VNIVERSITAT
IDȢVALÈNCIA

Nash Codes for Noisy Channels

> Penélope Hernandez
  Universitat de València and ERI-CES, Spain

> Bernhard von Stengel
  London School of Economics, United Kingdom

May, 2012

Nash Codes for Noisy Channels[*]

Penélope Hernández[†]   Bernhard von Stengel[‡]

May 6, 2012

Abstract

This paper studies the stability of communication protocols that deal with transmission errors. We consider a coordination game between an informed sender and an uninformed decision maker, the receiver, who communicate over a noisy channel. The sender's strategy, called a code, maps states of nature to signals. The receiver's best response is to decode the received channel output as the state with highest expected receiver payoff. Given this decoding, an equilibrium or "Nash code" results if the sender encodes every state as prescribed. We show two theorems that give sufficient conditions for Nash codes. First, a receiver-optimal code defines a Nash code. A second, more surprising observation holds for communication over a binary channel which is used independently a number of times, a basic model of information transmission: Under a minimal "monotonicity" requirement for breaking ties when decoding, which holds generically, *any* code is a Nash code.

Keywords: sender-receiver game, communication, noisy channel.

OR/MS subject classification: Communications; Games/group decisions: Noncooperative.

JEL subject classification: C72, D82.

AMS subject classification: 91A28, 94A05.

---

1 Introduction

Information transmission is central to economic interactions and to the operation of organizations. This paper is concerned with communication errors, which we call *noise*, and the stability of related communication protocols, as expressed by the condition of *Nash equilibrium* in a game-theoretic setting.

Information transmission is often modeled as a sender-receiver game between an informed expert and an uninformed decision maker. We also study a sender-receiver game, but should say at the outset that our goal and assumptions differ from the common economic models, most notably signalling games (Spence, 1973) and the seminal work of Crawford and Sobel (1982) and its extensions. These models typically assume perfect communication, but find that information is not transmitted faithfully for *strategic* reasons because of conflicting incentives of sender and receiver (see the surveys by Kreps and Sobel, 1994, and Sobel, 2010). In contrast, we assume communication errors as *given*, due to distorted signals or imprecisely worded or misunderstood messages. This kind of noise is considered in information theory (Shannon, 1948) and in studies of language and ambiguity (Nowak and Krakauer, 1999).

Communication errors are common and usually have negative consequences for both parties (to different degrees), as illustrated by the following examples: A pilot misunderstands the command of an air traffic controller, with a resulting aviation accident (Cushing, 1994). A map on a webpage displays a holiday home close to the beach when it is not, and the holiday maker sues the landlord for misrepresentation afterwards. A landing card is filled in wrongly and immigration is delayed. A student does not understand a lecture and scores poorly in the exam. These errors may be attributable in various ways to the sender (sometimes for strategic reasons), the transmission, or the receiver. We think that a basic model that assumes these errors as given deserves a game-theoretic analysis.

The model

We consider the classic model of the *discrete noisy channel*. The channel has a finite set of input and output symbols and known transition probabilities that represent the possible communication successes and errors. The channel may also be used repeatedly, with independent errors, which we study for the important case of the binary channel that has only two symbols.

The model of the noisy channel is due to Shannon (1948), who pioneered information theory, which has been crucial for the design of reliable data communication with devices that range from space probes to disk drives. These engineering efforts typically aim for "codes" that have a good reliability and rate of information transmission (see Cover and Thomas, 1991; Callager, 1968; MacKay, 2003). We ask a different question: Given a communication protocol defined by a code, do sender and receiver have an incentive to follow this protocol?

Using game theory, we study this question with a sender-receiver game where the interests of sender and receiver are aligned. One of finitely many states of nature is chosen at random. The sender is informed the state and transmits a signal via a discrete noisy channel to an uninformed receiver who makes a decision. The sender's strategy or *code* assigns to each state of nature a specific signal or "codeword" that is the input to the channel. The receiver's strategy decodes the distorted signal that is the channel output as one of the possible states. Both players receive a (possibly different) positive payoff only if the state is decoded correctly, otherwise payoff zero.

In equilibrium, the receiver decodes the channel output as the state with highest expected payoff. This receiver condition is the well-known "maximum likelihood" decoding in the special case of uniform priors and equal utilities. The equilibrium condition for the sender means that she chooses for each state the prescribed codeword as her best response, that is, no other channel input has a higher probability of being decoded correctly with the given receiver strategy.

A *Nash code* is a code together with a best-response decoding that defines a Nash equilibrium. So we assume the straightforward equilibrium condition for the receiver and require that the code fulfills the more involved sender condition. (Of course, both conditions are necessary for equilibrium.)

Our results

We present two main results about Nash codes. Our first result concerns discrete channels with arbitrary finite sets of input and output symbols. We show that already for three symbols, not every code defines a Nash equilibrium. However, a Nash code results if the expected payoff to the receiver cannot be increased by replacing a single codeword with another one (Theorem 5). So these *receiver-optimal* codes are Nash codes. This is closely related to *potential games* and provides a method to construct Nash codes (Proposition 6): Essentially, if the sender can change a codeword to improve the probability that a state is decoded correctly (which violates Nash equilibrium), then this defines a better code for the receiver.

In short, without any constraints on the channel, and for any best-response decoding, "good codes" (that is, receiver-optimal codes) are Nash codes. The method to show this result is not deep; its purpose is to analyze our model. The key assumption is that an improvement in decoding probability benefits both sender and receiver (see condition (10) below). However, a *sender-optimal* code is *not* necessarily a Nash code if sender and receiver give different utilities to a correct decoding of the state of nature. This happens if the sender can use an unused message to transmit the information about the state more reliably (see the discussion following the example in (13) below).

Our second, more surprising and technically challenging result concerns the *binary channel* where codewords are strings of bits with independent error probabilities for each bit, a fundamental model of information transmission. Then *any* code is a Nash code (Theorem 8). The only requirement for the decoding is that the receiver breaks ties between

states *monotonically*, that is, in a consistent manner; this holds for natural tie-breaking rules, and ties do not even occur if states of nature have different generic prior probabilities or utilities. (In a side result, Proposition 9, we also characterize deterministic monotonic tie-breaking rules as those that select among the tied states in a fixed given order.) So binary codes, as Nash codes, are very suitable for information transmission because the agents never have an incentive to deviate from them.

There are several interpretations of this second result. One is "good news for the engineer": Binary codes, which are fundamental to the practice and theory of information theory, are "incentive compatible". The fact that they are Nash codes (assuming monotonic decoding) means that engineering issues such as high reliability and rate or algorithmic ease of coding and decoding are "orthogonal" to the stability of the code in terms of incentives. On the other hand, this can also be seen as a negative result: Nash equilibrium is not related to transmission quality. This last observation is not restricted to binary channels; for example, the sender-receiver game has always a "babbling equilibrium" where the sender's action is independent of the state and the receiver's action is independent of the channel output, with no information transmitted. The equilibrium property of codes with low payoffs may explain why inefficient (say, very bureaucratic) communication protocols persist in practice because no single participant has an incentive to deviate from them.

Related literature

Sender-receiver games studied in the economic literature typically assume communication without transmission errors. In their seminal paper, Crawford and Sobel (1982) study such a game where the set of possible states, messages, and receiver's actions is an interval. Payoffs depend continuously on the difference between state and action, and differ for sender and receiver. In equilibrium, the interval is partitioned into finitely many intervals, and the sender sends as her message only the partition class that contains the state. Thus, the sender only reveals partial information about the state, which can be seen as noise being introduced strategically.

Even in rather simple sender-receiver games, players can get higher equilibrium payoffs when communicating over a channel with noise than with perfect communication (Myerson, 1994, Section 4). Blume, Board, and Kawamura (2007) extend the model by Crawford and Sobel (1982) by assuming communication errors. The noise allows for equilibria that improve welfare compared to the Crawford-Sobel model. The construction partly depends on the specific form of the errors so that erroneous transmissions can be identified; this does not apply in our discrete model. In addition, in our model players only get positive payoff when the receiver decodes the state correctly, unlike in the continuous models by Crawford and Sobel (1982) and Blume et al. (2007). On the other hand, compared to perfect communication, noise may prevent players from achieving common knowledge about the state of nature (Koessler, 2001).

