

Chapter 2

Pair Processes: Channels, Codes, and Couplings

Abstract We have considered a random process or source $\{X_n\}$ as a sequence of random entities, where the object produced at each time could be quite general, e.g., a random variable, vector, or waveform. Hence sequences of pairs of random objects such as $\{X_n, Y_n\}$ are included in the general framework. We now focus on the possible interrelations between the two components of such a pair process. First consider the situation where we begin with one source, say $\{X_n\}$, called the *input* and use either a random or a deterministic mapping of the input sequence $\{X_n\}$ to form an output sequence $\{Y_n\}$. We generally refer to the mapping as a *channel* if it is random and a *code* if it is deterministic. Hence a code is a special case of a channel and results for channels will immediately imply corresponding results for codes. The initial point of interest will be conditions on the structure of the channel under which the resulting pair process $\{X_n, Y_n\}$ will inherit stationarity and ergodic properties from the original source $\{X_n\}$. We will also be interested in the behavior resulting when the output of one channel serves as the input to another, that is, when we form a new channel as a cascade of other channels. Such cascades yield models of a communication system which typically has a code mapping (called the *encoder*) followed by a channel followed by another code mapping (called the *decoder*). Lastly, pair processes arise naturally in other situations, including *coupling* two separate processes by constructing a joint distribution. This chapter develops the context for the development in future chapters of the properties of information and entropy arising in pair processes.

2.1 Pair Processes

A common object throughout this book and the focus of this chapter is the idea of a pair process. The notation will vary somewhat depend-

ing on the specific application, but basically a pair process is a random process with two components, e.g., a sequence of random variables $\{(X_n, Y_n); n \in \mathbb{T}\}$ with alphabets A and B and process distribution p on $(A^{\mathbb{T}} \times B^{\mathbb{T}}, \mathcal{B}(A^{\mathbb{T}} \times B^{\mathbb{T}}))$. When we wish to emphasize the names of the separate component random variables and processes, we will often write A_X for A and A_Y for B and p_{XY} for p . The process X or $\{X_n\}$ will often have the interpretation of being the input of a code or channel or a cascade of such operations and Y or $\{Y_n\}$ the output. A pair process induces two “marginal” process $\{X_n\}$ with process distribution, say μ , and $\{Y_n\}$, with process distribution η . When we wish to emphasize the random variables we might write p_X or μ_X instead of μ and p_Y or μ_Y or η_Y instead of η . All of these notations have their uses, and the added subscripts often help sort out which random process or variables are important. Often we will use $\hat{X}_n$ as the second component instead of Y when it is viewed as an approximation to the first component X_n .

2.2 Channels

A channel converts one information source – typically called the *input* to the channel – into another – called the *output*. In general the operation is random and is specified by a conditional probability measure of output sequences given an input sequence. The combination of an input distribution with the channel yields a pair process, a process with an input component and an output component. If the channel is deterministic rather than random, the operation is called a *code*. In this section the basic definitions of channels and codes are introduced.

A fundamental nuisance in the development of channels and codes is the notion of time. So far we have considered pair processes where at each unit of time, one random object is produced for each coordinate of the pair. In the channel or code example, this corresponds to one output for every input. Interesting communication systems do not always easily fit into this framework, and this can cause serious problems in notation and in the interpretation and development of results. For example, suppose that an input source consists of a sequence of real numbers and let T denote the time shift on the real sequence space. Suppose that the output source consists of a binary sequence and let S denote its shift. Suppose also that the channel is such that for each real number in, three binary symbols are produced. This fits our usual framework if we consider each output variable to consist of a binary three-tuple since then there is one output vector for each input symbol. One must be careful, however, when considering the stationarity of such a system. Do we consider the output process to be physically stationary if it is stationary with respect to S or with respect to S^3 ? The former might make more

sense if we are looking at the output alone, the latter if we are looking at the output in relation to the input. How do we define stationarity for the pair process? Given two sequence spaces, we might first construct a shift on the pair sequence space as simply the cartesian product of the shifts, e.g., given an input sequence x and an output sequence y define a shift T^* by $T^*(x, y) = (Tx, Sy)$. While this might seem natural given only the pair random process $\{X_n, Y_n\}$, it is not natural in the physical context that one symbol of X yields three symbols of Y . In other words, the two shifts do not correspond to the same amount of *time*. Here the more physically meaningful shift on the pair space would be $T'(x, y) = (Tx, S^3y)$ and the more physically meaningful questions on stationarity and ergodicity relate to T' and not to T^* . The problem becomes even more complicated when channels or codes produce a varying number of output symbols for each input symbol, where the number of symbols depends on the input sequence. Such variable rate codes arise often in practice, especially for noiseless coding applications such as Huffman, Lempel-Ziv, and arithmetic codes. While we will not treat such variable rate systems in any detail, they point out the difficulty that can arise associating the mathematical shift operation with physical time when we are considering cartesian products of spaces, each having their own shift.

There is no easy way to solve this problem notationally. We adopt the following view as a compromise which is usually adequate for fixed-rate systems. We will be most interested in pair processes that are stationary in the physical sense, that is, whose statistics are not changed when both are shifted by an equal amount of *physical* time. This is the same as stationarity with respect to the product shift if the two shifts correspond to equal amounts of physical time. Hence for simplicity we will usually focus on this case. More general cases will be introduced when appropriate to point out their form and how they can be put into the matching shift structure by considering groups of symbols and different shifts. This will necessitate occasional discussions about what is meant by stationarity or ergodicity for a particular system.

The mathematical generalization of Shannon's original notions of sources, codes, and channels are due to Khinchine [87] [88]. Khinchine's results characterizing stationarity and ergodicity of channels were corrected and developed by Adler [2].

Say we are given a source $[A, X, \mu]$, that is, a sequence of A -valued random variables $\{X_n; n \in \mathbb{T}\}$ defined on a common probability space $(\Omega, \mathcal{F}, P)$ having a process distribution μ defined on the measurable sequence space $(B^{\mathbb{T}}, \mathcal{B}_A^{\mathbb{T}})$. We shall let $X = \{X_n; n \in \mathbb{T}\}$ denote the sequence-valued random variable, that is, the random variable taking values in $A^{\mathbb{T}}$ according to the distribution μ . Let B be another alphabet with a corresponding measurable sequence space $(A^{\mathbb{T}}, \mathcal{B}_B^{\mathbb{T}})$. We assume as usual that A and B are standard and hence so are their sequence

spaces and cartesian products. A *channel* $[A, \nu, B]$ with input alphabet A and output alphabet B (we denote the channel simply by ν when these alphabets are clear from context) is a family of probability measures $\{\nu_x; x \in A^{\mathbb{T}}\}$ on $(B^{\mathbb{T}}, \mathcal{B}_B^{\mathbb{T}})$ (the output sequence space) such that for every output event $F \in \mathcal{B}_B^{\mathbb{T}}$ $\nu_x(F)$ is a measurable function of x . This measurability requirement ensures that the set function p specified on the joint input/output space $(A^{\mathbb{T}} \times B^{\mathbb{T}}, \mathcal{B}_A^{\mathbb{T}} \times \mathcal{B}_B^{\mathbb{T}})$ by its values on rectangles as

$$p(G \times F) = \int_G d\mu(x) \nu_x(F); F \in \mathcal{B}_B^{\mathbb{T}}, G \in \mathcal{B}_A^{\mathbb{T}},$$

is well defined. The set function p is nonnegative, normalized, and countably additive on the field generated by the rectangles $G \times F$, $G \in \mathcal{B}_A^{\mathbb{T}}$, $F \in \mathcal{B}_B^{\mathbb{T}}$. Thus p extends to a probability measure on the joint input/output space, which is sometimes called the *hookup* of the source μ and channel ν . We will often denote this joint measure by $\mu\nu$. The corresponding sequences of random variables are called the *input/output process*.

Thus a channel is a probability measure on the output sequence space for each input sequence such that a joint input/output probability measure is well-defined. The above equation shows that a channel is simply a regular conditional probability, in particular,

$$\nu_x(F) = p((x, y) : y \in F | x); F \in \mathcal{B}_B^{\mathbb{T}}, x \in A^{\mathbb{T}}.$$

We can relate a channel to the notation used previously for conditional distributions by using the sequence-valued random variables $X = \{X_n; n \in \mathbb{T}\}$ and $Y = \{Y_n; n \in \mathbb{T}\}$:

$$\nu_x(F) = P_{Y|X}(F|x). \quad (2.1)$$

Eq. (1.28) then provides the probability of an arbitrary input/output event:

$$p(F) = \int d\mu(x) \nu_x(F_x),$$

where $F_x = \{y : (x, y) \in F\}$ is the *section* of F at x .

