) may be difficult, since it requires the speaker to invent a fictitious profile. The listener is aware of the fact that the speaker may be providing false information that is not verifiable. He is also aware of the procedure used by the speaker to come up with a persuasive false profile.

We model a persuasion situation as a leader-follower relationship. First, the listener (leader) announces and commits to a persuasion rule (a codex), i.e., a set of conditions that the profile presented by the speaker must satisfy in order for the listener to be persuaded. Then, the speaker (follower) chooses a profile to present. In order to persuade the listener, the speaker can present a false profile and this is where bounded rationality is introduced. We assume that the speaker's ability to come up with a persuasive profile is limited and depends on his true profile, the content of the persuasion rule and the way in which the rule is framed.

Modeling the idea that the speaker's ability to cheat is limited, could have been carried out in a framework similar to that of Green and Laffont (1986) (which was also the approach taken in Glazer and Rubinstein (2004, 2006)). In this type of models, the set of messages that the speaker had to choose from, is exogenously given and dependent on the speaker's profile. The novelty of the current paper lies in the assumption that not is cheating difficult, but also the speaker's ability to cheat effectively depends on the way in which the persuasion rule is framed. In such a case, the desirable persuasion rule should be complex enough that a speaker whose profile should not be persuasive will not be able to persuade the listener by manipulating the information but, at the same time, should be simple enough that a speaker whose profile should be persuasive will indeed be able to persuade the listener.

We will now introduce the model while emphasizing our new approach to modeling bounded rationality. We will then explore two notions of implementation. We will characterize the circumstances under which the listener's goal is implementable, in the sense that there exists a codex that enables the speaker to persuade the listener (either by telling the truth or by cheating) if and only if the speaker's true profile should be persuasive. We will also characterize conditions for truthful implementation under which the listener's goal is

implementable and any speaker who is able to persuade the listener can do so without lying.

2. The Model

The set of profiles

Let V be a set of $K \geq 2$ propositional variables denoted by $v_1,..,v_K$. Each variable can take one of two truth values: "*T*rue" or "*F*alse". A *profile* is a truth assignment for each of the variables. Denote by $s(v)$ the truth value of the variable v in the profile s. We will sometimes present a profile s as a K-vector $(s_1,..,s_K)$ of 0's and 1's, where $s_k = 1$ means that $s(v_k) = T$ and $s_k = 0$ means that $s(v_k) = F$.

Let S be the set of all profiles. We assume that all 2^K profiles are logically possible, namely that the content of the variables is such that the truth combination of some of the variables does not exclude the truth combination of any of the others (as would have been the case, for example, if v_1 was "being a female" and v_2 was "being a male").

The speaker and the listener

There are two agents: a speaker and a listener. The speaker knows which profile is true whereas the listener knows only the set S. The speaker wishes to persuade the listener to accept a particular request regardless of the true profile. The listener can either accept or reject the request. He would like to accept the speaker's request only if the profile belongs to a given set A. Let $R = S - A$ be the set of profiles for which the listener would like to reject the speaker's request.

We analyze the following leader-follower scenario: First, the listener announces and commits to a codex, which is a set of conditions that the profile presented by the speaker must satisfy in order for the speaker's request to be accepted. Then, the speaker (who knows the true profile) announces a profile which may or may not be the true one. The listener is committed to applying the codex to the profile announced by the speaker.

The codex

A *codex* is defined as a set of propositions in propositional logic that uses only the variables in the set V. A proposition in the codex is referred to as a *rule*. Only a profile that does not violate any of the propositions will "persuade" the listener. We impose two restrictions on a codex:

1) Structure: Each rule φ in the codex must have the structure $\wedge_{y \in W} \varphi_y \rightarrow \varphi_x$ where W is a

non-empty subset of V, $x \in V - W$ and each φ_v is either v or $-v$ (the negation of v). For example, the proposition $v_4 \wedge -v_1 \rightarrow v_3$ can be a rule in a codex but $v_1 \rightarrow -v_1$ cannot. For any given rule $\varphi = \wedge_{y \in I}\, \varphi_y \rightarrow \varphi_x$, we denote $a(\varphi) = \wedge_{y \in I}\, \varphi_y$ (the antecedent of φ) and $z(\varphi) = \varphi_x$ (the consequent of φ). We interpret a rule as a statement of the following form made by the listener: "If your profile satisfies the antecedent of the rule, then it should also satisfy the consequent."

2) Coherence: The codex cannot contain rules that conflict in the sense that there is no pair of rules such that their antecedents do not conflict and their consequents do (one consequent is v and the other is $-v$ for the same variable v). Formally, a codex is coherent if it does not contain two rules $\varphi = \wedge_{y \in W_1}\, \varphi_y \rightarrow x$ and $\psi = \wedge_{y \in W_2}\, \psi_y \rightarrow -x$ where for any $y \in W_1 \cap W_2$ we have $\varphi_y = \psi_y$. Thus, coherence does not only require that a codex not contain the two rules $v_1 \rightarrow v_2$ and $v_1 \rightarrow -v_2$ but also that it will not contain the two rules $v_1 \rightarrow v_3$ and $v_2 \rightarrow -v_3$ (i.e., the antecedents do not conflict but the consequents do). In our view, a codex containing these two rules is problematic: a speaker whose true profile, s, is such that $s(v_1) = s(v_2) = T$ will rightly complain that the codex imposes two conflicting requirements on him, with regard to the variable v_3.

To illustrate, in the second example that appeared in the introduction, the three variables are $v_1 = a \succ b$, $v_2 = b \succ c$ and $v_3 = c \succ a$, and the proposed codex consists of the following four rules: $v_1 \wedge v_2 \rightarrow -v_3$, $-v_1 \wedge -v_2 \rightarrow v_3$, $v_1 \wedge -v_3 \rightarrow -v_2$, and $v_3 \wedge -v_2 \rightarrow v_1$.

Given a codex Λ, let $T(\Lambda)$ be the set of profiles that satisfy all propositions in Λ. In other words, $T(\Lambda)$ is the set of profiles which if announced by the speaker, will persuade the listener. More precisely, using the notation $s \models \psi$ for "proposition ψ is true in profile s", $T(\Lambda) = \{s |\ s \models \varphi \text{ for all } \varphi \in \Lambda\}$.

(Recall that $s \models \wedge_{y \in I}\, \psi_y \rightarrow \psi_x$ unless:

(i) the antecedent of ψ is satisfied, i.e., for all $y \in I$ we have $s(y) = T$ if $\psi_y = y$ and $s(y) = F$ if $\psi_y = -y$; and

(ii) the consequent of ψ is violated, i.e., either $s(x) = T$ and $\psi_x = -x$ or $s(x) = F$ and $\psi_x = x$.)

The Speaker's Choice Procedure

The speaker can either state the true profile or make up a false one. A fully rational speaker can come up with a profile that satisfies the codex regardless of what the true profile is. We assume, however, that the speaker is boundedly rational in the sense that he is limited in his ability to come up with a persuasive false profile. Essentially we assume that the speaker applies the following procedure (a formal discussion will follow):

Step 1. Determine whether your true profile satisfies the codex.