Game-theoretic models of communication have been used in the study of language. Lewis (1969) describes language as a "convention" with mappings between states and signals,

and argues that these should be bijections. Nowak and Krakauer (1999) use evolutionary game theory to show how languages may evolve from "noisy" mappings; Wärneryd (2003) shows that only bijections are evolutionary stable. However, even ambiguous sender mappings (where one signal is used for more than one state) together with a mixed receiver population may be "neutrally stable" (Pawlowitsch, 2008); the randomized receiver strategy can be seen as noise. Blume and Board (2009) use the noisy channel to model vagueness in communication. Lipman (2009) discusses how vagueness can arise even for coinciding interests of sender and receiver. Ambiguous signals arise when the set of messages is smaller than the set of states, which may reflect communication costs for the sender (Jäger, Koch-Metzger, and Riedel, 2011). For the sender-receiver game with a noisy binary channel, Hernández, Urbano, and Vila (2010a) describe the equilibria for a specific code that can serve as a "universal grammar"; the explicit receiver strategy allows to characterize the equilibrium payoff.

Noise in communication is relevant to models of persuasion, where the sender wants to induce the receiver to take an action. Glazer and Rubinstein (2004; 2006) study binary receiver actions; the sender may reveal limited information about the state of nature as "evidence". The optimal way to do so is a receiver-optimal mechanism. In a more general setting, Kamenica and Gentzkow (2011) allow the sender to commit to a strategy that selects a message for each state, assuming the receiver's best response using Bayesian updating; the sender may generate noise by selecting the message at random. Subject to a certain Bayesian consistency requirement, the sender can commit to her best possible strategy.

Outline of the paper

Section 2 describes our model and characterizes the Nash equilibrium condition. For channels with any number of symbols, Section 3 gives an example that some codes may not be Nash codes, shows that receiver-optimal codes are, and discusses the relation to potential functions. In Section 4, we consider binary codes, and state the main Theorem 8, which is proved in the Appendix. It requires the condition of "monotonic" decoding when ties occur, for example in a fixed order among the states as when they have generic priors. In Section 5 it is shown that this is in fact the only general deterministic monotonic tie-breaking rule.

2   Nash codes

We consider a game of two players, a sender (she) and a receiver (he). First, nature chooses a *state i* from a set $\Omega = \{0, 1, \ldots, M-1\}$ with positive *prior* probability q_i. Then the sender is fully informed about i, and sends a message to the receiver via a noisy channel. After receiving the message as output by the channel, the receiver takes an action that affects the payoff of both players.

The channel has finite sets X and Y of input and output symbols, with noise given by transition probabilities $p(y|x)$ for each $x \in X$, $y \in Y$. The channel is used n times independently without feedback. When an input $x = (x_1, \ldots, x_n)$ is transmitted through the channel, it is altered to an output $y = (y_1, \ldots, y_n)$ according to the probability $p(y|x)$ given by

$$p(y|x) = \prod_{j=1}^{n} p(y_j|x_j). \tag{1}$$

This is the standard model of a memoryless noisy channel as considered in information theory (Cover and Thomas, 1991; MacKay, 2003).

The sender's strategy is to encode state i by means of a coding function or *code* $c : \Omega \to X^n$, which we write as $c(i) = x^i$. We call x^i the *codeword* or *message* for state i in Ω, which the sender transmits as input to the channel. The code c is completely specified by the list of M codewords $x^0, x^1, \ldots, x^{M-1}$, which is called the *codebook*.

The receiver's strategy is to decode the channel output y, given by a probabilistic *decoding function*

$$d : Y^n \times \Omega \to \mathbb{R}, \tag{2}$$

where $d(y, i)$ is the probability that y is decoded as i.

Sender and receiver have the common interest that the message is decoded correctly. That is, if the receiver decodes the channel output as the state i chosen by nature, then sender and receiver get positive payoff U_i and V_i, respectively, otherwise both get payoff zero. The channel transition probabilities, the transmission length n, and the prior probabilities q_i and utilities U_i and V_i for i in Ω are commonly known to the players.

We are interested in conditions so that the pair (c, d) defines a Nash equilibrium. In that case, we call c, under the assumption that decoding takes place according to d, a *Nash code*. We denote the expected payoffs to sender and receiver by $U(c, d)$ and $V(c, d)$, respectively.

The code c defines the sender's strategy. The best response of the receiver is the following. Given that he receives channel output y in Y^n, the probability that codeword x^i has been sent is, by Bayes's law, $q_i p(y|x^i)/\text{prob}(y)$, where $\text{prob}(y)$ is the overall probability that y has been received. The factor $1/\text{prob}(y)$ can be disregarded in the maximization of the receiver's expected payoff. Hence, a best response of the receiver is to choose with positive probability $d(y, i)$ only states i so that $q_i V_i p(y|x^i)$ is maximal, that is, so that y belongs to the set Y_i defined by

$$Y_i = \{y \in Y^n \mid q_i V_i p(y|x^i) \geq q_k V_k p(y|x^k) \ \forall k \in \Omega\}. \tag{3}$$

Hence, the best response condition for the receiver states that for any $y \in Y^n$ and $i \in \Omega$

$$d(y, i) > 0 \quad \Longrightarrow \quad y \in Y_i. \tag{4}$$

If $V_i = 1$ for all $i \in \Omega$, then Y_i in (3) is the set of channel outputs y so that the channel input x^i has *maximum likelihood*. (This term is sometimes used only for uniform prior probabilities, e.g. MacKay, 2003, p. 152, which we do not assume.) If the receiver has

different positive utilities V_i for different states i, then the receiver's best response maximizes $q_i V_i p(y|x^i)$.

We say that for a given channel output y, there is a *tie* between two states i and k (or the states are *tied*) if $y \in Y_i \cap Y_k$. If there are never any ties, then the sets Y_i for $i \in \Omega$ are pairwise disjoint, and the best-response decoding function is deterministic and unique according to (4).

We refer to the sets Y_i for $i \in \Omega$ as a "partition" of Y^n, which constrains the receiver's best-response decoding as in (4), even though some of these sets may be empty, and they may not always be disjoint if there are ties. In any case, $Y^n = \bigcup_{i \in \Omega} Y_i$.

Suppose that the receiver decodes the channel output with d according to (3) and (4) for the given code c with $c(i) = x^i$. Then (c, d) is a Nash equilibrium if and only if, for any state i, it is optimal for the sender to transmit x^i and not any other x in X^n as a message. When sending x, the expected payoff to the sender in state i is

$$U_i \sum_{y \in Y^n} p(y|x) d(y, i). \tag{5}$$

When maximizing (5), the utility U_i to the sender does not matter as long as it is positive; given that the state is i, the sender only cares about the probability that the channel output y is decoded as i. We summarize these observations as follows.

Proposition 1 *The code c with decoding function d is a Nash code if and only if the receiver decodes channel outputs according to* (3) *and* (4)*, and if and only if in every state i the sender transmits codeword $c(i) = x^i$ which fulfills for any other possible channel input x in X^n*

$$\sum_{y \in Y^n} p(y|x^i) d(y, i) \geq \sum_{y \in Y^n} p(y|x) d(y, i). \tag{6}$$

3   Receiver-optimal codes

In this section, we first ask whether every code is a Nash code, assuming that the receiver chooses a best response. We give a detailed example that demonstrates that this may not be the case, and that we use throughout the section. Then we show that every code that maximizes the receiver's payoff is a Nash code. The proof implies that this holds also if the receiver's payoff is locally maximal, that is, when changing only a single codeword, and the corresponding best response of the receiver, at a time. Finally, we discuss the connection with potential functions.