If we start with a hookup p , then we can obtain the input distribution μ as

$$\mu(F) = p(F \times B^{\mathbb{T}}); F \in \mathcal{B}_A^{\mathbb{T}}.$$

Similarly we can obtain the output distribution, say η , via

$$\eta(F) = p(A^{\mathbb{T}} \times F); F \in \mathcal{B}_B^{\mathbb{T}}.$$

Suppose one now starts with a pair process distribution p and hence also with the induced source distribution μ . Does there exist a channel ν

for which $p = \mu\nu$? The answer is yes since the spaces are standard. One can always define the conditional probability $\nu_x(F) = P(F \times A^\mathbb{T} | X = x)$ for all input sequences x , but this need not possess a regular version, that is, be a probability measure for all x , in the case of arbitrary alphabets. If the alphabets are standard, however, we have seen that a regular conditional probability measure always exists.

2.3 Stationarity Properties of Channels

We now define a variety of stationarity properties for channels that are related to, but not the same as, those for sources. The motivation behind the various definitions is that stationarity properties of channels coupled with those of sources should imply stationarity properties for the resulting source-channel hookups.

The classical definition of a stationary channel is the following: Suppose that we have a channel $[A, \nu, B]$ and suppose that T_A and T_B are the shifts on the input sequence space and output sequence space, respectively. The channel is *stationary* with respect to T_A and T_B or (T_A, T_B) -stationary if

$$\nu_x(T_B^{-1}F) = \nu_{T_A x}(F), x \in A^\mathbb{T}, F \in \mathcal{B}_B^\mathbb{T}. \quad (2.2)$$

If the transformations are clear from context then we simply say that the channel is stationary. Intuitively, a right shift of an output event yields the same probability as the left shift of an input event. The different shifts are required because in general only $T_A x$ and not $T_A^{-1}x$ exists since the shift may not be invertible and in general only $T_B^{-1}F$ and not $T_B F$ exists for the same reason. If the shifts are invertible, e.g., the processes are two-sided, then the definition is equivalent to

$$\nu_{T_A x}(T_B F) = \nu_{T_A^{-1}x}(T_B^{-1}F) = \nu_x(F), \text{ all } x \in A^\mathbb{T}, F \in \mathcal{B}_B^\mathbb{T} \quad (2.3)$$

that is, shifting the input sequence and output event in the same direction does not change the probability.

The fundamental importance of the stationarity of a channel is contained in the following lemma.

Lemma 2.1. *If a source $[A, \mu]$, stationary with respect to T_A , is connected to channel $[A, \nu, B]$, stationary with respect to T_A and T_B , then the resulting hookup $\mu\nu$ is stationary with respect to the cartesian product shift $T = T_{A \times B} = T_A \times T_B$ defined by $T(x, y) = (T_A x, T_B y)$.*

Proof: We have that

$$\mu\nu(T^{-1}F) = \int d\mu(x) \nu_x((T^{-1}F)_x).$$

Now

$$\begin{aligned}(T^{-1}F)_x &= \{\mathcal{Y} : T(x, \mathcal{Y}) \in F\} = \{\mathcal{Y} : (T_A x, T_B \mathcal{Y}) \in F\} \\ &= \{\mathcal{Y} : T_B \mathcal{Y} \in F_{T_A x}\} = T_B^{-1} F_{T_A x}\end{aligned}$$

and hence

$$\mu\nu(T^{-1}F) = \int d\mu(x) \nu_x(T_B^{-1}F_{T_A x}).$$

Since the channel is stationary, however, this becomes

$$\mu\nu(T^{-1}F) = \int d\mu(x) \nu_{T_A x}(F_{T_A x}) = \int d\mu T_A^{-1}(x) \nu_x(F_x),$$

where we have used the change of variables formula. Since μ is stationary, however, the right hand side is

$$\int d\mu(x) \nu_x(F),$$

which proves the lemma. $\square$

Suppose next that we are told that a hookup $\mu\nu$ is stationary. Does it then follow that the source μ and channel ν are necessarily stationary? The source must be since

$$\mu(T_A^{-1}F) = \mu\nu((T_A \times T_B)^{-1}(F \times B^{\mathbb{T}})) = \mu\nu(F \times B^{\mathbb{T}}) = \mu(F).$$

The channel need not be stationary, however, since, for example, the stationarity could be violated on a set of μ measure 0 without affecting the proof of the above lemma. This suggests a somewhat weaker notion of stationarity which is more directly related to the stationarity of the hookup. We say that a channel $[A, \nu, B]$ is *stationary with respect to a source* $[A, \mu]$ if $\mu\nu$ is stationary. We also state that a channel is stationary μ -a.e. if it satisfies (2.2) for all x in a set of μ -probability one. If a channel is stationary μ -a.e. and μ is stationary, then the channel is also stationary with respect to μ . Clearly a stationary channel is stationary with respect to all stationary sources. The reason for this more general view is that we wish to extend the definition of stationary channels to asymptotically mean stationary channels. The general definition extends; the classical definition of stationary channels does not.

Observe that the various definitions of stationarity of channels immediately extend to block shifts since they hold for any shifts defined on the input and output sequence spaces, e.g., a channel stationary with respect to T_A^N and T_B^K could be a reasonable model for a channel or code that puts out K symbols from an alphabet B every time it takes in N symbols from an alphabet A . We shorten the name (T_A^N, T_B^K) -stationary

to (N, K) -stationary channel in this case. A stationary channel (without modifiers) is simply a $(1, 1)$ -stationary channel in this sense.

The most general notion of stationarity that we are interested in is that of asymptotic mean stationarity. We define a channel $[A, \nu, B]$ to be *asymptotically mean stationary* or *AMS* for a source $[A, \mu]$ with respect to T_A and T_B if the hookup $\mu\nu$ is AMS with respect to the product shift $T_A \times T_B$. As in the stationary case, an immediate necessary condition is that the input source be AMS with respect to T_A . A channel will be said to be (T_A, T_B) -AMS if the hookup is (T_A, T_B) -AMS for all T_A -AMS sources.

The following lemma shows that an AMS channel is indeed a generalization of the idea of a stationary channel and that the stationary mean of a hookup of an AMS source to a stationary channel is simply the hookup of the stationary mean of the source to the channel.

Lemma 2.2. *Suppose that ν is (T_A, T_B) -stationary and that μ is AMS with respect to T_A . Let $\bar{\mu}$ denote the stationary mean of μ and observe that $\bar{\mu}\nu$ is stationary. Then the hookup $\mu\nu$ is AMS with stationary mean*

$$\overline{\mu\nu} = \bar{\mu}\nu.$$

Thus, in particular, ν is an AMS channel.

Proof: We have that

$$\begin{aligned} (T^{-i}F)_x &= \{\gamma : (x, \gamma) \in T^{-i}F\} = \{\gamma : T^i(x, \gamma) \in F\} \\ &= \{\gamma : (T_A^i x, T_B^i \gamma) \in F\} = \{\gamma : T_B^i \gamma \in F_{T_A^i x}\} \\ &= T_B^{-i} F_{T_A^i x} \end{aligned}$$

and therefore since ν is stationary

$$\begin{aligned} \mu\nu(T^{-i}F) &= \int d\mu(x) \nu_x(T_B^{-i} F_{T_A^i x}) \\ &= \int d\mu(x) \nu_{T_A^i x}(F_{T_A^i x}) = \int d\mu T_A^{-i}(x) \nu_x(F). \end{aligned}$$

Therefore

$$\begin{aligned} \frac{1}{n} \sum_{i=0}^{n-1} \mu\nu(T^{-i}F) &= \frac{1}{n} \sum_{i=0}^{n-1} \int d\mu T_A^{-i}(x) \nu_x(F) \\ &\xrightarrow{n \rightarrow \infty} \int d\bar{\mu}(x) \nu_x(F) = \bar{\mu}\nu(F) \end{aligned}$$

from Lemma 6.5.1 of [55] or Lemma 7.9 if [58]. This proves that $\mu\nu$ is AMS and that the stationary mean is $\bar{\mu}\nu$. $\square$

A final property crucial to quantifying the behavior of random processes is that of ergodicity. Hence we define a (stationary, AMS) channel

ν to be ergodic with respect to (T_A, T_B) if it has the property that whenever a (stationary, AMS) ergodic source (with respect to T_A) is connected to the channel, the overall input/output process is (stationary, AMS) ergodic. The following modification of Lemma 6.7.4 of [55] or Lemma 7.15 of [58] is the principal tool for proving a channel to be ergodic.