If it does, then announce the true profile.

If it does not, then go to Step 2

Step 2. Find a rule (not considered in a previous round of Step 2) that is violated by your true profile (i.e., your true profile satisfies the rule's antecedent but violates its consequent). Change the truth value of the variable that appears in the consequent of this rule and determine whether the modified profile satisfies the codex.

If it does, announce the new profile.

If it does not, iterate Step 2.

Step 3. If you are unable to come up with a modified profile that satisfies the codex in Step 2, announce your true profile.

Guidance

We say that, given Λ**, the speaker is guided to** s' **from** s (denoted as $s \to_\Lambda s'$) if for every variable v for which $s'(v) \neq s(v)$, there is a rule $\varphi \in \Lambda$ such that:

(1) $s \models a(\varphi)$ and $s' \models a(\varphi)$; and

(2) $s' \models z(\varphi)$ (that is, if $z(\varphi) = v$ and $s'(v) = T$ and if $z(\varphi) = -v$, then $s'(v) = F$).

In other words, the speaker is guided from s to s' if any switch from $s(v)$ to $s'(v)$ is triggered by a rule that requires that the value of the variable v will be $s'(v)$ and its antecedent is satisfied at s and refers only to the variables that are kept unchanged. We refer to the relation $\to_\Lambda$ as the guidance relation induced by Λ.

The speaker may be guided from one profile to several others. For example, suppose that $K = 4$ and Λ contains the three rules $v_1 \to -v_3$, $v_2 \to -v_4$ and $v_2 \wedge v_3 \wedge v_4 \to -v_1$. Then, the speaker is guided by Λ from $(1,1,1,1)$ to each of the profiles $(1,1,1,1)$, $(1,1,0,1)$, $(1,1,1,0)$,

$(1,1,0,0)$ and $(0,1,1,1)$.

Persuasion

Given a codex Λ, we say that the speaker whose profile is s can *persuade* the listener if $s \rightarrow_\Lambda s'$ for some $s' \in T(\Lambda)$. Define $P(\Lambda) = \{s |\ s \rightarrow_\Lambda s' \text{ for some } s' \in T(\Lambda)\}$. That is, $P(\Lambda)$ is the set of profiles for which the speaker can persuade the listener. Note that it is possible for the speaker to be guided from the true profile to profiles that are persuasive and others that are not. By our definition, the speaker is able to persuade the listener if he is guided to at least one persuasive profile. Note also that we do not allow the speaker to be guided sequentially, i.e., first from s to s' and then from s' to s''. Later on, we will comment on these two assumptions.

Implementation

The set A is implementable if there is a codex Λ such that $A = P(\Lambda)$.

The set A is truthfully implementable if there is a codex Λ such that $P(\Lambda) = T(\Lambda) = A$.

Thus, if a codex implements A then the speaker is able to persuade the listener in all profiles for which the listener should be persuaded and in none of the profiles for which he should not. However, in some of the cases in which the listener should be persuaded, the speaker has to "alter the truth" in order to persuade the listener. If a codex truthfully implements A, then a speaker whose profile should persuade the listener is able to do so by simply telling the truth.

Note that the "revelation principle" does not hold in our framework and, as we will see later, there are cases in which the set A is implementable but not *truthfully implementable*.

Comment: The following analogy may help clarify our concept of implementation. Suppose that you manage a large network of agents around the globe. The location of each agent is characterized by two coordinates. Suppose that you want to award a prize only to those agents whose locations are in the set A. You don't know who is located where but you do know that all agents use the same program to solve systems of equations. Whether the program will converge to a solution depends on the system and the initial conditions inserted into the program. You also know that people tend to input their true coordinates as the initial conditions. In such a case, you can try to come up with a system of equations such that the

program will converge to a solution within a specified time if and only if it starts from a point in the set A. If you can find such a system of equations it will serve as a mechanism for selecting the agents that you want to award. Note that a rule in our model is actually an equation where the propositional variables are the unknowns while a codex is in fact a system of equations.

3. Examples

Example 1: Assume that there are three "scenarios", numbered 1,2 and 3 and an individual's attitude towards each one can be either "positive" or "negative". A principal would like to identify those individuals who are consistent in their attitude towards the three scenarios, i.e., who have the same attitude towards all three. In order to do so, the principal performs the following test: each individual is asked to state his attitude (positive or negative) to each of the three scenarios. Let the variable v_i stand for: "the individual's attitude to scenario i is positive" and therefore, $A = \{(1,1,1),\ (0,0,0)\}$.

Consider the following three codexes:

Λ_1: "The second and third answers should be the same as the first" ($\Lambda_1 = \{v_1 \to v_2,\ -v_1 \to -v_2,\ v_1 \to v_3,\ -v_1 \to -v_3\}$)

In this case, $T(\Lambda_1) = A$ and $P(\Lambda_1) = S$ since for any profile (s_1, s_2, s_3) we have $(s_1, s_2, s_3) \to_{\Lambda_1} (s_1, s_1, s_1) \in T(\Lambda_1)$.

Λ_2: "The second answer should be the same as the first and the third answer should be the same as the second."

($\Lambda_2 = \{v_1 \to v_2,\ -v_1 \to -v_2,\ v_2 \to v_3, -v_2 \to -v_3\}$)

In this case, $T(\Lambda_2) = A$ but $P(\Lambda_2) = S - \{(1,0,0),(0,1,1)\}$ (since $(1,0,0)$ is guided only to $(1,1,0)$).

Λ_3: The three scenarios are ordered clockwise. For every scenario i the codex requires that if the answer regarding scenario $i+1$ (which follows scenario i) is different from the answer regarding scenario $i+2$ (which follows $i+1$), then the answer regarding scenario i should coincide with the answer regarding scenario $i+2$.

(Λ_3 contains the three rules $v_i \wedge v_{i+1} \to v_{i+2}$ $(\forall i)$ and the three rules $v_i \wedge -v_{i+1} \to -v_{i+2}$ $(\forall i)$.)

Λ_3 truthfully implements A, since $P(\Lambda_3) = T(\Lambda_3) = A$.

Thus, although the three codexes are satisfied by the same set of profiles, only the third codex implements the principal's goal.

Example 2: A principal would like to select "decisive" individuals (regardless of the opinions they hold) for a particular task. In order to do so he presents the candidates with a dilemma and three possible exclusive solutions (denoted by 1, 2 and 3). He then asks each candidate whether each of the three possible solutions is appropriate. The principal wishes to identify those individuals who view exactly one solution to be appropriate (regardless of which one it is). Let v_i stand for "solution i is appropriate" and therefore $A = \{(1,0,0),\ (0,1,0),\ (0,0,1)\}$.

We will show that A is not implementable. Assume that Λ implements A.

Case (1): $T(\Lambda) = A$. The profile $(0,0,0)$ is not in $T(\Lambda)$ and hence there is a rule in Λ that this profile violates; w.l.o.g. that rule is either $-v_1 \to v_3$ or $-v_1 \wedge -v_2 \to v_3$. If both case $(0,0,0) \to_\Lambda (0,0,1)$ and hence $(0,0,0) \in P(\Lambda)$ although $(0,0,0) \notin A$, a contradiction.