Consider a channel with three symbols, $X = Y = \{0, 1, 2\}$, which is used only once ($n = 1$), with the following transition probabilities:

$p(y\|x)$		y		
		0	1	2
	0	0.85	0.1	0.05
x	1	0.1	0.65	0.25
	2	0	0.3	0.7

$$(7)$$

Suppose that nature chooses the two states in $\{0,1\}$ with uniform priors $q_0 = q_1 = 1/2$. The sender's utilities are $U_0 = 2$ when the state is 0 and $U_1 = 8$ when the state is 1, and the receiver's utilities are $V_0 = 8$, $V_1 = 2$.

Consider the codebook c with $c(0) = x^0 = 0$ and $c(1) = x^1 = 1$, so the sender codifies the two states of nature as the two symbols 0 and 1, respectively. Given the parameters of this game and the sender's strategy c, the receiver's strategy assigns to each output symbol in $\{0,1,2\}$ one state. The following table (8) gives the expected payoff $q_i V_i\, p(y|x^i)$ for the receiver when the state is i and the output symbol is y.

$q_i V_i\, p(y\|x^i)$		y		
		0	1	2
i	0	3.4	0.4	0.2
	1	0.1	0.65	0.25

$$(8)$$

Table (8) allows us to compute the receiver's best response and the sets Y_i in (3). For each channel output y, the receiver chooses the state i with highest expected payoff. Hence, he decodes the channel output 0 as state 0 because $q_0 V_0\, p(0|x^0) = 3.4 > 0.1 = q_1 V_1\, p(0|x^1)$. In the same way, he decodes both channel outputs 1 and 2 as state 1. Notice that there are no ties, so the two sets Y_0 and Y_1 are disjoint, and the receiver's best response is unique and deterministic. That is, the receiver's best response d is given by $d(y,i) = 1$ if and only if $y \in Y_i$, where $Y_0 = \{0\}$ and $Y_1 = \{1,2\}$.

Is this code c given by the codebook $x^0, x^1 = 0,1$ a Nash code?

Given the partition of Y into Y_0 and Y_1 by the receiver strategy d, it easy to compute the sender payoff as in (5) when the states 0 and 1 are realized. For the first state 0, her payoff is $U_0 \sum_{y \in Y} p(y|0)d(y,0) = U_0 \sum_{y \in Y_0} p(y|0) = 2 \times p(0|0) = 1.7$. For the second state 1, her payoff is $U_1 \sum_{y \in Y} p(y|1)d(y,1) = U_1 \sum_{y \in Y_1} p(y|1) = 8 \times (p(1|1) + p(2|1)) = 8 \times (0.65 + 0.25) = 7.2$. The sender's (ex-ante) expected payoff is therefore $U(c,d) = q_0\, 1.7 + q_1\, 7.2 = 4.45$.

In order to check the Nash equilibrium property of (c,d), there should be no code c' so that $U(c',d) > U(c,d)$. Consider now the new sender strategy c' with codebook $0,2$, which differs from code c in the codeword $c'(1) = 2$ for state 1. The receiver's strategy d with Y_0 and Y_1 is fixed. State 0 is encoded by the same codeword $c(0) = c'(0) = 0$, so the sender's payoff for that state is 1.7 as before. However, for state 1, the signal sent is 2 instead of 1. Then the sender's payoff is $U_1 \sum_{y \in Y_1} U_1 p(y|2) = 8 \times (p(1|2) + p(2|2)) =$

$8 \times (0.3 + 0.7) = 8$, which is higher than her payoff 7.2 when sending signal 1. Her expected payoff increases to $U(c',d) = q_0 1.7 + q_1 8 = 4.85$. Consequently, the code c with codebook $0,1$ is not a Nash code.

In this example, changing the codebook c to c' improves the sender payoff from $U(c,d)$ to $U(c',d)$, where d is the receiver's best-response decoding for code c. In addition, it is easily seen that the receiver payoff also improves from $V(c,d)$ to $V(c',d)$, and his payoff $V(c',d')$ for the best response d' to c' is possibly even higher. This observation leads us to a sufficient condition for Nash codes.

Definition 2 *A* receiver-optimal code *is a code c with highest expected payoff to the receiver, that is, so that*

$$V(c,d) \geq V(\hat{c}, \hat{d})$$

for any other code $\hat{c}$, where d is a best response to c and $\hat{d}$ is a best response to $\hat{c}$.

Note that in this definition, the expected payoff $V(c,d)$ (and similarly $V(\hat{c}, \hat{d})$) does not depend on the particular best-reponse decoding function d in case d is not unique when there are ties, because the receiver's payoff is the same for all best responses d.

The following is the central theorem of this section. It is proved in three simple steps,[1] which give rise to a generalization that we discuss afterwards, along with examples and further observations.

Theorem 3 *Every receiver-optimal code is a Nash code.*

Proof. Let c be a receiver-optimal code with codebook $x^0, x^1, \ldots, x^{M-1}$, and associated best-response decoding d according to (3) and (4). Suppose c is not a Nash code. Then there exists a code $\hat{c}$ with codebook $\hat{x}^0, \hat{x}^1, \ldots, \hat{x}^{M-1}$ so that $U(\hat{c}, d) > U(c,d)$, that is,

$$\sum_{i \in \Omega} q_i U_i \sum_{y \in Y^n} p(y|\hat{x}^i) d(y,i) > \sum_{i \in \Omega} q_i U_i \sum_{y \in Y^n} p(y|x^i) d(y,i). \tag{9}$$

Step one: Clearly, (9) implies[2] that there exists at least one $i \in \Omega$ so that

$$\sum_{y \in Y^n} p(y|\hat{x}^i) d(y,i) > \sum_{y \in Y^n} p(y|x^i) d(y,i). \tag{10}$$

Consider the new code c' which coincides with c except for the codeword for state i, where we set $c'(i) = \hat{x}^i$. So the codebook for c' is $x^0, \ldots, x^{i-1}, \hat{x}^i, x^{i+1}, \ldots, x^{M-1}$. By (10), we also have

$$U(c',d) = \sum_{j \in \Omega,\ j \neq i} q_j U_j \sum_{y \in Y^n} p(y|x^j) d(y,j) + q_i U_i \sum_{y \in Y^n} p(y|\hat{x}^i) d(y,i)$$

$$> \sum_{j \in \Omega} q_j U_j \sum_{y \in Y^n} p(y|x^j) d(y,j) = U(c,d). \tag{11}$$

---

[1] We are indebted to Drew Fudenberg who suggested steps two and three.

[2] This claim follows also directly from Proposition 1, but we want to refer later to (9) as well.

Step two: In the same manner, (10) implies an improvement of the receiver function, that is,

$$V(c',d) > V(c,d). \tag{12}$$

Step three: Let d' be the best response to c', which with (12) implies

$$V(c',d') \geq V(c',d) > V(c,d).$$

Hence, code c' has higher expected receiver payoff than c. This contradicts the assumption that c is a receiver-optimal code. $\qquad\square$

The preceding theorem asserts that there is at least one Nash code. It can be found as a code with highest receiver payoff.

x^0, x^1	Y_0	Y_1	$p(y \in Y_0 \mid x^0)$	$p(y \in Y_1 \mid x^1)$	U	V
$0,1$	$\{0\}$	$\{1,2\}$	0.85	0.90	4.45	4.30
$0,2$	$\{0,1\}$	$\{2\}$	0.95	0.70	3.75	4.50
$1,0$	$\{1,2\}$	$\{0\}$	0.90	0.85	4.30	4.45
$1,2$	$\{0,1,2\}$	$\{\}$	1.00	0.00	1.00	4.00
$2,0$	$\{1,2\}$	$\{0\}$	1.00	0.85	4.40	4.85
$2,1$	$\{1,2\}$	$\{0\}$	1.00	0.10	1.40	4.10

$$\tag{13}$$

For our example, the table in (13) lists the six possible codebooks x^0, x^1, shown in the first column, that have distinct codewords ($x^0 \neq x^1$). For each code, the receiver's best response is unique. The best-response partition Y_0, Y_1 is shown in the second column. Using this partition, the third column gives the probabilities $p(y \in Y_i \mid x^i) = \sum_{y \in Y_i} p(y|x^i)$ that the codeword x^i is decoded correctly. The overall expected payoffs to sender and receiver are shown as U and V.