Lemma 2.3. *An AMS (stationary) channel $[A, \nu, B]$ is ergodic if for all AMS (stationary) sources μ and all sets of the form $\bar{F} = F_A \times F_B$, $\bar{G} = G_A \times G_B$ for rectangles $F_A, G_A \in \mathcal{B}_A^\infty$ and $F_B, G_B \in \mathcal{B}_B^\infty$ we have that for $p = \mu\nu$*

$$\lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=0}^{n-1} p(T_{A \times B}^{-i} \bar{F} \cap \bar{G}) = \bar{p}(\bar{F})p(\bar{G}), \quad (2.4)$$

where $\bar{p}$ is the stationary mean of p (p if p is already stationary).

Proof: The proof parallels that of Lemma 6.7.4 of [55] or Lemma 7.15 of [58]. The result does not follow immediately from that lemma since the collection of given sets does not itself form a field. Arbitrary events $F, G \in \mathcal{B}_{A \times B}^\infty$ can be approximated arbitrarily closely by events in the field generated by the above rectangles and hence given $\epsilon > 0$ we can find finite disjoint rectangles of the given form F_i, G_i , $i = 1, \dots, L$ such that if $F_0 = \bigcup_{i=1}^L F_i$ and $G_0 = \bigcup_{i=1}^L G_i$, then $p(F \Delta F_0)$, $p(G \Delta G_0)$, $\bar{p}(F \Delta F_0)$, and $\bar{p}(G \Delta G_0)$ are all less than ϵ . Then

$$\begin{aligned} & \left| \frac{1}{n} \sum_{k=0}^{n-1} p(T^{-k} F \cap G) - \bar{p}(F)p(G) \right| \leq \\ & \left| \frac{1}{n} \sum_{k=0}^{n-1} p(T^{-k} F \cap G) - \frac{1}{n} \sum_{k=0}^{n-1} p(T^{-k} F_0 \cap G_0) \right| + \\ & \left| \frac{1}{n} \sum_{k=0}^{n-1} p(T^{-k} F_0 \cap G_0) - \bar{p}(F_0)p(G_0) \right| + |\bar{p}(F_0)p(G_0) - \bar{p}(F)p(G)|. \end{aligned}$$

Exactly as in Lemma 6.7.4 of [55], the rightmost term is bound above by 2ϵ and the first term on the left goes to zero as $n \rightarrow \infty$. The middle term is the absolute magnitude of

$$\begin{aligned} & \frac{1}{n} \sum_{k=0}^{n-1} p(T^{-k} \bigcup_i F_i \cap \bigcup_j G_j) - \bar{p}(\bigcup_i F_i)p(\bigcup_j G_j) = \\ & \sum_{i,j} \left(\frac{1}{n} \sum_{k=0}^{n-1} p(T^{-k} F_i \cap G_j) - \bar{p}(F_i)p(G_j) \right). \end{aligned}$$

Each term in the finite sum converges to 0 by assumption. Thus p is ergodic from Lemma 6.7.4 of [55] or Lemma 7.15 of [58]. $\square$

Because of the specific class of sets chosen, the above lemma considered separate sets for shifting and remaining fixed, unlike using the same set for both purposes as in Lemma 6.7.4 of [55] or Lemma 7.15 of [58]. This was required so that the cross products in the final sum considered would converge accordingly.

2.4 Extremes: Noiseless and Completely Random Channels

The first two examples of channels are the simplest, the first doing nothing to the input but reproducing it perfectly and the second being useless (at least for communication purposes) since the output is random and independent of the input. Both extremes provide simple examples of the properties of channels, and the completely random example will reappear when applying channel structure ideas to sources.

Noiseless Channels

A channel $[A, \nu, B]$ is said to be *noiseless* if $A = B$ and

$$\nu_x(F) = \begin{cases} 1 & x \in F \\ 0 & x \notin F \end{cases}$$

that is, with probability one the channel puts out what goes in, it acts as an ideal wire. In engineering terms, it is discrete-time linear system with impulse response equal to an impulse.

A noiseless channel is clearly stationary and ergodic.

Completely Random Channels

Suppose that η is a probability measure on the output space $(B^{\mathbb{T}}, \mathcal{B}_B^{\mathbb{T}})$ and define a channel

$$\nu_x(F) = \eta(F), F \in \mathcal{B}_B^{\mathbb{T}}, x \in A^{\mathbb{T}}.$$

Then it is easy to see that the input/output measure satisfies

$$p(G \times F) = \eta(F)\mu(G); F \in \mathcal{B}_B^{\mathbb{T}}, G \in \mathcal{B}_A^{\mathbb{T}},$$

and hence the input/output measure is a product measure and the input and output sequences are therefore independent of each other. This

channel is called a *completely random channel* or *product channel* because the output is independent of the input.

This channel is quite useless because the output tells us nothing of the input. The completely random channel is stationary (AMS) if the measure η is stationary (AMS). Perhaps surprisingly, such a channel need not be ergodic even if η is ergodic since the product of two stationary and ergodic sources need not be ergodic. (See, e.g., [22].) We shall later see that if η is also assumed to be weakly mixing, then the resulting channel is ergodic.

A generalization of the noiseless channel that is of much greater interest is the deterministic channel. Here the channel is not random, but the output is formed by a general mapping of the input rather than being the input itself.

2.5 Deterministic Channels and Sequence Coders

A channel $[A, \nu, B]$ is said to be *deterministic* if each input string x is mapped into an output string $f(x)$ by a measurable mapping $f : A^{\mathbb{T}} \rightarrow B^{\mathbb{T}}$. The conditional probability defining the channel is

$$\nu_x(G) = \begin{cases} 1 & f(x) \in G \\ 0 & f(x) \notin G. \end{cases}$$

Note that such a channel can also be written as

$$\nu_x(G) = 1_{f^{-1}(G)}(x).$$

A *sequence coder* is a deterministic channel, that is, a measurable mapping from one sequence space into another. It is easy to see that for a deterministic code the hookup is specified by

$$p(F \times G) = \mu(F \cap f^{-1}(G))$$

and the output process has distribution

$$\eta(G) = \mu(f^{-1}(G)).$$

A sequence coder is said to be (T_A, T_B) -stationary (or just *stationary*) or (T_A^N, T_B^K) -stationary (or just (N, K) -stationary) if the corresponding channel is. Thus a sequence coder f is stationary if and only if $f(T_A x) = T_B f(x)$ and it is (N, K) -stationary if and only if $f(T_A^N x) = T_B^K f(x)$.

Lemma 2.4. *A stationary deterministic channel is ergodic.*

Proof: From Lemma 2.3 it suffices to show that

$$\lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=0}^{n-1} p(T_{A \times B}^{-i} F \cap G) = p(F)P(G)$$

for all rectangles of the form $F = F_A \times F_B$, $F_A \in \mathcal{B}_B^{\mathbb{T}}$, $F_B \in \mathcal{B}_A^{\mathbb{T}}$ and $G = G_A \times G_B$. Then

$$\begin{aligned} p(T_{A \times B}^{-i} F \cap G) &= p((T_A^{-i} F_A \cap G_A) \times (T_B^{-i} F_B \cap G_B)) \\ &= \mu((T_A^{-i} F_A \cap G_A) \cap f^{-1}(T_B^{-i} F_B \cap G_B)). \end{aligned}$$

Since f is stationary and since inverse images preserve set theoretic operations,

$$f^{-1}(T_B^{-i} F_B \cap G_B) = T_A^{-i} f^{-1}(F_B) \cap f^{-1}(G_B)$$

and hence

$$\begin{aligned} \frac{1}{n} \sum_{i=0}^{n-1} p(T_{A \times B}^{-i} F \cap G) &= \frac{1}{n} \sum_{i=0}^{n-1} \mu(T_A^{-i}(F_A \cap f^{-1}(F_B)) \cap G_A \cap f^{-1}(G_B)) \\ &\xrightarrow{n \rightarrow \infty} \mu(F_A \cap f^{-1}(F_B)) \mu(G_A \cap f^{-1}(G_B)) \\ &= p(F_A \times F_B) p(G_A \times G_B) \end{aligned}$$

since μ is ergodic. This means that the rectangles meet the required condition. Some algebra then will show that finite unions of disjoint sets meeting the conditions also meet the conditions and that complements of sets meeting the conditions also meet them. This implies from the good sets principle (see, for example, p. 14 of [55] or p. 50 in [58]) that the field generated by the rectangles also meets the condition and hence the lemma is proved. $\square$