Case (2): One of the profiles in A, w.l.o.g. $(0,0,1)$, is not in $T(\Lambda)$. Then, there must be another profile in A, w.l.o.g. $(0,1,0)$, such that $(0,0,1) \to_\Lambda (0,1,0)$. This requires that $-v_1 \to v_2$ be in the codex. However, in that case, $(0,0,0) \to_\Lambda (0,1,0) \in T(\Lambda)$ and therefore $(0,0,0) \in P(\Lambda)$ although $(0,0,0) \notin A$, a contradiction.

Note that even though the above set is not implementable its complement is. Let $A' = S - A$. Consider the codex Λ' that consists of the $K(K-2)$ rules $v_i \to v_j$ where $j \neq i+1$ ($K+1$ is taken to be 1). Obviously, $T(\Lambda') = \{all\ F, all\ T\}$. The codex guides the speaker to "*all T*" from every profile in R' except for "*all F*". For any $s \in R'$ where there is a unique v_i for which $s(v_i) = T$, the speaker is guided from s only to profiles for which v_{i+1} receives the value F and hence violates the codex. Thus, $s \notin P(\Lambda')$.

Example 3: A certain individual (the listener) holds a positive opinion on K issues. He would like to find out whether another individual (the speaker) shares his opinion on at least m of those issues, where $0 < m < K$. Let $A_m = \{s|\ s$ receives the value T for at least m variables$\}$ where $0 < m < K$. We will show that A_m is implementable.

Let Λ be the codex that consists of all rules $R(y, W)$ (where y is a variable and W is a set of at most m variables which does not contain y), which states that if the variables in W receive

the value T and the variables in $V - W - \{y\}$ receive the value F then y should also get the value T. (Formally, $R(y, W) = [\wedge_{v \in W} v] \wedge [\wedge_{v \in X-W-\{y\}} -v] \rightarrow y$.) Obviously, $T(\Lambda) = A_{m+1}$ and $P(\Lambda) = A_m$. Thus, the speaker whose profile assigns the truth value T to up to m variables is guided to "slightly exaggerate" and to claim that there is one more variable that receives the value T. This codex will not guide speakers whose profiles have less than m true variables to cheat effectively. In this case, the implementation is not truthful, but as will be shown later in Proposition 3, A_m is in fact truthfully implementable for $K > 3$ and $m > 2$.

4. Auxiliary concepts and results

Before characterizing the implementable sets, we need to introduce some auxiliary concepts and results.

Properties of the relation $\rightarrow_\Lambda$

Lemma 1:

(a) The relation $\rightarrow_\Lambda$ is reflexive and anti-symmetric (i.e., for any two distinct profiles s and s', if $s \rightarrow_\Lambda s'$ then $s' \nrightarrow_\Lambda s$).

(b) If s is opposed to s' ($s(v) \neq s'(v)$ for all v), then $s \nrightarrow_\Lambda s'$.

(c) If $s \rightarrow_\Lambda t$ and s' is between s and t (that is $s(v) \neq s'(v)$ implies that $s'(v) = t(v)$), then $s \rightarrow_\Lambda s'$ and $s' \rightarrow_\Lambda t$.

Proof: Anti-symmetry follows from the assumption that the codex is coherent. The rest of the Lemma follows immediately from the definition of the relation $\rightarrow_\Lambda$. ■

The next lemma shows that the guidance relation $\rightarrow_\Lambda$ fully conveys the information about $T(\Lambda)$, the set of profiles that satisfy the codex Λ. Given a binary relation $\rightarrow$, denote $T(\rightarrow) = \{s \mid \text{for no } t \neq s,\ s \rightarrow t\}$ and $P(\rightarrow) = \{s \mid \text{there is } t \in T(\rightarrow) \text{ such that } s \rightarrow t\}$.

Lemma 2:

(a) $T(\Lambda) = T(\rightarrow_\Lambda)$

(b) $P(\Lambda) = P(\rightarrow_\Lambda)$

Proof: (a) Assume that $s \notin T(\Lambda)$. Then there is a rule $\varphi = \wedge_{y \in I} \varphi_y \rightarrow \varphi_x$ in Λ such that $s \models \varphi$ is not true, i.e., s satisfies the antecedent $\wedge_{y \in I} \varphi_y$ but not the consequent φ_x. Thus,

$s \to_\Lambda s'$ where s' is the profile that differs from s only in the truth value of the variable x, i.e., $s \notin T(\to_\Lambda)$.

In the other direction, assume that $s \notin T(\to_\Lambda)$. Then there is a profile $t \neq s$ such that $s \to_\Lambda t$. Thus, there is a variable x and a rule $\varphi = \wedge_{y \in I} \varphi_y \to \varphi_x$ such that s and t satisfy φ's antecedent, $t(x) \neq s(x)$, and $t \models \varphi$. Hence, s does not satisfy φ and therefore $s \notin T(\Lambda)$.

(b) The proof follows from (a) and the definitions $P(\Lambda) = \{s \mid s \to_\Lambda s' \text{ for some } s' \in T(\Lambda)\}$

and $P(\to_\Lambda) = \{s \mid s \to_\Lambda s' \text{ for some } s' \in T(\to_\Lambda)\}$. ■

The neighborhood relation

A key element in the analysis is the neighborhood binary relation N on the set S. Define sNs' to mean that s and s' differ in the truth value of exactly one variable. The relation N is symmetric and irreflexive. Define a distance function $d(s,s') = |\{v \mid s(v) \neq s'(v)\}|$.

A *path* is a sequence of distinct profiles $(s_1,\ldots,s_L)$ such that $s_1Ns_2N\ldots Ns_L$. If $L > 2$ and s_LNs_1, then the path is a *cycle*. Any cycle must contain an even number of profiles. We say that a cycle is a *counting cycle* (referred to in graph theory as a Hamiltonian Cycle) of the set X if it contains all elements of X. Obviously, S has a counting cycle. A sequence $(s^0, s^1, ,, \ldots, s^L)$ is a *ray* from s^0 if $s^{l+1}Ns^l$ and $d(s^l, s^0) = l$.

Let $N(s)$ be the set of neighbors of s. If sNs' then $N(s) \cap N(s') = \emptyset$. For any two profiles s and s', $|N(s) \cap N(s')|$ is either 0 or 2. In particular, if $rNsNt$ then there is a unique u such that (r,s,t,u) is a cycle. Denote this u by $N(r,s,t)$.

Complete rules

A *complete rule* is a proposition of the type $\wedge_{v \in V-\{x\}} \varphi_v \to \varphi_x$. In other words, its antecedent refers to $K-1$ variables and the consequent to the remaining one. If a codex Λ contains the complete rule $\wedge_{v \in V-\{x\}} \varphi_v \to \varphi_x$, then $s \to_\Lambda s'$ where s and s' are the two neighbors defined by $s \models \wedge_{v \in V-\{x\}} \varphi_v \wedge -\varphi_x$ and $s' \models \wedge_{v \in V-\{x\}} \varphi_v \wedge \varphi_x$.