According to the rightmost column in (13), the unique receiver-optimal codebook is $2,0$, which is a Nash code by Theorem 3. We have already shown that $0,1$ is not a Nash code. Note, however, that this is the code with highest sender payoff. Hence, a "sender-optimal" code is not necessarily a Nash code. The reason is that, because sender and receiver have different payoffs for the two states, the sender prefers the code with large partition class Y_1 for state 1, but then can deviate to a better, unused message within Y_1.

It also easily seen from (13) that $1,0$ and $2,1$ are not Nash codes, either: Both codebooks have the same best-response partition $Y_0 = \{1,2\}$ and $Y_1 = \{0\}$ as the codebook $2,0$, but have lower payoff to the sender, so the sender can profitably deviate from $1,0$ or $2,1$ to $2,0$.

In (13), the codebook $1,2$ has the interesting property that the receiver decodes *any* channel output y as state 0; this holds because even the unaltered codeword $x^1 = 2$, when received as $y = 2$, fulfills $q_1 V_1 \, p(2|x^1) = 1 \times 0.7 < 4 \times 0.25 = q_0 V_0 \, p(2|x^0)$, so the receiver prefers to decode it as state 0. So here $Y_0 = Y^n$ and Y_1 is the empty set. Given

10

that the receiver's action is the same for any received channel output, the sender cannot improve her payoff by transmitting anything else. So the codebook $1, 2$ is a Nash code.

In fact, any sender-receiver game, irrespective of the players' payoffs, has a trivial "pooling" equilibrium where the sender's signal does not depend on the state,[3] and the receiver's best response decodes the uninformative channel output as the state i with highest expected payoff, in our game $q_i V_i$. In our example, such codes have equal codewords, with $x^0 = x^1$, all decoded as state 0; they are not listed in (13). The codebook $1, 2$ is potentially informative, but the receiver ignores the channel output due to his utility function.

Finally, the codebook c with codebook $0, 2$ in (13) is also a Nash equilibrium, which is seen as follows. Let d be the best response to c, with $Y_0 = \{0, 1\}$, $Y_1 = \{2\}$. As shown in the proof of Theorem 3, if the sender could profitably deviate from c to $\hat{c}$, then she could also profitably deviate to a code c' that differs from c in one codeword only. The possible codes c' have codebooks $1, 2$, where $c(0)$ is changed to $c'(0) = 1$, and $0, 1$, where $c(1)$ is changed to $c'(1) = 1$. In the first case, by (7), changing $c(0)$ from 0 to 1 changes $q_0 U_0 \sum_{y \in Y_0} p(y|0) = 0.85 + 0.1 = 0.95$ to $q_0 U_0 \sum_{y \in Y_0} p(y|1) = 0.1 + 0.65 = 0.75$, which is not an improvement. In the second case, changing $c(1)$ from 2 to 1 changes $q_1 U_1 \sum_{y \in Y_1} p(y|2) = 4 \times 0.70 = 2.8$ to $q_1 U_1 \sum_{y \in Y_1} p(y|1) = 4 \times 0.25 = 1$, which is not an improvement either. So c is indeed a Nash code.

The code c with codebook $0, 2$ is also seen to be a Nash code with the help of table (13) according to the proof of Theorem 3. Namely, it suffices to look for profitable sender deviations c' where only one codeword is altered, which would also imply an improvement to the receiver's payoff from $V(c, d)$ to $V(c', d)$, and hence certainly an improvement to his payoff $V(c', d')$ where d' is the best response to c'. For the two possible codes c' given by $1, 2$ and $0, 1$, the receiver payoff V does not improve according to (13), so c is a Nash code. By this reasoning, any "locally" receiver-optimal code, according the following definition, is also a Nash code.

Definition 4 A locally receiver-optimal code *is a code c so that no code c' that differs from c in only a single codeword gives higher expected payoff to the receiver. That is, for all c' with $c'(i) \neq c(i)$ for some state i, and $c'(j) = c(j)$ for all $j \neq i$,*

$$V(c, d) \geq V(c', d')$$

where d is a best response to c and d' is a best response to c'.

Theorem 5 *Every locally receiver-optimal code is a Nash code.*

Proof. Apply the proof of Theorem 3 from Step two onwards. □

Clearly, every receiver-optimal code is also locally receiver-optimal, so Theorem 3 can be considered as a corollary to the stronger Theorem 5.

---

[3]In Crawford and Sobel (1982), it is the uninformative equilibrium with a single partition class for the sender.

Local receiver-optimality is more easily verified than global receiver-optimality, because much fewer codes c' have to be considered as possible improvements for the receiver payoff according to Definition 4. A locally receiver-optimal code can be reached by iterating profitable changes of single codewords at a time. This simplifies the search for Nash codes.

To conclude this section, we consider the connection to *potential games* which also allow for iterative improvements in order to find a Nash equilibrium. As in Monderer and Shapley (1996, p. 127), consider a game in strategic form with finite player set N, and pure strategy set S_i and utility function u^i for each player i. Then the game has an (ordinal) *potential function* $P : \prod_{j \in N} S_j \to \mathbb{R}$ if for all $i \in N$ and $s^{-i} \in \prod_{j \neq i} S_j$ and $s^i, \hat{s}^i \in S_i$,

$$u^i(s^{-i}, \hat{s}^i) > u^i(s^{-i}, s^i) \quad \Longleftrightarrow \quad P(s^{-i}, \hat{s}^i) > P(s^{-i}, s^i). \tag{14}$$

The question is if in our game, the receiver's payoff is a potential function.[4] The following proposition gives an answer.

Proposition 6 *Consider the game with $M + 1$ players where for each state i in Ω, a separate agent i transmits a codeword $c(i)$ over the channel, which defines a function $c : \Omega \to X^n$, and where the receiver decodes each channel output with a decoding function d as before. Each agent receives the same payoff $U(c, d)$ as the original sender. Then*

(a) Any Nash equilibrium (c, d) of the $(M + 1)$-player game is a Nash equilibrium of the original two-player game, and vice versa.

(b) The receiver's expected payoff is a potential function for the $(M + 1)$-player game.

(c) The receiver's expected payoff is not necessarily a potential function for the original two-player game.

Proof. Every profile c of M strategies for the agents in the $(M + 1)$-player game can be seen as a sender strategy in the original game, and vice versa. To see (a), let (c, d) be a Nash equilibrium of the $(M + 1)$-player game. If there was a profitable deviation $\hat{c}$ from c for the sender in the two-player game as in (9), then there would also be a profitable deviation c' that changes only one codeword $c(i)$ as in (11), which is a profitable deviation for agent i, a contradiction. The "vice versa" part of (a) holds because any profitable deviation of a single agent is also a deviation for the sender in the original game.

Assertion (b) holds because for any i in Ω, (11) is, via (10), equivalent to (12).

To see (c), consider our example (7) with c and $\hat{c}$ given by the codebooks $0, 1$ and $1, 2$, respectively, and d decoding channel outputs $y = 0, 1, 2$ as states $0, 0, 1$, respectively. Then the payoffs to sender and receiver are

$$U(c, d) = q_0 U_0 (p(0|0) + p(1|0)) + q_1 U_1 p(2|1) = 1 \times (0.85 + 0.1) + 4 \times 0.25 = 1.95$$
$$V(c, d) = q_0 V_0 (p(0|0) + p(1|0)) + q_1 V_1 p(2|1) = 4 \times (0.85 + 0.1) + 1 \times 0.25 = 4.05$$
$$U(\hat{c}, d) = q_0 U_0 (p(0|1) + p(1|1)) + q_1 U_1 p(2|2) = 1 \times (0.1 + 0.65) + 4 \times 0.7 = 3.55$$
$$V(\hat{c}, d) = q_0 V_0 (p(0|1) + p(1|1)) + q_1 V_1 p(2|2) = 4 \times (0.1 + 0.65) + 1 \times 0.7 = 3.7$$

---

[4]We thank Rann Smorodinsky for raising this question.

which shows that (14) does not hold with u^i as sender payoff and P as receiver payoff, because these payoffs move in opposite directions when changing the sender's strategy from c to $\hat{c}$, for this d. $\qquad\square$

A global maximum of the potential function gives a Nash equilibrium of the potential game (Monderer and Shapley, 1996, Lemma 2.1). Hence, (a) and (b) of Proposition 6 imply that a maximum of the receiver payoff defines a Nash equilibrium, as stated in Theorem 3. It is also known that a "local" maximum of the potential function defines a Nash equilibrium (Monderer and Shapley, 1996, footnote 4). However, this does not imply our Theorem 5. The reason is that in a local maximum of the potential function, the function cannot be improved by unilaterally changing a single player's strategy. In contrast, in a locally receiver-optimal code, the receiver's payoff cannot be improved by changing a single codeword *together* with the receiver's best response. For example, the Nash code $1,2$ in (13) with best response partition $Y_0 = \{0,1,2\}$ is not locally receiver-optimal, but is a "local maximum" of the receiver payoff.