2.6 Stationary and Sliding-Block Codes

A stationary deterministic channel is also called a *stationary code*, so it follows that the output of a stationary code with a stationary input process is also stationary. A stationary code has a simple and useful structure. Suppose one has a mapping $f : A^{\mathbb{T}} \rightarrow B$, that is, a mapping that maps an input sequence into a single output symbol. We can define a complete output sequence y corresponding to an input sequence x by

$$y_n = f(T_A^n x); n \in \mathbb{T}, \quad (2.5)$$

that is, we produce an output, then shift or slide the input sequence by one time unit, and then we produce another output using the same function, and so on. A mapping of this form is called a *sliding-block code*

because it produces outputs by successively sliding an infinite-length input sequence and each time using a fixed mapping to produce the output. The sequence-to-symbol mapping implies a sequence coder, say $\bar{f}$, defined by $\bar{f}(x) = \{f(T_A^n x); n \in \mathbb{T}\}$. Furthermore, $\bar{f}(T_A x) = T_B \bar{f}(x)$, that is, a sliding-block code induces a stationary sequence coder. Conversely, any stationary sequence coder $\bar{f}$ induces a sliding-block code f for which (2.5) holds by the simple identification $f(x) = (\bar{f}(x))_0$, the output at time 0 of the sequence coder. Thus the ideas of stationary sequence coders mapping sequences into sequences and sliding-block codes mapping sequences into letters by sliding the input sequence are equivalent. We can similarly define an (N, K) -sliding-block code which is a mapping $f : A^{\mathbb{T}} \rightarrow B^K$ which forms an output sequence y from an input sequence x via the construction

$$y_{nK}^K = f(T_A^{Nn} x).$$

By a similar argument, (N, K) -sliding-block coders are equivalent to (N, K) -stationary sequence coders. When dealing with sliding-block codes we will usually assume for simplicity that K is 1. This involves no loss in generality since it can be made true by redefining the output alphabet.

The following stationarity property of sliding-block codes follows from the properties for stationary channels, but the proof is given for completeness.

Lemma 2.5. *If f is a stationary coding of an AMS process, then the process $\{f_n = fT^n\}$ is also AMS. If the input process is ergodic, then so is $\{f_n\}$.*

Proof: Suppose that the input process has alphabet A_X and distribution P and that the measurement f has alphabet A_f . Define the sequence mapping $\bar{f} : A_X^{\infty} \rightarrow A_f^{\infty}$ by $\bar{f}(x) = \{f_n(x); n \in \mathbb{T}\}$, where $f_n(x) = f(T^n x)$ and T is the shift on the input sequence space A_X^{∞} . If T also denotes the shift on the output space, then by construction $\bar{f}(Tx) = T\bar{f}(x)$ and hence for any output event F , $\bar{f}^{-1}(T^{-1}F) = T^{-1}\bar{f}^{-1}(F)$. Let m denote the process distribution for the encoded process. Since $m(F) = P(\bar{f}^{-1}(F))$ for any event $F \in \mathcal{B}(A_f)^{\infty}$, we have using the stationarity of the mapping f that

$$\begin{aligned} \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=0}^{n-1} m(T^{-i}F) &= \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=0}^{n-1} P(\bar{f}^{-1}(T^{-i}F)) \\ &= \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=0}^{n-1} P(T^{-i}\bar{f}^{-1}(F)) = \bar{P}(\bar{f}^{-1}(F)), \end{aligned}$$

where $\bar{P}$ is the stationary mean of P . Thus m is AMS. If G is an invariant output event, then $\bar{f}^{-1}(G)$ is also invariant since $T^{-1}\bar{f}^{-1}(G) =$

$\overline{f}^{-1}(T^{-1}G)$. Hence if input invariant sets can only have probability 1 or 0, the same is true for output invariant sets. $\square$

Finite-length Sliding-Block Codes

Stationary or sliding-block codes have a simple description when the sequence-to-symbol mapping characterizing the code depends on only a finite number of the sequence values; that is, the mapping is measurable with respect to a finite number of coordinates. As a particularly simple example, consider the code depicted in Figure 2.1, where an IID process $\{Z_n\}$ consisting of equiprobable coin flips is shifted into a length 3 shift register at the completion of the shift the table is used to produce one output value given the three binary numbers in the shift register. For

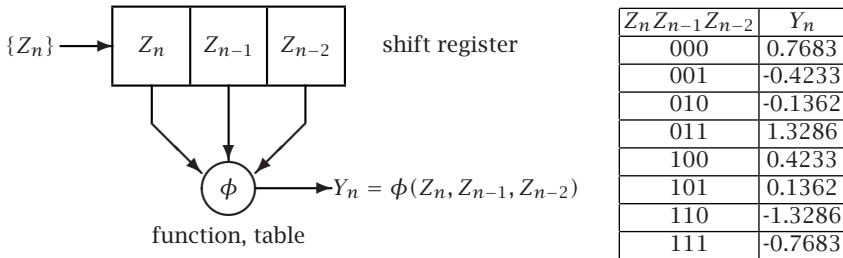


Fig. 2.1 A length 3 stationary code

the curious, this simple code tries to map coin flips into a sequence that looks approximately Gaussian. The output values correspond to eight possible values of an inverse cdf for a 0 mean Gaussian random variable with variance 3/4 evaluated at 8 equally spaced points in the unit interval. The values are “scrambled” to reduce correlation, but the marginal distribution is an approximation to Shannon optimal distribution when simulating or source coding an IID Gaussian sequence with mean 0 and variance 1. All of these ideas will be encountered later in the book.

More generally, suppose that we consider two-sided processes and that we have a measurable mapping

$$\phi : \prod_{i=-M}^D A_i \rightarrow B$$

and we define a sliding-block code by

$$f(x) = \phi(x_{-M}, \dots, x_0, \dots, x_D),$$

so that the output process is

$$Y_n = \phi(X_{n-M}, \dots, X_n, \dots, X_{n+D}),$$

a mapping of the contents of a shift register as depicted in [Figure 2.2](#). Note that the time order is reversed in the shift-register representa-

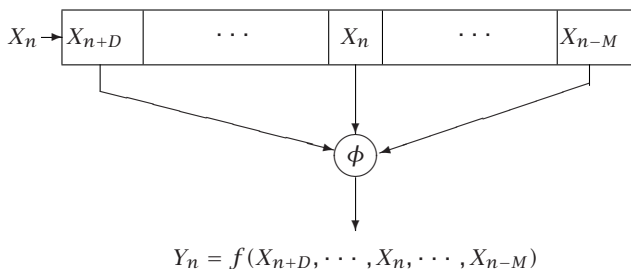


Fig. 2.2 Stationary or sliding-block code

tion since in the shift register new input symbols flow in from the left and exit from the right, but the standard way of writing a sequence is $\dots, X_{n-2}, X_{n-1}, X_n, X_{n+1}, X_{n+2}, \dots$ with “past” symbols on the left and “future” symbols on the right. The standard shift is the *left* shift so that shifting the above sequence results in the new sequence $\dots, X_{n-1}, X_n, X_{n+1}, X_{n+2}, X_{n+3}, \dots$. Rather than adding to the notational clutter by formally mapping sequences or vectors into a reversed-time form, we shall suffer the minor abuse of notation and follow tradition by using the first format (time increases to the left) for shift-registers, and the second notation (time increases to the right) when dealing with theory and stationary mappings. Context should make the usage clear and clarification will be added when necessary.

The *length* of the code is the length of the shift register or dimension of the vector argument, $L = D + M + 1$.

The mapping ϕ induces a sequence-to-symbol mapping f and a corresponding stationary sequence coder $\bar{f}$. The mapping ϕ is also called a sliding-block code or a finite-length sliding-block code or a finite-window sliding-block code. M is called the *memory* of the code and D is called the *delay* of the code since M past source symbols and D future symbols are required to produce the current output symbol. The *window length* or *constraint length* of the code is $M + D + 1$, the number of input symbols viewed to produce an output symbol. If $D = 0$ the code is said to be *causal*. If $M = 0$ the code is said to be *memoryless*.