For any two neighbors s and s', let $\varphi(s,s')$ be the complete rule $\varphi = \wedge_{v \in V-\{x\}} \varphi_v \to \varphi_x$. Thus, $s \to_\Lambda s'$ for any codex Λ that contains φ.

The last Lemma in this section demonstrates that the language we use for codexes does not limit the sets that can be specified, that is, it allows the specification of any subset $X \subseteq S$:

Lemma 3: For every set $X \subseteq S$, there is a codex Λ such that $T(\Lambda) = X$.

Proof: Let $(s^1,,,\ldots,s^L)$ be a counting cycle of S. The set $\Lambda = \{\varphi(s^l, s^{l+1}) \mid s^l \notin X\}$ is coherent and thus Λ is a codex. Obviously, $T(\Lambda) = X$. ■

A *Canonical Codex*

A particular type of codexes, to be termed canonical, will play a central role in our analysis. A codex is *canonical* if:

(i) It consists of complete rules.

(ii) For every s, there is at most one t such that $s \rightarrow_\Lambda t$.

(iii) For every $s \in P(\Lambda) - T(\Lambda)$, there is $r \in S - P(\Lambda)$ such that $r \rightarrow_\Lambda s$.

Thus, a canonical codex that implements the set A is a set of complete rules such that (a) for every profile $r \in R$ the codex contains a unique rule that is violated by r and (b) a profile $s \in A$ violates the codex only if the codex contains a rule that is violated by some $r \in R$ and guides the speaker to s.

A canonical codex is analytically simple, although it does not necessarily have a natural interpretation. If it implements the set A, then the number of rules it contains is at least equal to the number of profiles in R and thus can be very large. A canonical codex makes the speaker's task relatively simple since by (ii) it guides the speaker to at most one alternative profile. Condition (iii) is relevant only in the case of non-truthful implementation and it requires that a profile in A not be rejected by the codex unless the listener uses that particular profile to "deal" with some other profiles in R that the listener would like to block.

5. Truthful Implementation

In this section, we fully characterize the truthfully implementable sets. In particular, we show that when a set A is truthfully implementable, implementation can be achieved by a canonical codex that consists of $|R|$ complete rules, each of which guides a distinct profile s in R to a neighboring profile in R.

Proposition 1: If the set A is truthfully implementable, then it is truthfully implementable by a canonical codex.

Proof: Let Λ be a codex such that $T(\Lambda) = P(\Lambda) = A$.

By Lemma 2, $T(\Lambda) = T(\rightarrow_\Lambda)$ and thus for every $s \in R$ there is a profile $t \neq s$ such that

$s \to_\Lambda t$. Let $n(s)$ be some neighbor of s that is between s and t. By Lemma 1, we have $s \to_\Lambda n(s) \to_\Lambda t$ and therefore $n(s) \notin T(\Lambda)$. The canonical codex $\Lambda' = \{\varphi(s, n(s)) \mid s \in R\}$ truthfully implements A. ■

We say that a set of profiles C is *connected* if for any two profiles $s, s' \in C$ there is a path of elements in C connecting s and s'. The set C is a connected component of R if it is a maximal connected subset of R.

The next proposition states that a set A is truthfully implementable if and only if the set R is a union of connected components, each of which contains a cycle. Truthful implementation is accomplished by means of a codex that traps all "undeserving" speakers (i.e., speakers whose profile should not be accepted) in a "circle of lies." In other words, an undeserving speaker is (mis)guided by the codex to pretend to be a neighboring undeserving speaker whose profile is rejected by the codex and who, in turn, is guided by the codex to pretend to be a third neighboring undeserving speaker whose profile is rejected and so on. Eventually this chain creates a cycle.

Proposition 2: The set A is truthfully implementable if and only if every connected component of R contains a cycle.

Proof: Assume that A is truthfully implementable. By Proposition 1, the set is implementable by a canonical codex Λ. Then, for every $s \in R$ there is a unique profile $n(s) \in R$ such that $sNn(s)$ and $s \to_\Lambda n(s)$. Let s_1 be an arbitrary profile in R. Define $s_{l+1} = n(s_l)$. By the finiteness of R we have $s_L = s_{L'}$ for some $L' < L$. Thus, s_1 is connected in R to a cycle in R.

In the other direction, assume that any connected component of R has a cycle. Define the binary relation $\to$ on R as follows: Let C be a connected component of R. Select a subset of profiles in C that form a cycle $s_1 N s_2 N .., N s_L N s_1$. For any l, add $s_l \to s_{l+1}$ to the relation ($L + 1$ is taken to be 1). For any element $s \in C - \{s_1 \ldots, s_L\}$, choose one of the shortest paths $t_1 N t_2 .., N t_N$ of profiles in C where $t_1 = s$ and t_N is in the cycle and add $t_1 \to t_2$ to the relation. Let $\Lambda = \{\varphi(s, s') \mid s \to s'\}$. Obviously, the relation $\to$ is anti-symmetric and thus Λ is coherent. The relation $\to_\Lambda$ is identical to $\to$ and $P(\Lambda) = T(\Lambda) = A$. ■

The following proposition describes families of sets that are truthfully implementable. The

first family consists of all sets that are "small" in the sense that they contain no more than $K-1$ profiles. Each of the sets in the second family consists of all profiles for which the number of variables that are true exceeds a certain threshold. The sets belonging to the third family have the property that a particular variable is true (or false) for all profiles included in the set. The third family consists of all sets for which there are two variables, such that the inclusion of a profile in the set is independent of their truth values. These two "degenerate" variables are used in the codex merely to "confuse" the undeserving speaker.

Proposition 3: For $K \geq 3$, any set A that satisfies at least one of the following conditions is truthfully implementable:

(1) A is "small" with at most $K-1$ profiles.

(2) The number of true variables must exceed a threshold: there exists a number $m \geq 3$, such that $A = A_m = \{s|$ at least m variables receive the value T at $s\}$.

(3) There is a particular variable whose value must be true (or false): there exists a variable v such that $A \subseteq T(v)$ (or $T(-v)$) where $T(v)$ is the set of all profiles in which v receives the value T.

(4) There are two irrelevant variables v' and v'' such that if $s \in A$, then so is any profile s' for which $s(v) = s'(v)$ for all v other than v' and v''.

Proof: By Proposition 2, it is sufficient to show that every $s \in R$ is connected by a path in R to a cycle in R.

(1) First, we show that the set R is connected. It is well known that for any two profiles s and t in R that are not neighbors, there are K "disjoint" paths connecting s and t. Since A contains at most $K-1$ elements, at least one of the paths contains only elements of R. Thus, R is connected.

Second, we show that R contains a cycle. Otherwise, let $s_1 N s_2 N \ldots N s_L$ be a longest path of distinct elements in R. Since R contains more than half of the profiles, there must be two opposing profiles belonging to R and thus $L \geq K+1 \geq 4$.