In a potential game, improvements of the potential function can be used for dynamics that lead to Nash equilibria. For our games, the study of such dynamics may be an interesting topic for future research.

4 Binary channels and monotonic decoding

The main result of this section concerns the important *binary channel* with $X = Y = \{0,1\}$. The two possible symbols 0 and 1 for a single use of the channel are called *bits*. The binary channel is the basic model for the transmission of digital data and of central theoretical and practical importance in information theory (see, for example, Cover and Thomas, 1991, or MacKay, 2003).

We assume that the channel errors $\varepsilon_0 = p(1|0)$ and $\varepsilon_1 = p(0|1)$ fulfill

$$\varepsilon_0 > 0, \qquad \varepsilon_1 > 0, \qquad \varepsilon_0 + \varepsilon_1 < 1, \tag{15}$$

where $\varepsilon_0 + \varepsilon_1 < 1$ is equivalent to either of the inequalities

$$1 - \varepsilon_0 > \varepsilon_1, \qquad 1 - \varepsilon_1 > \varepsilon_0. \tag{16}$$

These assert that a received bit 0 is more likely to have been sent as 0 (with probability $1 - \varepsilon_0$) than sent as bit 1 and received with error (with probability ε_1), and similarly that a received bit 1 is more likely to have been sent as 1 than received erroneously. It may still happen that bit 0, for example, is transmitted with higher probability incorrectly than correctly, for example if $\varepsilon_0 = 3/4$ and $\varepsilon_1 = 1/8$.

Condition (15) can be assumed with very little loss of generality. If $\varepsilon_0 = \varepsilon_1 = 0$ then the channel is error-free and every message can be decoded perfectly. If $\varepsilon_0 + \varepsilon_1 = 1$ then the channel output is independent of the input and no information can be transmitted. For $\varepsilon_0 + \varepsilon_1 > 1$ the signal is more likely to be inverted than not, so that one obtains (15) by exchanging 0 and 1 in Y.

Condition (15) does exclude the interesting case of a "Z-channel" that has only one-sided errors, that is, $\varepsilon_0 = 0$ or $\varepsilon_1 = 0$. We assume instead that this is modelled by vanishingly small error probabilities, in order to avoid case distinctions about channel outputs y in Y^n that cannot occur for some inputs x when $\varepsilon_0 = 0$ or $\varepsilon_1 = 0$. With (15), every channel output y has positive, although possibly very small, probability.

The binary channel is *symmetric* when $\varepsilon_0 = \varepsilon_1 = \varepsilon > 0$, where $\varepsilon < 1/2$ by (15).

The binary channel is used n times independently. A code $c : \Omega \to X^n$ for $X = \{0,1\}$ is also called a *binary code*. The main result of this section (Theorem 8 below) states that *any* binary code is a Nash code,[5] provided the decoding is *monotone*. This monotonicity condition concerns how the receiver resolves ties when a received channel output y can be decoded in more than one way.

We first consider an example of a binary code that shows that the equilibrium property may depend on how the receiver deals with ties. Assume that the channel is symmetric with error probability ε. Let $M = 4$, $n = 3$, and consider the codebook x^0, x^1, x^2, x^3 given by $000, 100, 010, 001$. All four states i have equal prior probabilities $q_i = 1/4$ and equal sender and receiver utilities $U_i = V_i = 1$. The sets Y_i in (3) are given by

$$
\begin{aligned}
Y_0 &= \{000\}, & Y_2 &= \{010, 011, 110, 111\}, \\
Y_1 &= \{100, 101, 110, 111\}, & Y_3 &= \{001, 011, 101, 111\}.
\end{aligned}
\tag{17}
$$

This shows that for any channel output y other than an original codeword x^i, there are ties between at least two states. For example, $110 \in Y_1 \cap Y_2$ because 110 is received with probability $\varepsilon(1-\varepsilon)^2$ for x^1 and x^2 as channel input. For $y = 111$, all three states $1, 2, 3$ are tied.
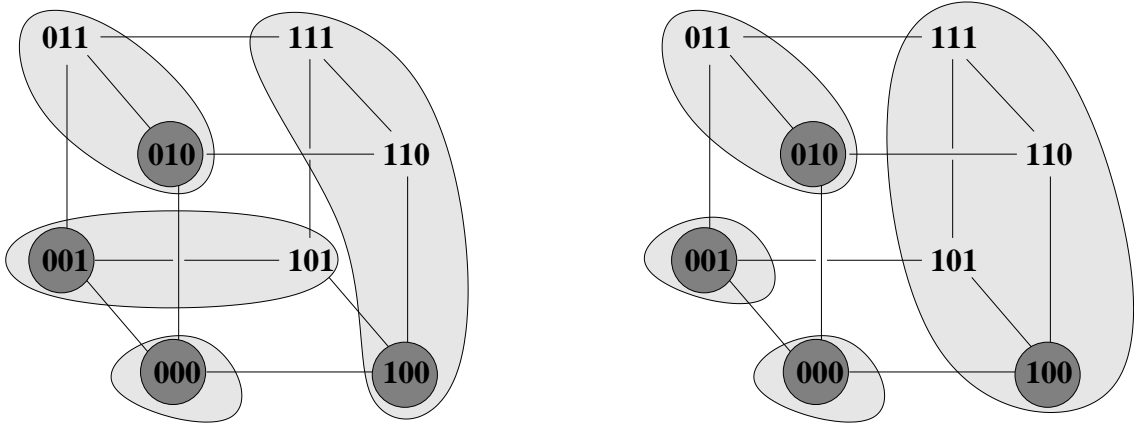

Figure 1: Binary code with four codewords 000, 100, 010, 001, with non-monotonic decoding (left) and monotonic decoding (right). The light-grey sets indicate how a channel output is decoded.

[5]Hernández, Urbano, and Vila (2010b) show that for a binary noisy channel, the decoding rule of "joint typicality" used in a standard proof of Shannon's channel coding theorem (Cover and Thomas, 1991, Section 8.7) may not define a Nash equilibrium.

Consider first the case that the receiver decodes the channel outputs $110, 011, 101$ as states $1, 2, 3$, respectively, that is, according to

$$d(110, 1) = 1, \quad d(011, 2) = 1, \quad d(101, 3) = 1. \tag{18}$$

We claim that this cannot be a Nash code, irrespective of the decoding probabilities $d(111, i)$ which can be positive for any $i = 1, 2, 3$ by (17). The situation is symmetric for $i = 1, 2, 3$, so assume that $d(111, i)$ is positive when $i = 1$; the case of a deterministic decoding where $d(111, 1) = 1$ is shown on the left in Figure 1. Then the receiver decodes y as state 1 with positive probability when y equals 100, 110, or 111. When $x^1 = 100$ is sent, these channel outputs are received with probabilities $(1 - \varepsilon)^3$, $\varepsilon(1 - \varepsilon)^2$, and $\varepsilon^2(1 - \varepsilon)$, respectively, so the sender payoff is

$$(1 - \varepsilon)^3 + \varepsilon(1 - \varepsilon)^2 + \varepsilon^2(1 - \varepsilon)d(111, 1)$$

in (5). Given this decoding, the sender can improve her payoff in state 1 by sending $x = 110$ rather than $x^1 = 100$ because then the probabilities of the channel outputs 100 and 110 are just exchanged, whereas the probability that output 111 is decoded as state 1 increases to $\varepsilon(1 - \varepsilon)^2 d(111, 1)$; that is, given this decoding, sending $x = 110$ is more likely to be decoded correctly as state 1 than sending $x^1 = 100$. This violates (6).