There is a problem with the above model if we wish to code a one-sided source since if we start coding at time 0, there are no input symbols with negative indices. Hence we either must require the code be memoryless ($M = 0$) or we must redefine the code for the first M instances (e.g., by “stuffing” the code register with arbitrary symbols) or we must only define the output for times $i \geq M$. For two-sided sources a finite-length sliding-block code is stationary. In the one-sided case it is not even defined precisely unless it is memoryless, in which case it is stationary.

While codes that depend on infinite input sequences may not at first glance seem to be a reasonable physical model of a coding system, it is possible for such codes to depend on the infinite sequence only through a finite number of coordinates. In addition, some real codes may indeed depend on an unboundedly large number of past inputs because of feedback.

Sliding-Block Codes and Partitions

Codes mapping sequences (or vectors) into discrete alphabets have an alternative representation in terms of partitions and range spaces or *codebooks*. Given a sliding-block code $f : A^\infty \rightarrow B$ where B is discrete, suppose that we index the members of the set B as $B = \{b_i; i \in \mathbb{I}\}$ where $\mathbb{I}$ is a finite or infinite collection of positive integers. Since codes are assumed to be measurable mappings, the sets $P_i = \{x : x \in A^\infty : f(x) = b_i\} = f^{-1}(b_i)$, $i \in \mathbb{I}$, collectively form a measurable partition $\mathcal{P} = \{P_i, i \in \mathbb{I}\}$ of A^∞ ; that is, they are disjoint and collectively exhaustive. The sets P_i are referred to as the *atoms* of the partition. The range space $B = \{b_i; i \in \mathbb{I}\}$ is called the *codebook* of the code f or *output alphabet* and it will be assumed without loss of generality that its members are distinct. The code f can be expressed in terms of its partition and codebook by

$$f(x) = \sum_i b_i 1_{P_i}(x), \quad (2.6)$$

where $1_P(x)$ is the indicator function for a set P . Conversely, given a partition and a codebook, (2.6) describes the corresponding code.

B-Processes

One use of sliding-block codes is to provide an easy yet powerful generalization of the simplest class of random processes. IID random pro-

cesses provide the simplest nontrivial example of a random process, typically the first example of a random process encountered in introductory courses is that of sequence of coin flips or rolls of a die. IID processes have no memory and are generally the easiest example to analyze. Stationary or sliding-block coding of an IID process preserves many of the most useful properties of the IID process, including stationarity, ergodicity, and mixing. In addition to providing a common mathematical model of many real processes, processes formed this way turn out to be one of the most important classes of processes in ergodic theory in a way that is relevant to this book — the class of stationary codings of IID processes is exactly the class of random processes for which equal entropy rate is both necessary and sufficient for two processes to be isomorphic in the sense that one can be coded by a stationary code into the other in an invertible way. This result is Ornstein's isomorphism theorem, a result far beyond the scope of this book. But the importance of the class was first recognized in ergodic theory, and adds weight to its emphasis in this presentation of entropy and information theory.

A process is said to be a *B-process* if it can be represented as a finite-alphabet stationary coding of an independent identically distributed (IID) process, where the IID process need not have a finite alphabet. Such processes are also called or *Bernoulli processes* in ergodic theory, but in information theory that name usually implies IID processes (often binary) and not the more general case of any stationary coding of an IID process, so here the name B-process will be used exclusively. The definition also extends to continuous alphabet processes, for example a stationary Gaussian autoregressive processes is also a B-process since it can be represented as the result of passing an IID Gaussian process through a stable autoregressive filter, which is a stationary mapping [173]. The emphasis here, however, will be on ordinary finite-alphabet B-processes. There are many other characterizations of this class of random processes, but the class of stationary codings of IID processes is the simplest and most suitable for the purposes of this book.

Let μ denote the original distribution of the IID process and let η denote the induced output distribution. Then for any output events F and G

$$\eta(F \cap T_B^{-n}G) = \mu(\bar{f}^{-1}(F \cap T_B^{-n}G)) = \mu(\bar{f}^{-1}(F) \cap T_A^{-n}\bar{f}^{-1}(G)),$$

since $\bar{f}$ is stationary. But μ is stationary and mixing since it is IID (see Section 6.7 of [55] or Section 7.7 of [58]) and hence this probability converges to

$$\mu(\bar{f}^{-1}(F))\mu(\bar{f}^{-1}(G)) = \eta(F)\eta(G)$$

and hence η is also mixing. Thus a B-process is mixing of all orders and hence is ergodic with respect to T_B^n for all positive integers n .

B-processes can be thought of as the most random of random processes since they have at their heart an IID process such as coin flips or dice rolls.

2.7 Block Codes

Another case of sequence coding arises when we have a measurable mapping $\alpha : A^N \rightarrow B^K$ and we define a sequence coder $f(x) = y$ by

$$y_{nK}^K = (y_{nK}, y_{nK+1}, \dots, y_{(n+1)K-1}) = \alpha(x_{nN}^N),$$

that is, the input is parsed into nonoverlapping blocks of length N and each is successively coded into a block of length K outputs without regard to past or previous input or output blocks. Clearly N input time units must correspond to K output time units in physical time if the code is to make sense. A code of this form is called a *block code* and it is a special case of an (N, K) sliding block code so that such a code is (T_A^N, T_A^K) -stationary.

Block Independent Processes

As sliding-block coding of an IID process leads to a more general class of random processes, one can also apply a block code to an IID process to obtain a more general class of random processes including IID processes as a special case (with blocklength = 1). The resulting process will be *block independent* in the sense that successive K -blocks will be independent since they depend on independent input N blocks. Unlike B -processes, however, the new processes are not in general stationary or ergodic even if the input was. The process can be modified by inserting a random uniformly distributed start time to convert the K -stationary process into a stationary process, but in general ergodicity is lost and sample functions will still exhibit blocking artifacts.

Sliding-Block vs. Block Codes

We shall be interested in constructing sliding-block codes from block codes and vice versa. Each has its uses. The random process obtained in the next section by sliding-block coding a stationary and ergodic process

Proof. A sliding-block coding is stationary and hence coding a stationary and ergodic process will yield a stationary and ergodic process (Lemma 2.4), which proves the first part. Pick an $\epsilon > 0$ such that $\epsilon N < \delta$. Given the stationary and ergodic process $\{X_n\}$ (that is also assumed to be aperiodic in the sense that it does not place all of its probability on a finite set of sequences) we can find an event $G \in \mathcal{B}_A^{\mathbb{T}}$ having probability less than ϵ . Consider the event $F = G - \bigcup_{i=1}^{N-1} T^{-i}G$, that is, F is the collection of sequences x for which $x \in G$, but $T^i x \notin G$ for $i = 1, \dots, N-1$. We next develop several properties of this set.

First observe that obviously $\mu(F) \leq \mu(G)$ and hence $\mu(F) \leq \epsilon$. The sequence of sets $T^{-i}F$ are disjoint since if $y \in T^{-i}F$, then $T^i y \in F \subset G$ and $T^{i+l} y \notin G$ for $l = 1, \dots, N-1$, which means that $T^j y \notin G$ and hence $T^j y \notin F$ for $N-1 \geq j > i$. Lastly we need to show that although F may have small probability, it is not 0. To see this suppose the contrary, that is, suppose that $\mu(G - \bigcup_{i=1}^{N-1} T^{-i}G) = 0$. Then

$$\mu(G \cap (\bigcup_{i=1}^{N-1} T^{-i}G)) = \mu(G) - \mu(G \cap (\bigcup_{i=1}^{N-1} T^{-i}G)^c) = \mu(G)$$

and hence $\mu(\bigcup_{i=1}^{N-1} T^{-i}G|G) = 1$. In words, if G occurs, then it is certain to occur again within the next N shifts. This means that with probability 1 the relative frequency of G in a sequence x must be no less than $1/N$ since if it ever occurs (which it must with probability 1), it must thereafter occur at least once every N shifts. This is a contradiction, however, since this means from the ergodic theorem that $\mu(G) \geq 1/N$ when it was assumed that $\mu(G) \leq \epsilon < 1/N$. Thus it must hold that $\mu(F) > 0$.