Since $s_3 \in N(s_2) \cap N(s_4)$ there is another profile x such that $s_2 N x N s_4$. The profile x must be in A since otherwise (s_2, s_3, s_4, x) forms a cycle in R. The profile x is not a neighbor of s_1 since s_1 is a neighbor of s_2. The set $N(s_1)$ consists of $s_2 \in R$ and $K-1$ other profiles. It is impossible that all of them are in A since x is not one of them. Thus, $N(s_1)$ contains another

element in R (in addition to s_2) and we can extend the path.

(2) R is connected since each profile in R is connected to the "all F" profile. The set R contains the $2K$-element cycle:

$((1,0,\ldots,0),(1,1,0,\ldots,0),(0,1,0,\ldots,0),(0,1,1,0,\ldots,0),\ldots,(0,0,\ldots,1),(1,0,\ldots,0,1))$.

(3) Since $A \subseteq T(v)$ the set $T(-v) \subseteq R$ and it has a counting cycle. Any element in R is either in $T(-v)$ or is a neighbor of a profile in $T(-v)$. Thus, R is connected and contains a cycle.

(4) Any $s \in R$ belongs to a cycle consisting of the four profiles in the set $\{t|\ t(v) = s(v)$ for any $v \notin \{v', v''\}\}$. By assumption these four profiles are in R. ■

An alternative interpretation of truthful implementation: Let $K = 3$ and let $\Lambda = \{v_1 \to v_2,\ v_2 \to v_3\}$. Then, $(1,0,0) \to_\Lambda (1,1,0)$ and $(1,1,0) \to_\Lambda (1,1,1)$. However, by our assumptions, the speaker is not guided iteratively and thus is not guided from $(1,0,0)$ to the persuasive profile $(1,1,1)$. Had we allowed the speaker to be guided iteratively, the following alternative definition of implementation would have applied:

We say that A is *implementable in the alternative sense* if there exists a codex Λ such that:

(i) for every $s \in A$ there is a chain $s = s_1 \to_\Lambda s_2 \ldots \to_\Lambda s_L$ where $s_L \in T(\Lambda)$.

(ii) for no $s \in R$ does there exist a chain $s = s_1 \to_\Lambda s_2 \ldots \to_\Lambda s_L$ where $s_L \in T(\Lambda)$.

Proposition 4: The set A is implementable in the alternative sense if and only if it is truthfully implementable.

Proof: If A is truthfully implementable, then there exists a codex Λ such that $P(\Lambda) = T(\Lambda) = A$. Part (i) of the alternative definition is satisfied since for any $s \in A$, $s \to_\Lambda s \in T(\Lambda)$. Part (ii) is satisfied since if there exists $s \in R$ and a chain $s = s_1 \to_\Lambda s_2 \ldots \to_\Lambda s_L$ where $s_L \in T(\Lambda)$, then there exists some l for which $s_l \in R$, $s_{l+1} \in T(\Lambda)$ and $s_l \to_\Lambda s_{l+1}$, contradicting the assumption that Λ implements A. Hence, A is implementable in the alternative sense.

On the other hand, assume that Λ implements the set A in the alternative sense. By (ii), there is no member of R in $T(\Lambda)$ and thus by Lemma 2 for any $s \in R$ there exists some s' such that $s \to_\Lambda s'$ and by Lemma 3 we can assume w.l.o.g. that $s'Ns$. Had s' been in A, then by (i) there would have been a chain $s' = s_1 \to_\Lambda s_2 \ldots \to_\Lambda s_L$ with $s_L \in T(\Lambda)$ and then we would have $s \to_\Lambda s_1 \to_\Lambda s_2 \ldots \to_\Lambda s_L$, contradicting (ii). Thus, $s' \in R$. Consider the codex

$\Lambda' = \{\varphi(s,s') \mid s \in R\}$. Then, $P(\Lambda') = T(\Lambda') = A$. ■

6. Implementation (not necessarily truthful)

The main two goals of this section are to show that implementation can be achieved by using a canonical codex (Proposition 6) and to characterize the class of implementable sets (Proposition 7). We start with an auxiliary claim:

Proposition 5: A set A is implementable by a canonical codex if and only if there is a reflexive binary relation $\rightarrow$ satisfying:

(1) Anti-symmety.

(2) $P(\rightarrow) = A$.

(3) If $s \rightarrow s'$ and $s \neq s'$, then sNs'.

(4) for every s there is at most one s' such that $s \rightarrow s'$.

(5) for every $s \in P(\rightarrow) - T(\rightarrow)$, there is $t \in R$ such that $t \rightarrow s$.

Proof: Assume that A is implementable by a canonical codex Λ. The relation $\rightarrow_\Lambda$ satisfies properties (1,2,3,4,5) since the coherence of the codex implies (1), the implementability of A by the codex is equivalent to (2) and the fact that the codex is canonical implies (3,4,5).

On the other hand, given a relation $\rightarrow$ that satisfies (1,2,3,4,5), consider $\Lambda = \{\varphi(s,s') \mid s \neq s' \text{ and } s \rightarrow s'\}$. (1) implies that the codex is coherent. The relation $\rightarrow_\Lambda$ is equal to $\rightarrow$ and using (2) we have $P(\Lambda) = P(\rightarrow_\Lambda) = P(\rightarrow) = A$. (3,4,5) imply that the codex is canonical. ■

Proposition 6: If the set A is implementable, then it is implementable by a canonical codex.

Proof: Let Λ be a codex that implements A. We start with the relation $\rightarrow_\Lambda$ and modify it to become a relation satisfying the five properties in Proposition 5.

The relation $\rightarrow_\Lambda$ is reflexive, satisfies (1,2) and in addition has the following property:

(6) Betweenness: If $s \rightarrow_\Lambda s'$ and t is a profile "between" s and s', then $s \rightarrow_\Lambda t \rightarrow_\Lambda s'$.

First, define a new reflexive relation $\rightarrow$ as follows:

(a) For every $s \in A - T(\Lambda)$, choose one profile $s' \in T(\Lambda)$ such that $s \rightarrow_\Lambda s'$ and define

$s \to s'$.

(b) For every $s \in R$, choose one profile $s' \neq s$ for which $s \to_\Lambda s'$. Since $\to_\Lambda$ satisfies (6), we can assume that $s'Ns$. Since $s \notin P(\Lambda)$, $s' \notin T(\Lambda)$. Define $s \to s'$.

The relation $\to$ satisfies (1,2,4) and:

(7) If $s \in R$ then there is a unique s' such that $s \to s'$ and $s' \notin T(\to)$ and $s'Ns$. If $s \in A$ and $s \to s'$, then $s' \in T(\to)$ and all profiles between s and s' are in A.

We now modify the relation $\to$ recursively as follows:

(i) For every $s \in A - T(\to)$ such that the set $N(s) \cap T(\to) \neq \emptyset$ and $s \to x$ for $x \notin N(s)$, divert the relation from $s \to x$ to $s \to y$ for some $y \in N(s) \cap T(\to)$.