The problem with the decoding in (18) is that when the receiver is tied between states 1, 2, and 3 when the channel output is $y' = 111$, he decodes y' as state 1 with positive probability $d(111, 1)$, but when he is tied between even fewer states 1 and 3 when receiving $y = 101$, that decoding probability $d(101, 1)$ decreases to zero. This violates the following *monotonicity* condition.

Definition 7 *Consider a codebook with codewords x^i for $i \in \Omega$ and a decoding function d in* (2). *Then d is called* monotonic *if it is a best response decoding function with* (3) *and* (4), *and if for all $y, y' \in Y^n$ and states i,*

$$T = \{k \in \Omega \mid y \in Y_k\}, \quad T' = \{k \in \Omega \mid y' \in Y_k\}, \quad i \in T \subseteq T' \implies d(y, i) \geq d(y', i). \tag{19}$$

In (19), T is the set of tied states for channel output y, and T' is the set of tied states for channel output y', and both sets include state i. The condition states that the probability of decoding the channel output as state i can only decrease when the set of tied states increases.

We study the monotonicity condition in Definition 7 in more detail in the next section. We conclude with the main result of this section; its proof and some technical comments are given in the Appendix.

Theorem 8 *Every monotonically decoded binary code is a Nash code.*

5 Monotonic decoding functions

When is a decoding function monotonic? Suppose there is some fixed order on the set of states so that always the first tied state is chosen according to that order. In this section, we show that this is essentially the only way to break ties with a deterministic monotonic decoding function.

The monotonicity condition in Definition 7 implies

$$T = \{k \in \Omega \,|\, y \in Y_k\}, \quad T' = \{k \in \Omega \,|\, y' \in Y_k\}, \quad i \in T = T' \implies d(y,i) = d(y',i). \quad (20)$$

That is, the decoding probability $d(y,i)$ of state i may only depend on the set T of states that are tied with i, but not on the received channel output y. For that reason, we can define a monotonic decoding function also as a function $d(T,i)$ of the set T of best-response states,

$$d(T,i) := d(y,i) \quad \text{if } T = \{k \in \Omega \,|\, y \in Y_k\} \quad (21)$$

which is well defined by (20).

A natural example of a probabilistic monotonic decoding function is to break ties uniformly with $d(T,i) = 1/|T|$ for $i \in T$. A more general monotonic decoding function is $d(T,i) = w_i / \sum_{k \in T} w_k$ for $i \in T$ with a fixed positive weight w_k for each state k. There are many other probabilistic monotonic decoding functions. For example, if ties between three or more states are broken uniformly, then ties between only two states are decoded monotonically if the decoding probabilities for both tied states are at least $1/3$.

We will show that *deterministic* monotonic decoding functions are more restrictive. Consider again the example (18) with $d(111,1) = 1$ as shown on the left in Figure 1. (Note that this decoding is not monotonic but fulfills the weaker condition (20) which therefore does not suffice to guarantee a Nash code.)

The following decoding function, changed from (18) so that 101 is decoded as state 1, is monotonic,

$$d(110,1) = 1, \quad d(011,2) = 1 \quad d(101,1) = 1, \quad d(111,1) = 1, \quad (22)$$

shown in the right picture in Figure 1. This is a Nash code because all y in the set Y_1, see (17), are decoded as state 1; whichever x in Y_1 the sender decides to transmit instead of x^1, there is one y in Y_1 for which $p(y|x) = \varepsilon^2(1-\varepsilon)$, so that the payoff to the sender in (5) does not increase by changing from x^1 to x.

As the right picture in Figure 1 shows, the decoding function in (22) can be defined by the following condition: Consider a fixed linear order $\prec$ on Ω (in this case $0 \prec 1 \prec 2 \prec 3$) so that

$$d(T,i) = 1 \quad \Longleftrightarrow \quad i \in T \text{ and } \forall k \in T, \, k \neq i \,:\, i \prec k. \quad (23)$$

A *fixed-order* decoding function d fulfills (23) for some $\prec$. Such a decoding function is deterministic and clearly monotonic.

We want to show that any deterministic monotonic decoding function is a fixed-order decoding function. We have to make the additional assumption that the decoding function

$d(T, i)$ is *general* in the sense that it is defined for *any* nonempty set T of states, not only the sets T that occur as sets of tied states for some channel output y as in (21).

Without this assumption, we could add to the above example another state with codeword $x^4 = 111$ so that the "circular" decoding function in (18) is monotonic and gives a Nash code, but is clearly not a fixed-order decoding function. It is reasonable to require that a decoding function is defined generally and does not just coincidentally lead to a Nash code because certain ties do not occur (as argued above, with the decoding (18) we do not have a Nash code when ties have to be resolved for $y = 111$).

For general decoding functions, (19) translates to the requirement that for any $T, T' \subseteq \Omega$,

$$i \in T \subseteq T' \implies d(T, i) \geq d(T', i). \tag{24}$$

Proposition 9 *Every general deterministic monotonic decoding function is a fixed-order decoding function.*

Proof. Because the decoding function is deterministic, $d(T, i) \in \{0, 1\}$ for any nonempty set T of states. Define the following binary relation $\prec$ on Ω:

$$i \prec j \iff d(\{i, j\}, i) = 1.$$

Clearly, either $i \prec j$ or $j \prec i$ for any two states i, j. We claim that $\prec$ is transitive, that is, if $i \prec j$ and $j \prec k$, then $i \prec k$. Otherwise, there would be a "cycle" of distinct i, j, k with $i \prec j$ and $j \prec k$ and $k \prec i$. This is symmetric in i, j, k, so assume $d(\{i, j, k\}, i) = 1$ and therefore $d(\{i, j, k\}, j) = 0$ and $d(\{i, j, k\}, k) = 0$. However, with $T = \{i, k\}$ and $T' = \{i, j, k\}$ we have $d(T, i) = 0 < 1 = d(T', i)$, which contradicts (24).

So $\prec$ defines a linear order on Ω. We show that (23) holds, that is, for any nonempty set of states T' the state i so that $d(T', i) = 1$ is the state i that fulfills $i \prec k$ for all $k \in T'$. This holds trivially and by definition if T' has at most two elements, otherwise, if $k \prec i$ for some $k \in T'$, then we obtain with $T = \{i, k\}$ the same contradiction $d(T, i) = 0 < 1 = d(T', i)$ as before. So the decoded state is chosen according to the fixed order $\prec$ on Ω as claimed. $\square$

When the prior probabilities q_i or the receiver utilities V_i for the states i are *generic*, then Y_i in (3) is always a singleton, so no ties occur and decoding is deterministic. One can make any prior probabilities generic by perturbing them minimally so that ties are broken uniquely but decoding is otherwise unaffected. That is, if i and j are tied for some y because $q_i V_i p(y|x^i) = q_j V_j p(y|x^j)$, this tie is broken in favor of i by slightly increasing q_i, which will then always happen whenever i and j are tied originally. This induces a fixed-order decoding, where any linear order among the states can be chosen. Thus, Proposition 9 asserts that general deterministic monotonic decoding functions are those obtained by generic perturbation of the priors.

Finally, we observe that the above codebook $000, 100, 010, 001$ with decoding as in (22) defines a Nash code (and if priors are minimally perturbed so that $q_1 > q_2 > q_3$ there are no ties and decoding is unique), but this code is not locally optimal as in Theorem 5.

Namely, by changing the codeword 100 to 110, all possible channel outputs y differ in at most bit from one of the four codewords, which clearly improves the payoff to the receiver. So not all binary Nash codes are locally receiver-optimal.