We now use the rare event F to define a sliding-block code. The general idea is simple, but a more complicated detail will be required to handle a special case. Given a sequence x , define $n(x)$ to be the smallest i for which $T^i x \in F$; that is, we look into the future to find the next occurrence of F . Since F has nonzero probability, $n(x)$ will be finite with probability 1. Intuitively, $n(x)$ should usually be large since F has small probability. Once F is found, we code backwards from that point using blocks of a 0 prefix followed by $N-1$ 1's. The appropriate symbol is then the output of the sliding block code. More precisely, if $n(x) = kN + l$, then the sliding-block code prints a 0 if $l = 0$ and prints a 1 otherwise. This idea suffices until the event F actually occurs at the present time, that is, when $n(x) = 0$. At this point the sliding-block code has just completed printing an N -cell of $0111 \dots 1$. It should not automatically start a new N -cell, because at the next shift it will be looking for a new F in the future to code back from and the new cells may not align with the old cells. Thus the coder looks into the future for the next F ; that is, it again seeks $n(x)$, the smallest i for which $T^i x \in F$. This time $n(x)$ must be greater than or equal to N since x is now in F and $T^{-i}F$ are disjoint for $i = 1, \dots, N-1$. After finding $n(x) = kN + l$, the coder again codes back

to the origin of time. If $l = 0$, then the two codes are aligned and the coder prints a 0 and continues as before. If $l \neq 0$, then the two codes are not aligned, that is, the current time is in the middle of a new code word. By construction $l \leq N - 1$. In this case the coder prints l 2's (filler poop) and shifts the input sequence l times. At this point there is an $n(x) = kN$ for such that $T^{n(x)}x \in F$ and the coding can proceed as before. Note that k is at least one, that is, there is at least one complete cell before encountering the new F .

By construction, 2's can occur only following the event F and then no more than N 2's can be produced. Thus from the ergodic theorem the relative frequency of 2's (and hence the probability that Z_n is not in an N -block) is no greater than

$$\lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=0}^{n-1} 1_2(Z_0(T^i x)) \leq \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=0}^{n-1} 1_F(T^i x)N = N\mu(F) \leq N\frac{\delta}{N} = \delta, \quad (2.7)$$

that is,

$$\Pr(Z_n \text{ is in an } N - \text{cell}) \geq 1 - \delta.$$

Since Z_n is stationary by construction,

$$\Pr(Z_k^N = 011 \cdots 1) = \Pr(Z_0^N = 011 \cdots 1) \text{ for all } k.$$

Thus

$$\Pr(Z_0^N = 011 \cdots 1) = \frac{1}{N} \sum_{k=0}^{N-1} \Pr(Z_k^N = 011 \cdots 1).$$

The events $\{Z_k^N = 011 \cdots 1\}$, $k = 0, 1, \dots, N - 1$ are disjoint, however, since there can be at most one 0 in a single block of N symbols. Thus

$$\begin{aligned} N\Pr(Z^N = 011 \cdots 1) &= \sum_{k=0}^{N-1} \Pr(Z_k^N = 011 \cdots 1) \\ &= \Pr\left(\bigcup_{k=0}^{N-1} \{Z_k^N = 011 \cdots 1\}\right). \end{aligned} \quad (2.8)$$

Thus since the rightmost probability is between $1 - \delta$ and 1,

$$\frac{1}{N} \geq \Pr(Z_0^N = 011 \cdots 1) \geq \frac{1 - \delta}{N}$$

which completes the proof. $\square$

The following corollary shows that a finite-length sliding-block code can be used in the lemma.

Corollary 2.1. *Given the assumptions of the lemma, a finite-length sliding-block code exists with properties (a)-(c).*

Proof. The sets G and hence also F can be chosen in the proof of the lemma to be finite dimensional, that is, to be measurable with respect to $\sigma(X_{-K}, \dots, X_K)$ for some sufficiently large K . Choose these sets as before with $\delta/2$ replacing δ . Define $n(x)$ as in the proof of the lemma. Since $n(x)$ is finite with probability one, there must be an L such that if

$$B_L = \{x : n(x) > L\},$$

then

$$\mu(B_L) < \frac{\delta}{2}.$$

Modify the construction of the lemma so that if $n(x) > L$, then the sliding-block code prints a 2. Thus if there is no occurrence of the desired finite dimensional pattern in a huge bunch of future symbols, a 2 is produced. If $n(x) < L$, then f is chosen as in the proof of the lemma. The proof now proceeds as in the lemma until (2.7), which is replaced by

$$\lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=0}^{n-1} 1_2(Z_0(T^i x)) \leq \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=0}^{n-1} 1_{B_L}(T^i x) + \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=0}^{n-1} 1_F(T^i x) N \leq \delta.$$

The remainder of the proof is the same. □

Application of the lemma to an IID source and merging the symbols 1 and 2 in the punctuation process immediately yield the following result since coding an IID process yields a B-process.

Corollary 2.2. *Given an integer N and a $\delta > 0$ there exists an (N, δ) -punctuation sequence $\{Z_n\}$ with the following properties:*

- (a) $\{Z_n\}$ is B-process (and hence stationary, ergodic, and mixing).
- (b) $\{Z_n\}$ has a binary alphabet $\{0, 1\}$ and it can output only N -cells of the form $011 \dots 1$ (0 followed by $N - 1$ ones) or individual ones; that is, each zero is always followed by at least $N - 1$ ones.
- (c) For all integers k

$$\frac{1 - \delta}{N} \leq \Pr(Z_k^N = 011 \dots 1) \leq \frac{1}{N}$$

and hence for any n

$$\Pr(Z_n \text{ is in an } N - \text{cell}) \geq 1 - \delta.$$

Random punctuation sequences are closely related to the Rohlin-Kakutani theorem, a classic result of ergodic theory. The language and notation is somewhat different and we shall return to the topic at the end of this chapter.

2.9 Memoryless Channels

Suppose that $q_{x_0}(\cdot)$ is a probability measure on $\mathcal{B}_B$ for all $x_0 \in A$ and that for fixed F , $q_{x_0}(F)$ is a measurable function of x_0 . Let ν be a channel specified by its values on output rectangles by

$$\nu_x(\times_{i \in \mathbb{J}} F_i) = \prod_{i \in \mathbb{J}} q_{x_i}(F_i),$$

for any finite index set $\mathbb{J} \subset \mathbb{T}$. Then ν is said to be a *memoryless channel*. Intuitively,

$$\Pr(Y_i \in F_i; i \in \mathbb{J} | X) = \prod_{i \in \mathbb{J}} \Pr(Y_i \in F_i | X_i).$$

In fact two forms of memorylessness are evident in a memoryless channel. The channel is *input memoryless* in that the probability of an output event involving $\{Y_i; i \in \{k, k+1, \dots, m\}\}$ does not involve any inputs before time k , that is, the past inputs. The channel is also *input nonanticipatory* since this event does not depend on inputs after time m , that is, the future inputs. The channel is also *output memoryless* in the sense that for any given input x , output events involving nonoverlapping times are independent, i.e.,

$$\nu_x(Y_1 \in F_1 \bigcap Y_2 \in F_2) = \nu_x(Y_1 \in F_1) \nu_x(Y_2 \in F_2).$$

2.10 Finite-Memory Channels

A channel ν is said to have finite input memory of order M if for all one-sided events F and all n

$$\nu_x((Y_n, Y_{n+1}, \dots) \in F) = \nu_{x'}((Y_n, Y_{n+1}, \dots) \in F)$$

whenever $x_i = x'_i$ for $i \geq n - M$. In other words, for an event involving Y_i 's after some time n , knowing only the inputs for the same times and M time units earlier completely determines the output probability. Similarly ν is said to have finite anticipation of order L if for all one-sided events F and all n

$$\nu_x((\dots, Y_n) \in F) = \nu_{x'}((\dots, Y_n) \in F)$$

provided $x'_i = x_i$ for $i \leq n + L$. That is, at most L future inputs must be known to determine the probability of an event involving current and past outputs.

Channels with finite input memory were introduced by Feinstein [41].

A channel ν is said to have *finite output memory of order K* if for all one-sided events F and G and all inputs x , if $k > K$ then

$$\nu_x((\cdots, Y_n) \in F \bigcap (Y_{n+k}, \cdots) \in G) = \nu_x((\cdots, Y_n) \in F) \nu_x((Y_{n+k}, \cdots) \in G);$$

that is, output events involving output samples separated by more than K time units are independent.

Channels with finite output memory were introduced by Wolfowitz [195].

Channels with finite memory and anticipation are historically important as the first real generalizations of memoryless channels for which coding theorems could be proved. Furthermore, the assumption of finite anticipation is physically reasonable as a model for real-world communication channels. The finite memory assumptions, however, exclude many important examples, e.g., finite-state or Markov channels and channels with feedback filtering action. Hence we will emphasize more general notions which can be viewed as approximations or asymptotic versions of the finite memory assumption. The generalization of finite input memory channels requires some additional tools and is postponed to the next chapter. The notion of finite output memory can be generalized by using the notion of mixing.