(ii) Let $s \in A$ be such that $s \to s'$ and $s' \notin N(s)$. Let s'' be a neighbor of s between s and s'. By (7), $s'' \in A$ and by (2) there exists $s''' \in T(\to)$ such that $s'' \to s'''$. Delete $s'' \to s'''$ and $s \to s'$ from the relation and add $s \to s''$. If there is a profile $r \to s''$, then $r \in R$ and by (7), s'' and r are neighbors. Both s and r are neighbors of s'' and let $t = N(s, s'', r)$ (the other joint neighbor of s and r). By (i), $t \notin T(\to)$. If $t \in A$, then add $r \to t$. If $t \in R$, then delete $t \to t'$ (t' can be r!) and add $r \to t$ and $t \to s$. The new relation satisfies (1), (2), (4) and (7) but with one less element in A, which goes to a non-neighbor.

Go back to (i). Following a finite number of iterations we obtain a relation satisfying (1,2,3,4).

Finally, for every $s \in A$ for which $s \to t$ and there is no $r \to s$ for some $r \in R$, we can omit the arrow $s \to t$ to obtain a relation that satisfies 5 as well. ■

Proposition 7: The set A is implementable if and only if every connected component of R contains (i) a cycle or (ii) a profile r such that there are two profiles $s, t \in A$ and $rNsNt$.

Proof: Assume that A is implementable. By Proposition 6, it is implementable by a canonical codex Λ and by Proposition 5 there is a binary relation $\to$ satisfying (1,2,3,4,5). Consider a connected component Y of R. By (2,3), every $r \in Y$ has a neighbor $s(r)$ such that $r \to s(r)$. If for every $r \in Y$ the profile $s(r) \in R$, then Y must contain a cycle. Otherwise, there is an $r \in Y$ with $r \to s$ and $s \in A$. Then, by (2), it must be that $s \in P(\to) - T(\to)$ and thus there must be some $t \in T(\to) \subseteq A$ such that $s \to t$ and by (3) $rNsNt$.

In the other direction, let $Y_1,..,Y_N$ be a sequence of all connected components of R. If $N = 0$, the set $A = S$ is truthfully implementable (Proposition 3(1)). If $N > 0$, we inductively construct a relation $\rightarrow$ which at the end of stage $n-1$ will satisfy (1,3,4,5) and $P(\rightarrow) = S - Y_1 \cup...\cup Y_{n-1}$ as well as $P(\rightarrow) - T(\rightarrow) \subseteq A$ (and thus $Y_n \cup...\cup Y_N \subseteq T(\rightarrow)$). At the end of stage $n = N$, we obtain a relation satisfying (1,2,3,4,5) and by Proposition 5 the set A is implementable.

We now describe the n'th stage of the inductive construction of $\rightarrow$:

(i) The modification of $\rightarrow$ for the case in which Y_n contains a cycle is straightforward (following the construction in Proposition 2).

(ii) If there exists $r \in Y_n$ that is a neighbor of $s \in P(\rightarrow) - T(\rightarrow)$, then we can extend the relation $\rightarrow$ by adding $r \rightarrow s$ and $\{x \rightarrow y \mid x \in Y_N$ and y is a neighbor of x on the path from x to $r\}$ (there is only one path from x to r since Y_n does not contain a cycle).

We can now concentrate on the case in which there is $r^* \in Y_n$ such that $r^*Ns^*Nt^*$ and $s^*,t^* \in A$ and there is no $r \in Y_n$ that has a neighbor $s \in P(\rightarrow) - T(\rightarrow)$.

(iii) Next, we show that it can be assumed that there is no s such that $s \rightarrow s^*$.

If there is a profile s such that $s \rightarrow s^*$, then $s \notin R$ since if $s \in R$ it must be that $s^* \in P(\rightarrow) - T(\rightarrow)$, a situation already covered in (ii). Therefore, assume that $s \rightarrow s^*$ and $s \in A$. By property (5) of $\rightarrow$, there is $r \in R$ such that $r \rightarrow s$. The profile $x = N(r^*,s^*,s) \notin R$ since if $x \in R$ it must belong to Y_n and xNs, a case already covered in (ii). Also, $x \notin P(\rightarrow) - T(\rightarrow)$ since r^*Nx. Thus, $x \in T(\rightarrow)$ and we can delete $s \rightarrow s^*$ and add $s \rightarrow x$.

(iv) We are left with the situation in which $r^*Ns^*Nt^*$, $s^*,t^* \in A$, $s^* \in T(\rightarrow)$ and there is no $s \rightarrow s^*$.

If s^* has a neighbor x in $A \cap T(\rightarrow)$, then we can extend the relation $\rightarrow$ such that $r^* \rightarrow s^* \rightarrow x$ and for any other $r \in Y_N$ we can add $r \rightarrow s$ where $(r,s,..,r^*)$ is the path from r to r^* in Y_N.

Otherwise, t^*, which is in A, is not in $T(\rightarrow)$ and by (5) there are some profiles in R which are directed to t^*.

For every r such that $r \rightarrow t^*$, let $x(r) = N(r,t^*,s^*)$. We have already dealt with the case in which for at least one r we have $x(r) \in A \cap T(\rightarrow)$. We are left with two possibilities to consider:

(a) If $x(r) \in P(\rightarrow) - T(\rightarrow)$, i.e., there is $y \in A$ such that $x(r) \rightarrow y$, we can redirect $r \rightarrow x(r)$.

(b) If $x(r) \in R$ it must be in $Y_1 \cup \ldots \cup Y_{n-1}$ since $x(r)Nr$ and $r \in Y_1 \cup \ldots \cup Y_{n-1}$. Then, for each such r redirect $r \to x(r)$ and $x(r) \to s^*$.

There are no remaining profiles directed to t^* and as before we can extend the relation such that $r^* \to s^* \to t^*$ and $\{r \to s \mid r \in Y_n$ and $(r, s, .., r^*)$ is the path from r to r^* in $Y_N\}$. ■

Corollary: (1) If there exists $s^* \in R$ such that $A \supseteq N(s^*)$ and for any $x \in N(s^*)$ we have $N(x) \subseteq R$, then A is not implementable.

(2) If all connected components of A are singletons and A is not truthfully implementable, then A is not implementable.

The set $A = \{(0,1,0,0), (0,0,1,0), (0,0,0,1), (1,1,1,0), (1,1,0,1), (1,0,1,1)\}$ is an example of a set satisfying (2) but not (1).

Using the above characterization, Proposition 8 presents three families of implementable sets. In the first, each set A has the property that the truth of a particular variable in a profile guarantees that the profile is in A. The second consists of all sets A, each of which contains all but at most K profiles. The third consists of all sets A that have the property that if a profile s is in A then any other profile that agrees with s on the variables for which $s(v) = T$ is also in A. (For example: the set of all profiles in which $(v_1 \wedge v_2) \vee (v_3 \wedge v_4 \wedge v_5)$ is satisfied.)

Proposition 8: For $K \geq 3$, any set A that satisfies at least one of the following conditions is implementable:

(1) $A \supseteq T(v)$ for some variable v (recall that $T(v)$ is the set of all profiles in which the variable v receives the value T).