Appendix: Proof of Theorem 8

We first give an outline of the proof of Theorem 8. We want to show that for each state i in Ω, the sender maximizes the probability of correct decoding by sending the prescribed codeword x^i, so that (6) holds for any $x \in X^n$. For any channel output y, comparing $p(y|x^i)$ and $p(y|x)$ is only affected by the bits where x^i and x differ, defined by the set D in (25) below. For these bits, the corresponding channel outputs are ordered according to how far they agree with x^i (and hence differ from x), indicated by the subset A of D in (30). The key property is that with increasing A, such a channel output is more likely to be decoded as state i, which is stated in (37) and the main technical challenge, proved with the help of the monotonicity assumption (19). The payoff in (5) is a multilinear function of the probabilities for receiving the individual output bits, see (32) and (38). By considering this multilinear expression for each of the transmitted bits in D and using the error inequalities (16), the monotonicity condition (37) translates to the inequality (6), as shown in (43).

Proof of Theorem 8. Conditions (3) and (4) state that the receiver uses a best response, so the equilibrium property holds on the receiver's side.

Let $i \in \Omega$ be the state chosen by nature. Let x in X^n be an arbitrary alternative message to the codeword x^i. We want to prove (6). Let S and D be the sets of bits in x and x^i that are the same and different, respectively, that is,

$$S = \{j \mid x_j = x_j^i,\ 1 \le j \le n\}, \qquad D = \{j \mid x_j \ne x_j^i,\ 1 \le j \le n\}. \tag{25}$$

For any sets Z and A and elements z_j in Z for $j \in A$ we write $z_A = (z_j)_{j \in A}$ and denote the set of these vectors z_A by Z^A.

For any $z \in \{0,1\}^n$ we write $z = (z_S, z_D)$, so that with (1)

$$p(y|z) = p(y_S|z_S) \cdot p(y_D|z_D). \tag{26}$$

In particular, by (25),

$$\begin{aligned} p(y|x^i) &= p(y_S|x_S^i) \cdot p(y_D|x_D^i) = p(y_S|x_S) \cdot p(y_D|x_D^i), \\ p(y|x) &= p(y_S|x_S) \cdot p(y_D|x_D). \end{aligned} \tag{27}$$

Fix $y_S \in Y^S$. We will show that

$$\sum_{y_D \in Y^D} p((y_S, y_D)|x^i)\, d((y_S, y_D), i) \ge \sum_{y_D \in Y^D} p((y_S, y_D)|x)\, d((y_S, y_D), i). \tag{28}$$

Because $y = (y_S, y_D)$ for $y \in Y^n$, summation of (28) over all $y_S \in Y^S$ then implies (6).

18

By (3), $y = (y_S, y_D) \in Y_i$ if and only if for all $k \in \Omega$,

$$q_i V_i \, p(y_S | x_S^i) \cdot p(y_D | x_D^i) \geq q_k V_k \, p(y_S | x_S^k) \cdot p(y_D | x_D^k). \tag{29}$$

If equality holds in (29), then $y \in Y_k$ and there is a tie between states i and k, which affects $d(y, i)$ where we will use (19).

It is useful to consider the channel outputs y_D (for the bits in D) according to how they agree with x_D^i. For $A \subseteq D$, let

$$y_D^A = (y_j^A)_{j \in D}, \qquad y_j^A = \begin{cases} x_j^i & \text{if } j \in A, \\ 1 - x_j^i & \text{if } j \in D - A. \end{cases} \tag{30}$$

Clearly, any y_D in Y^D can be written as $y_D = y_D^A$ for a unique subset A of D.

Let $A \subseteq D$ and $y_D = y_D^A \in Y^D$. For $l \in \Omega$, consider the sets

$$\begin{aligned} D_0^l &= \{ j \in D \mid x_j^l = 0 \}, & A_0^l &= \{ j \in D \mid y_j = x_j^l = 0 \}, \\ D_1^l &= \{ j \in D \mid x_j^l = 1 \}, & A_1^l &= \{ j \in D \mid y_j = x_j^l = 1 \}, \end{aligned} \tag{31}$$

so that $A = A_0^i \cup A_1^i$. Then according to (1),

$$p(y_D | x_D^l) = (1 - \varepsilon_0)^{|A_0^l|} \varepsilon_0^{|D_0^l - A_0^l|} (1 - \varepsilon_1)^{|A_1^l|} \varepsilon_1^{|D_1^l - A_1^l|}. \tag{32}$$

For $k \in \Omega$, let

$$Q_k(A) = p(y_D^A | x_D^i) - R_k \cdot p(y_D^A | x_D^k), \qquad R_k = \frac{q_k V_k \, p(y_S | x_S^k)}{q_i V_i \, p(y_S | x_S^i)}. \tag{33}$$

Then (29) is equivalent to $Q_k(A) \geq 0$ for all $k \in \Omega$.

Let $\text{sign}[t]$ for $t \in \mathbb{R}$ be the usual sign function defined by

$$\text{sign}[t] = \begin{cases} -1 & \text{if } t < 0, \\ 0 & \text{if } t = 0, \\ 1 & \text{if } t > 0. \end{cases}$$

Let $j \in D - A$, where we write $A \cup j$ instead of $A \cup \{j\}$. We will show

$$\text{sign}[Q_k(A \cup j)] \geq \text{sign}[Q_k(A)]. \tag{34}$$

Because $j \notin A = A_0^i \cup A_1^i$, either $j \in D_0^i - A_0^i$ or $j \in D_1^i - A_1^i$.

Consider the case $j \in D_0^i - A_0^i$, that is, $x_j^i = 0$ and $y_j^A = 1$ by (31). The change from A to $A \cup j$ means that $y_D^{A \cup j}$ is obtained from y_D^A by changing y_j from 1 to 0, so that the input bit x_j^i is now correctly transmitted (which happens with probability $1 - \varepsilon_0$) rather than incorrectly (probability ε_0). By (32), this means

$$p(y_D^{A \cup j} | x_D^i) = \frac{1 - \varepsilon_0}{\varepsilon_0} p(y_D^A | x_D^i). \tag{35}$$

19

Note that it is possible that $\varepsilon_0 > 1 - \varepsilon_0$, which means that the "more correct" channel output $y_D^{A \cup j}$ (relative to the input bits in x_D) is less likely than y_D^A; this is why we consider signs in (34) because $Q_k(A \cup j) \geq Q_k(A)$ is not generally true.

When comparing the output bit $y_j^A = 1$ with the input bit x_j^k from the codeword x^k, either $j \in D_0^k - A_0^k$, in which case (35) holds with k instead of i, and, by (33), $Q_k(A \cup j) = (1 - \varepsilon_0)/\varepsilon_0 \cdot Q_k(A)$, so that (34) holds with equality; or, alternatively, $j \in A_1^k$, that is, $x_j^k = 1$. Changing y_j^A from 1 to 0 to obtain $y_j^{A \cup j}$ implies that the input x_j^k is now transmitted with error, so that

$$p(y_j^{A \cup j}|x_D^k) = \frac{\varepsilon_1}{1 - \varepsilon_1} p(y_D^A|x_D^k).$$

Using (33) and (35), this means

$$Q_k(A \cup j) = \frac{1 - \varepsilon_0}{\varepsilon_0} \left[p(y_D^A|x_D^i) - R_k \cdot \frac{\varepsilon_0 \varepsilon_1}{(1 - \varepsilon_0)(1 - \varepsilon_1)} p(y_D^A|x_D^k) \right] \geq \frac{1 - \varepsilon_0}{\varepsilon_0} Q_k(A)$$

by (16). Again, (34) holds, where here the sign of $Q_k(A \cup j)$ relative to that of $Q_k(A)$ may strictly increase.

The case $j \in D_1^i - A_1^i$ where $x_j^i = 1$ and $y_j^A = 0$ is entirely analogous, by exchanging 0 and 1 (and thus ε_0 and ε_1) in the preceding reasoning. This shows (34).