2.11 Output Mixing Channels

A channel is said to be *output mixing* (or *asymptotically output independent* or *asymptotically output memoryless*) if for all output rectangles F and G and all input sequences x

$$\lim_{n \rightarrow \infty} |\nu_x(T^{-n}F \bigcap G) - \nu_x(T^{-n}F)\nu_x(G)| = 0.$$

More generally it is said to be *output weakly mixing* if

$$\lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=0}^{n-1} |\nu_x(T^{-i}F \bigcap G) - \nu_x(T^{-i}F)\nu_x(G)| = 0.$$

Unlike mixing systems, the above definitions for channels place conditions only on output rectangles and not on all output events. Output mixing channels were introduced by Adler [2].

The principal property of output mixing channels is provided by the following lemma.

Lemma 2.7. *If a channel is stationary and output weakly mixing, then it is also ergodic. That is, if ν is stationary and output weakly mixing and if μ is stationary and ergodic, then also $\mu\nu$ is stationary and ergodic.*

Proof: The process $\mu\nu$ is stationary by Lemma 2.1. To prove that it is ergodic it suffices from Lemma 2.3 to prove that for all input/output rectangles of the form $F = F_B \times F_A$, $F_B \in \mathcal{B}_A^{\mathbb{T}}$, $F_A \in \mathcal{B}_B^{\mathbb{T}}$, and $G = G_B \times G_A$ that

$$\lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=0}^{n-1} \mu\nu(T^{-i}F \cap G) = \mu\nu(F)\mu\nu(G).$$

We have that

$$\begin{aligned} & \frac{1}{n} \sum_{i=0}^{n-1} \mu\nu(T^{-i}F \cap G) - m(F)m(G) = \\ & \frac{1}{n} \sum_{i=0}^{n-1} \mu\nu((T_B^{-i}F_B \cap G_B) \times (T_A^{-i}F_A \cap G_A)) - \mu\nu(F_B \times F_A)\mu\nu(G_B \times G_A) = \\ & \frac{1}{n} \sum_{i=0}^{n-1} \int_{T_A^{-i}F_A \cap G_A} d\mu(x) \nu_x(T_B^{-i}F_B \cap G_B) - \mu\nu(F_B \times F_A)\mu(G_B \times G_A) = \\ & \frac{1}{n} \sum_{i=0}^{n-1} \left(\int_{T_A^{-i}F_A \cap G_A} d\mu(x) \nu_x(T_B^{-i}F_B \cap G_B) \right. \\ & \quad \left. - \int_{T_A^{-i}F_A \cap G_A} d\mu(x) \nu_x(T_B^{-i}F_B) \nu_x(G_B) \right) + \\ & \frac{1}{n} \sum_{i=0}^{n-1} \left(\int_{T_A^{-i}F_A \cap G_A} d\mu(x) \nu_x(T_B^{-i}F_B) \nu_x(G_B) - \mu\nu(F_B \times F_A)\mu\nu(G_B \times G_A) \right). \end{aligned}$$

The first term is bound above by

$$\begin{aligned} & \frac{1}{n} \sum_{i=0}^{n-1} \int_{T_A^{-i}F_A \cap G_A} d\mu(x) |\nu_x(T_B^{-i}F_B \cap G_B) - \nu_x(T_B^{-i}F_B) \nu_x(G_B)| \leq \\ & \int d\mu(x) \frac{1}{n} \sum_{i=0}^{n-1} |\nu_x(T_B^{-i}F_B \cap G_B) - \nu_x(T^{-i}F_B) \nu_x(G_B)| \end{aligned}$$

which goes to zero from the dominated convergence theorem since the integrand converges to zero from the output weakly mixing assumption. The second term can be expressed using the stationarity of the channel as

$$\int_{F_A} d\mu(x) \nu_x(G_B) \frac{1}{n} \sum_{i=0}^{n-1} 1_{F_A}(T_A^i x) \nu_{T_A^i x}(F_B) - \mu\nu(F)\mu\nu(G).$$

The ergodic theorem implies that as $n \rightarrow \infty$ the sample average goes to its expectation

$$\int d\mu(x) 1_{F_A}(x) \nu_x(F_B) = \mu\nu(F)$$

and hence the above formula converges to 0, proving the lemma. $\square$

The lemma provides an example of a completely random channel that is also ergodic in the following corollary.

Corollary 2.3. *Suppose that ν is a stationary completely random channel described by an output measure η . If η is weakly mixing, then ν is ergodic. That is, if μ is stationary and ergodic and η is stationary and weakly mixing, then $\mu\nu = \mu \times \eta$ is stationary and ergodic.*

Proof: If η is weakly mixing, then the channel ν defined by $\nu_x(F) = \eta(F)$, all $x \in A^{\mathbb{T}}$, $F \in \mathcal{B}_B^{\mathbb{T}}$ is output weakly mixing. Thus ergodicity follows from the lemma. $\square$

2.12 Block Independent Channels

The idea of a memoryless channel can be extended to a block memoryless or block independent channel. Given integers N and K (usually $K = N$) and a probability measure $q_{x^N}(\cdot)$ on $\mathcal{B}_B^K$ for each $x^N \in A^N$ such that $q_{x^N}(F)$ is a measurable function of x^N for each $F \in \mathcal{B}_B^K$. Let ν be specified by its values on output rectangles by

$$\nu_x(\gamma : \gamma_i \in G_i; i = m, \dots, m+n-1) = \prod_{i=0}^{\lfloor \frac{n}{K} \rfloor} q_{x_{iN}^N}(G_i),$$

where $G_i \in \mathcal{B}_B$, all i , where $\lfloor z \rfloor$ is the largest integer contained in z , and where

$$G_i = \bigtimes_{j=m+iK}^{m+(i+1)K-1} F_j$$

with $F_j = B$ if $j \geq m+n$. Such channels are called *block memoryless channels* or *block independent channels*. A deterministic block independent and block stationary channel is a sequence coder formed by a block code.

The primary use of block independent channels is in the construction of a channel given finite-dimensional conditional probabilities; that is, one has probabilities for output K -tuples given input N -tuples and one wishes to model a channel consistent with these finite-dimensional distributions. The finite-dimensional distributions themselves may be the result of an optimization problem or an estimate based on observed behavior. An immediate problem is that a channel constructed in this

manner may not be stationary, although it is clearly (N, K) -stationary. In Section 2.14 it is seen how to modify a block independent channel so as to produce a stationary channel. The basic idea is to occasionally insert some random spacing between the blocks so as to “stationarize” the channel.

Block independent channels are a special case of the class of conditionally block independent channels, which are considered next.

2.13 Conditionally Block Independent Channels

A *conditionally block independent (CBI)* channel resembles the block independent channel in that for a given input sequence the outputs are block independent. It is more general, however, in that the conditional probabilities of the output block may depend on the entire input sequence (or at least on parts of the input sequence not in the same time block).

A channel is CBI if its values on output rectangles satisfy

$$v_x(\gamma : \gamma_i \in F_i; i = m, \dots, m+n-1) = \prod_{i=0}^{\lfloor \frac{n}{K} \rfloor} v_x(\gamma : \gamma_{iN}^N \in G_i).$$

where as before

$$G_i = \bigtimes_{j=m+iK}^{m+(i+1)K-1} F_j$$

with $F_j = B$ if $j \geq m+n$. Block memoryless channels are clearly a special case of CBI channels.

These channels have only finite output memory, but unlike the block independent channels they need not have finite input memory or anticipation.

2.14 Stationarizing Block Independent Channels

Block memoryless channels (and CBI channels) are both block stationary channels. Connecting a stationary input to a block stationary channel will yield a block stationary input/output pair process, but it is sometimes desirable to have a stationary model. In this section we consider a technique of “stationarizing” a block independent channel in order to produce a stationary channel. Intuitively, a stationarized block independent (SBI) channel is a block independent channel with random spacing inserted between the blocks according to a random punctuation process.

That is, when the random blocking process produces N -cells (which is most of the time), the channel uses the N -dimensional conditional distribution. When it is not using an N cell, the channel produces some arbitrary symbol in its output alphabet. We now make this idea precise. Let N , K , and $q_{x^N}(\cdot)$ be as in block independent channel of Section 2.12. We now assume that $K = N$, that is, one output symbol is produced for every input symbol and hence output blocks have the same number of symbols as input blocks. This is done for simplicity as the more general case adds significant notational clutter for minimal conceptual gain.