(2) $|R| \leq K$.

(3) A is monotonic in the following sense: if $s \in A$ and s' is a profile such that, for every variable v, whenever $s(v) = T$ also $s'(v) = T$, then $s' \in A$.

Proof:

(1) Every profile $s \in R$ assigns the truth value F to the variable v and is a neighbor of a profile in $T(v)$, which has another neighbor in $T(v)$.

(2) If $|R| \leq K$, then any $r \in R$ has a neighbor s in A and if s does not have K neighbors in R

it must have a neighbor in A. If there exists $s^* \in A$ such that $R = N(s^*)$, then for every $r \in R$ there is a ray $(s^*, r, n(r), n^2(r))$ and $n(r)$ and $n^2(r)$ must be in A.

(3) The case $A = \{alltruth\}$ is dealt with in Proposition 3(1). Otherwise A is a connected set (all profiles are connected to *alltruth*) which is not a singleton. The set R is connected (since if it is not empty all profiles are connected to *allfalse*). There must be a profile in R which is a neighbor of a profile in A which in turn is a neighbor of another profile in A. ■

7. Discussion

7.1. Experimental Evidence

We obviously do not view the bounded rationality element in our model as an exact description of reality. Nevertheless, we believe that it captures some elements of real life. The following series of experiments provides some supporting evidence. Subjects from more than 30 countries who had all taken a game theory course and had registered on the site gametheory.tau.ac.il were asked to participate in a short web-based experiment. The subjects were first asked the following three questions:

1) On most days, do you go to bed **before** midnight or **after** midnight?

2) Which of the following do you prefer: **cheese** cake or **chocolate** cake?

3) Were you born on an **odd** or **even** day of the month?

After answering the three questions, the subjects were presented with a new screen:

"Assume now that as part of a marketing campaign you have been offered the chance to participate in a lottery. The winner of the lottery will be awarded one million dollars (in this experiment the prize is only $100). In order to be eligible to participate, you must answer three questions about yourself and your answers must not violate any of the following six restrictions [the restrictions were presented in random order]:

R1: If you usually go to bed before midnight and you prefer chocolate cake,
then you must have been born on an even day of the month.

R2: If you prefer chocolate cake and you were born on an odd day of the month,
then you must usually go to bed before midnight.

R3: If you usually go to bed after midnight and you prefer cheese cake,
then you must have been born on an odd day of the month.

R4: If you usually go to bed after midnight and you prefer chocolate cake,

then you must have been born on an odd day of the month.

R5: If you prefer cheese cake and you were born on an even day of the month,

then you must usually go to bed after midnight .

R6: If you usually go to bed before midnight and you were born on an even day of the month,

then you must prefer cheese cake.

Assume that you very much want to participate in the lottery and you know that the organizers have no way of verifying whether your answers are true. How would you answer the following three questions in this case?

1) Do you usually go to bed before or after midnight?

2) Which of the following do you prefer: cheese cake or chocolate cake?

3) Were you born on an odd or even day of the month?"

Letting v_1 ="before midnight", v_2 = "cheese cake" and v_3="odd day of the month", the codex above, denoted by Λ_1, consists of six rules: $v_1 \wedge -v_2 \rightarrow -v_3$, $-v_2 \wedge v_3 \rightarrow v_1$, $-v_1 \wedge v_2 \rightarrow v_3$, $-v_1 \wedge -v_2 \rightarrow v_3$, $v_2 \wedge v_3 \rightarrow -v_1$ and $v_1 \wedge -v_3 \rightarrow v_2$. The induced guidance relation is: $111 \rightarrow_{\Lambda_1} 011$, $100 \rightarrow_{\Lambda_1} 110$, $010 \rightarrow_{\Lambda_1} 011$, $101 \rightarrow_{\Lambda_1} 100$, $001 \rightarrow_{\Lambda_1} 101$ and $000 \rightarrow_{\Lambda_1} 001$. Thus, $T(\Lambda_1) = \{011,\ 110\}$ and $P(\Lambda_1) = T(\Lambda_1) \cup \{111,\ 100, 010\}$.

We partitioned the subjects into three groups $T = T(\Lambda_1)$, $P = P(\Lambda_1) - T(\Lambda_1)$ and $R = R(\Lambda_1)$, according to their "declared profile" on the first screen. Each row in the following table refers to one of these groups. The first column presents the proportion of subjects in each group whose answers in the second screen belong to *T*. The second column, denoted by "Honest", presents the proportion of subjects in each group who submitted the same profile in the second screen as in the first. (Notice that 9% of the subjects in *T* answered successfully by reporting the profile in *T* on the second screen, which is not the one they declared initially.) The third column, denoted by "Other", presents the proportion of subjects in each group whose answer was neither in *T* nor honest.

Λ_1	**Success Rate**	**Honest**	**Other**	**N**
T	80%	71%	20%	104
P	54%	29%	17%	180
R	36%	34%	30%	261

Following are our main observations:

1) The results support our basic assumption that the ability of a subject to come up with a persuasive profile strongly depends on his true profile. While 80% of the subjects in *T* submitted a persuasive profile, the success rate dropped to 54% among the subjects in *P* and to 36% among the subjects in *R*.

2) The median response time of successful subjects increased from 125*s* for subjects in *T* to 157*s* for subjects in *P* and even more dramatically to 317*s* for subjects in *R*. This supports our assumption that subjects in *R* find it more difficult to come up with a persuasive profile than subjects in *P* and *T*.

3) According to Λ_1, each of the three profiles in *P* is guided by the codex to a single profile in *T* (two are guided to 011 and one to 110). Indeed, of the 97 subjects in *P* who submitted a persuasive profile, 68% followed the guide. This result supports our main assumption that subjects use the codex as a guide in coming up with a persuasive profile using their true profile as a starting point.

4) The choices of the 251 subjects in $P \cup R$ who failed to submit a persuasive profile are far from being random. 56% of these subjects were honest while 35% chose a profile that is confirmed by a rule in the codex, in the sense that the profile satisfies both its antecedent and its consequent (100, 101 or 001). Only 9% chose a profile that was not confirmed by any of the rules (111, 010 or 000).

5) One could suggest an alternative model of bounded rationality according to which a subject considers only his true profile and the (three) neighboring ones. However, the results do not support this hypothesis. First, note that for subjects with the true profiles 111 and 010, the two persuasive profiles are neighboring ones. However, they are guided by the codex only to 011 (and not to 110). Indeed, 75% of the 72 subjects who submitted a persuasive profile followed the guide and chose 011. Second, the success rate of the 001 subjects (37%) who had a neighboring profile in *T* was no different than those for the other two *R* profiles (101 and

000), which do not have a neighboring profile in *T* (37% and 33%, respectively).