For $A \subseteq D$, let

$$h_A = d((y_S, y_D^A), i). \tag{36}$$

We show that for $j \in D - A$,

$$h_{A \cup j} \geq h_A. \tag{37}$$

With $y = (y_S, y_D^{A \cup j})$ and $y' = (y_S, y_D^A)$, let T and T' be defined as in (19). We are going to show that $T \subseteq T'$. As observed after (33), $y' \in Y_i$ if and only if $Q_k(A) \geq 0$ for all $k \in \Omega$, and $y \in Y_i$ if and only if $Q_k(A \cup j) \geq 0$ for all $k \in \Omega$. If $Q_k(A) < 0$ for some $k \in \Omega$, then $h_A = 0$ by (4), and (37) holds trivially. So we can assume that $Q_k(A) \geq 0$ for all $k \in \Omega$ and therefore $Q_k(A \cup j) \geq 0$ for all $k \in \Omega$ by (34), that is, $i \in T$, and the sets of all states k that are tied with i are given by

$$T' = \{k \in \Omega \mid Q_k(A) = 0\}, \qquad T = \{k \in \Omega \mid Q_k(A \cup j) = 0\},$$

which implies that $T \subseteq T'$ by (34). Using the assumption (19) then implies (37) as claimed.

Consider now the function $f : [0,1]^D \to \mathbb{R}$ which for $z \in [0,1]^D$ is defined by

$$f(z_D) = \sum_{A \subseteq D} h_A \prod_{l \in A} z_l \prod_{l \in D - A} (1 - z_l), \tag{38}$$

which is the unique multilinear interpolation of the values h_A defined on the vertices $(1_A, 0_{D-A})$ of the unit cube $[0,1]^D$, where $(1_A, 0_{D-A})_j$ is 1 for $j \in A$ and 0 otherwise, with $f(1_A, 0_{D-A}) = h_A$ by (38).

The monotonicity (37) extends to the monotonicity of $f(z_j, z_{D-j})$ in each variable z_j, where we write $D - j$ for $D - \{j\}$ and $z_D = (z_j, z_{D-j})$, because by (38),

$$f(z_j, z_{D-j}) = \sum_{A \subseteq D-j} h_A \prod_{l \in A} z_l \prod_{l \in D-A-j} (1 - z_l) + z_j \cdot \sum_{A \subseteq D-j} (h_{A \cup j} - h_A) \prod_{l \in A} z_l \prod_{l \in D-A-j} (1 - z_l).$$

That is, because $h_{A \cup j} - h_A \geq 0$ by (37) and all products are nonnegative,

$$1 \geq z_j \geq z'_j \geq 0 \implies f(z_j, z_{D-j}) \geq f(z'_j, z_{D-j}) \qquad (j \in D). \tag{39}$$

Using (31), let

$$D_0 = D_0^i, \qquad D_1 = D_1^i, \qquad A_0 = A_0^i, \qquad A_1 = A_1^i \tag{40}$$

and define z_D^i and z_D in $[0,1]^D$ by

$$\begin{aligned}
z_j^i &= 1 - \varepsilon_0, & z_j &= \varepsilon_1 & \text{for } j \in D_0, \\
z_j^i &= 1 - \varepsilon_1, & z_j &= \varepsilon_0 & \text{for } j \in D_1.
\end{aligned} \tag{41}$$

Then $z_D^i > z_D$ in each component by (16). Using (39) inductively shows

$$f(z_D^i) \geq f(z_D). \tag{42}$$

The grand finale is to expand (38), using (31) and (40), to

$$f(z_D) = \sum_{A_0 \subseteq D_0,\, A_1 \subseteq D_1} h_{A_0 \cup A_1} \prod_{l \in A_0} z_l \prod_{l \in D_0 - A_0} (1 - z_l) \prod_{l \in A_1} z_l \prod_{l \in D_1 - A_1} (1 - z_l)$$

and to observe that by (41), (31), (32) for $l = i$, (36), (42), (25) and again (41) and (32) for $x_D^l = x_D$,

$$\sum_{A \subseteq D} p(y_D^A | x_D^i)\, d((y_S, y_D^A), i) = f(z_D^i) \geq f(z_D) = \sum_{A \subseteq D} p(y_D^A | x_D)\, d((y_S, y_D^A), i). \tag{43}$$

Multiplying this inequality by $p(y_S | x_S)$ on both sides and using (27) then gives (28) (with y_D written as y_D^A), which was to be shown. $\qquad\square$

We conclude with two small remarks:

First, the equilibrium condition for the sender does not necessarily hold strictly; if all codewords have the same bit in one particular position, then that bit is ignored by the receiver and correspondingly can be altered by the sender in any codeword.

Second, the preceding proof works also if for each of the n times that the channel is used independently, different error probabilities apply, as long as these are common knowledge. We have not made this assumption to avoid further notational complications.

References

Blume, A., and O. J. Board (2009), Intentional vagueness. Mimeo, University of Pittsburgh.

Blume, A., O. J. Board, and K. Kawamura (2007), Noisy talk. Theoretical Economics 2, 395–440.

Cover, T. M., and J. A. Thomas (1991), Elements of Information Theory. Wiley, New York.

Crawford, V., and J. Sobel (1982), Strategic information transmission. Econometrica 50, 1431–1451.

Cushing, S. (1994), Fatal Words: Communication Clashes and Aircraft Crashes. University of Chicago Press, Chicago.

Gallager, R. G. (1968), Information Theory and Reliable Communication. Wiley, New York.

Glazer, J., and A. Rubinstein (2004), On the optimal rules of persuasion. Econometrica 72, 1715–1736.

Glazer, J., and A. Rubinstein (2006), A study in the pragmatics of persuasion: A game theoretical approach. Theoretical Economics 1, 395–410.

Hernández, P., A. Urbano, and J. E. Vila (2010a), Pragmatic languages with universal grammars. Discussion Papers in Economic Behaviour ERI-CES 01/2010, University of Valencia.

Hernández, P., A. Urbano, and J. E. Vila (2010b), Nash equilibrium and information transmission coding and decoding rules. Discussion Papers in Economic Behaviour ERI-CES 09/2010, University of Valencia.

Jäger, G., L. Koch-Metzger, and F. Riedel (2011), Voronoi languages: Equilibria in cheap talk games with high-dimensional types and few signals. Games and Economic Behavior 73, 517–537.

Kamenica, E., and M. Gentzkow (2011), Bayesian persuasion. American Economic Review 101, 2590–2615.

Koessler, F. (2001), Common knowledge and consensus with noisy communication. Mathematical Social Sciences 42, 139–159.

Kreps, D. M., and J. Sobel (1994), Signalling. In: R. J. Aumann and S. Hart, eds., Handbook of Game Theory with Economic Applications, Vol. 2, Elsevier, Amsterdam, 849–867.

Lewis, D. (1969), Convention: A Philosophical Study. Harvard University Press, Cambridge, MA.

Lipman, B. (2009), Why is language vague? Mimeo, Boston University.

MacKay, D. J. C. (2003), Information Theory, Inference, and Learning Algorithms. Cambridge University Press, Cambridge, UK.

Monderer, D., and L. S. Shapley (1996), Potential games. Games and Economic Behavior 14, 124–143.

Myerson, R. B. (1994), Communication, correlated equilibria and incentive compatibility. In: R. J. Aumann and S. Hart, eds., Handbook of Game Theory with Economic Applications, Vol. 2, Elsevier, Amsterdam, 827–847.

Nowak, M., and D. Krakauer (1999), The evolution of language. Proc. Nat. Acad. Sci. USA 96, 8028–8033.

Pawlowitsch, C. (2008), Why evolution does not always lead to an optimal signaling system. Games and Economic Behavior 63, 203–226.

Shannon, C. E. (1948), A mathematical theory of communication. Bell System Technical Journal 27, 379–423; 623–656.

Sobel, J. (2010), Giving and receiving advice. Mimeo, University of California at San Diego, presented at the Econometric Society 10th World Congress.

Spence, M. (1973), Job market signaling. The Quarterly Journal of Economics 87, 355–374.

Wärneryd, K. (1993), Cheap talk, coordination and evolutionary stability. Games and Economic Behavior 5, 532–546.