An alternative explanation for the popularity of 011 among the 111 and 010 subjects is that 011 is confirmed by two rules. Therefore, we conducted a second experiment with a modified codex, denoted by Λ_2, whose guidance relation is $111 \rightarrow_{\Lambda 2} 011$, $100 \rightarrow_{\Lambda_2} 110$, $010 \rightarrow_{\Lambda_2} 110$, $001 \rightarrow_{\Lambda 2} 011$, $101 \rightarrow_{\Lambda_2} 100$ and $000 \rightarrow_{\Lambda_2} 001$. For this codex, $T(\Lambda_2) = T(\Lambda_1)$ but $P(\Lambda_2) - T(\Lambda_2)$ consists of four profiles: 111 and 001 (guided by the codex to 011) and 100 and 010 (guided to 110). The following table summarizes the main results:

Λ_2	**Success Rate**	**Honest**	**Other**	**N**
T	88%	75%	12%	52
P	63%	27%	10%	123
R	45%	15%	40%	65

Once again, we observe a strong dependence of the success rate on the subject's true profile. Almost all *T* profiles, 63% of the *P* profiles and only 45% of the *R* profiles came up with a persuasive profile. Particularly interesting is the group of 123 subjects whose profile is in *P*. Each of the four profiles in *P* is guided by the codex to a unique profile in *T*. Of the 78 successful subjects in *P*, 51 subjects (65%) seem to have been guided by the codex. We believe that this result strongly supports our main assumption that individuals first determine whether their true profile satisfies the codex and if it does not then they consider a profile to which they are guided by the codex.

Finally, we also tried another codex, denoted by Λ_3, which truthfully implements $\{110, 011\}$. The induced guidance relation is $111 \rightarrow_{\Lambda_3} 101$ $100 \rightarrow_{\Lambda_3} 101$ $010 \rightarrow_{\Lambda_3} 000$, $101 \rightarrow_{\Lambda_3} 100$ $001 \rightarrow_{\Lambda_3} 101$ and $000 \rightarrow_{\Lambda_3} 001$. The following table summarizes the results:

Λ_3	**T**	**Honest**	**Other**	**N**
T	81%	77%	19%	26
R	34%	44%	22%	100

Once again, there is a dramatic difference between the success rates of the *T*'s (81%) and the *R*'s (34%). The *R*'s success rate and their median response time (332s) are similar to those

of the *R*'s in the previous experiments and only one *R* subject chose a profile not confirmed by any of the rules in the codex.

7.2. *Related Literature*

The idea that cheating is difficult is, of course, not a new one. Within the economic literature, it appears in Kamien and Zemel (unpublished, 1990) among others. They reinterpreted Cook's Theorem (see Cook (1971)), which proves the NP completeness of deciding whether a given Boolean formula in conjunctive normal form has an assignment that makes the formula true.

Kartik (2009) analyzed a model of persuasion in which a speaker incurs a cost if he chooses to misrepresent his private information. Inflated language naturally arises in this environment.

The idea that the framing of a mechanism may also provide some guidance to the participants appeared in (the completely ignored) Glazer and Rubinstein (1996). In that paper, we introduced the concept of "implementation via guided iterative elimination of dominated strategies in a normal form game" and showed that it is equivalent to "implementation using a subgame perfect equilibrium of an extensive game with perfect information".

The idea that the mechanism itself can affect agents' preferences and thus the implementability of social outcomes appears in Glazer and Rubinstein (1998). In that paper, a number of experts receive noisy signals regarding a public decision. Two "cultures" were compared: In the first, the experts are driven only by the public motive to increase the probability that the desirable action will be taken. In the second, each expert is also driven by a private motive to have his recommendation adopted. We show that only the second culture gives rise to a mechanism whose unique equilibrium outcome achieves the public target.

A model of implementation with bounded rationality can be found in Eliaz (2002) who investigated the implementation problem when some of the players are "faulty", in the sense that they fail to act optimally. Eliaz introduces a solution concept called "fault-tolerant implementation", which requires robustness to deviations from equilibrium and shows that under symmetric information any choice rule that satisfies certain properties can be

implemented if the number of faulty players is sufficiently small.

7.3. *Conclusion*

The model presented here facilitates the analysis of some basic considerations used by a principal in attempting to elicit information from an agent who may have an incentive to cheat. The principal would like the mechanism to be complex enough that an agent, whose interests clash with his own, will not be guided by the mechanism itself to successfully distort the information he is conveying. At the same time, the principal would like the mechanism to be simple enough that an agent whose interests coincide with his own will be able to persuade him.

Following are some of our main insights:

(1) In some cases, it is optimal for the listener to use a codex that will help the speaker to "alter the truth", that is, present a false but persuasive profile. This result is consistent with the casual observation that some exaggeration is sometimes viewed as necessary in real-life situations (see Kartik, Ottaviani and Squintani (2007)).

(2) If the circumstances under which the listener should (from his point of view) accept the speaker's request are rare, then truthful implementation is easy. This will be accomplished by means of a codex that will trap all "undeserving" speakers (i.e., speakers whose profile should not be accepted) in a "circle of lies." In other words, an undeserving speaker is (mis)guided by the codex to pretend to be another undeserving speaker whose profile is rejected by the codex and who, in turn, is guided by the codex to pretend to be a third undeserving speaker whose profile is rejected and so on. This procedure continues until one of the undeserving speakers is guided by the codex to present a profile that appears previously in the chain.

(3) If the circumstances under which the listener should reject the speaker's request are rare, then the optimal mechanism requires the speaker, in some circumstances, to cheat successfully. This occurs because the codex sometimes guides a speaker with an undeserving profile to pretend to be a speaker with a deserving one, who himself is rejected by the codex but is guided to another profile which is accepted.

Most importantly, the paper suggests a new direction for the study of mechanism design with boundedly rational agents.

References

Cook, Stephen A. (1971). "The Complexity of Theorem Proving Procedures". *Proceedings Third Annual ACM Symposium on Theory of Computing*, 151-158.

Eliaz, Kfir (2002). "Fault Tolerant Implementation". *Review of Economic Studies*, 69, 589-610.

Glazer, Jacob and Ariel Rubinstein (1996). "An Extensive Game as a Guide for Solving a Normal Game". *Journal of Economic Theory*, 70, 32-42.

Glazer, Jacob and Ariel Rubinstein (1998). "Motives and Implementation: On the Design of Mechanisms to Elicit Opinions". *Journal of Economic Theory*, 79, 157-173.

Glazer, Jacob and Ariel Rubinstein (2004). "On Optimal Rules of Persuasion". *Econometrica*, 72, 1715-1736.

Glazer, Jacob and Ariel Rubinstein (2006). "A Study in the Pragmatics of Persuasion: A Game Theoretical Approach". *Theoretical Economics*, 1, 395-410.

Green, Jerry R. and Jean-Jacques Laffont (1986). "Partially Verifiable Information and Mechanism Design". The Review of Economic Studies, 53, 447-456.

Kartik, Navin (2009). "Strategic Communication with Lying Costs". *Review of Economic Studies*, 76, 1359-1395.

Kartik, Navin, Marco Ottaviani and Francesco Squintani (2007). "Credulity, Lies, and Costly Talk". *Journal of Economic Theory*, 134, 93-116.

Kamien , Morton I. and Eitan Zemel (1990). "Tangled Webs: A Note on the Complexity of Compound Lying". (mimeo)