Given $\delta > 0$ let γ denote the distribution of an (N, δ) -random punctuation sequence $\{Z_n\}$. Let $\mu \times \gamma$ denote the product distribution on $(A^\mathbb{T} \times \{0, 1\}^\mathbb{T}, \mathcal{B}_A^\mathbb{T} \times \mathcal{B}_{\{0,1\}}^\mathbb{T})$; that is, $\mu \times \gamma$ is the distribution of the pair process $\{X_n, Z_n\}$ consisting of the original source $\{X_n\}$ and the random punctuation source $\{Z_n\}$ with the two sources being independent of one another. Define a regular conditional probability (and hence a channel) $\pi_{x,z}(F)$, $F \in \{\mathcal{B}_B\}^\mathbb{T}$, $x \in A^\mathbb{T}$, $z \in \{0, 1\}^\mathbb{T}$ by its values on rectangles as follows: Given z , let $J_2(z)$ denote the collection of indices i for which $z_i = 2$ and hence for which z_i is not in an N -cell and let $J_0(z)$ denote those indices i for which $z_i = 0$, that is, those indices where N -cells begin. Let q^* denote a trivial probability mass function on B placing all of its probability on a reference letter b^* . Given an output rectangle

$$F = \{\gamma : \gamma_j \in F_j; j \in \mathbb{J}\} = \times_{j \in \mathbb{J}} F_j,$$

define

$$\pi_{x,z}(F) = \prod_{i \in \mathbb{J} \cap J_2(z)} q^*(F_i) \prod_{i \in \mathbb{J} \cap J_0(z)} q_{x_i^N} \left(\times_{j=i}^{i+N-1} F_j \right),$$

where we assume that $F_i = B$ if $i \notin \mathbb{J}$. Connecting the product source $\mu \times \gamma$ to the channel π yields a hookup process $\{X_n, Z_n, Y_n\}$ with distribution, say, r , which in turn induces a distribution p on the pair process $\{X_n, Y_n\}$ having distribution μ on $\{X_n\}$. If the alphabets are standard, p also induces a regular conditional probability for Y given X and hence a channel ν for which $p = \mu\nu$. A channel of this form is said to be an (N, δ) -stationarized block independent or SBI channel.

Lemma 2.8. *An SBI channel is stationary and ergodic. Thus if a stationary (and ergodic) source μ is connected to an SBI channel ν , then the output is stationary (and ergodic).*

Proof: The product source $\mu \times \gamma$ is stationary and the channel π is stationary, hence so is the hookup $(\mu \times \gamma)\pi$ or $\{X_n, Z_n, Y_n\}$. Thus the pair process $\{X_n, Y_n\}$ must also be stationary as claimed. The product source $\mu \times \gamma$ is ergodic from Corollary 2.3 since it can be considered as the input/output process of a completely random channel described by a mixing (hence also weakly mixing) output measure. The channel π is output

strongly mixing by construction and hence is ergodic from Lemma 2.4. Thus the hookup $(\mu \times \gamma)\pi$ must be ergodic. This implies that the coordinate process $\{X_n, Y_n\}$ must also be ergodic. This completes the proof. $\square$

The block independent and SBI channels are useful primarily for proving theorems relating finite-dimensional behavior to sequence behavior and for simulating channels with specified finite-dimensional behavior. The SBI channels will also play a key role in deriving sliding-block coding theorems from block coding theorems by replacing the block distributions by trivial distributions, i.e., by finite-dimensional deterministic mappings or block codes.

The SMB channel was introduced by Pursley and Davisson [29] for finite-alphabet channels and further developed by Gray and Saadat [70], who called it a randomly blocked conditionally independent (RBCI) channel. We opt for the first name because these channels resemble block memoryless channels more than CBI channels.

2.15 Primitive Channels

Primitive channels were introduced by Neuhoff and Shields [136, 133] as a physically motivated general channel model. The idea is that most physical channels combine the input process with a separate noise process that is independent of the signal and then filter the combination in a stationary fashion. The noise is assumed to be IID since the filtering can introduce dependence. The construction of such channels strongly resembles that of the SBI channels. Let γ be the distribution of an IID process $\{Z_n\}$ with alphabet W , let $\mu \times \gamma$ denote the product source formed by an independent joining of the original source distribution μ and the noise process Z_n , let π denote the deterministic channel induced by a stationary sequence coder $f: A^{\mathbb{T}} \times W^{\mathbb{T}} \rightarrow B^{\mathbb{T}}$ mapping an input sequence and a noise sequence into an output sequence. Let $r = (\mu \times \gamma)\pi$ denote the resulting hookup distribution and $\{X_n, Z_n, Y_n\}$ denote the resulting process. Let p denote the induced distribution for the pair process $\{X_n, Y_n\}$. If the alphabets are standard, then p and μ together induce a channel $v_x(F)$, $x \in A^{\mathbb{T}}$, $F \in \mathcal{B}_B^{\mathbb{T}}$. A channel of this form is called a *primitive channel*.

Lemma 2.9. *A primitive channel is stationary with respect to any stationary source and it is ergodic. Thus if μ is stationary and ergodic and v is primitive, then μv is stationary and ergodic.*

Proof: Since μ is stationary and ergodic and γ is IID and hence mixing, $\mu \times \gamma$ is stationary and ergodic from Corollary 2.3. Since the deterministic channel is stationary, it is also ergodic from Lemma 2.4 and the

resulting triple $\{X_n, Z_n, Y_n\}$ is stationary and ergodic. This implies that the component process $\{X_n, Y_n\}$ must also be stationary and ergodic, completing the proof. $\square$

2.16 Additive Noise Channels

Suppose that $\{X_n\}$ is a source with distribution μ and that $\{W_n\}$ is a “noise” process with distribution γ . Let $\{X_n, W_n\}$ denote the induced product source, that is, the source with distribution $\mu \times \gamma$ so that the two processes are independent. Suppose that the two processes take values in a common alphabet A and that A has an addition operation $+$, e.g., it is a semi-group. Define the sliding-block code f by $f(x, w) = x_0 + w_0$ and let $\bar{f}$ denote the corresponding sequence coder. Then as in the primitive channels we have an induced distribution r on triples $\{X_n, W_n, Y_n\}$ and hence a distribution on pairs $\{X_n, Y_n\}$ which with μ induces a channel ν if the alphabets are standard.

Example 2.1. A channel of this form is called a *additive noise channel* or a *signal-independent additive noise channel*.

If the noise process is a B-process, then this is easily seen to be a special case of a primitive channel and hence the channel is stationary with respect to any stationary source and ergodic. If the noise is only known to be stationary, the channel is still stationary with respect to any stationary source. Unless the noise is assumed to be at least weakly mixing, however, it is not known if the channel is ergodic in general.

2.17 Markov Channels

We now consider a special case where A and B are finite sets with the same number of symbols. For a fixed positive integer K , let $\mathbf{P}$ denote the space of all $K \times K$ stochastic matrices $P = \{P(i, j); i, j = 1, 2, \dots, K\}$. Using the Euclidean metric on this space we can construct the Borel field $\mathcal{P}$ of subsets of $\mathbf{P}$ generated by the open sets to form a measurable space $(\mathbf{P}, \mathcal{P})$. This, in turn, gives a one-sided or two-sided sequence space $(\mathbf{P}^{\mathbb{T}}, \mathcal{P}^{\mathbb{T}})$.

A map $\phi : A^{\mathbb{T}} \rightarrow \mathbf{P}^{\mathbb{T}}$ is said to be *stationary* if $\phi T_A = T_P \phi$. Given a sequence $P \in \mathbf{P}^{\mathbb{T}}$, let $\mathcal{M}(P)$ denote the set of all probability measures on $(B^{\mathbb{T}}, \mathcal{B}^{\mathbb{T}})$ with respect to which $Y_m, Y_{m+1}, Y_{m+2}, \dots$ forms a Markov chain with transition matrices $P_m, P_{m+1}, \dots$ for any integer m , that is, $\lambda \in \mathcal{M}(P)$ if and only if for any m

$$\lambda[Y_m = y_m, \dots, Y_n = y_n] = \lambda[Y_m = y_m] \prod_{i=m}^{n-1} P_i(y_i, y_{i+1}),$$

$$n > m, y_m, \dots, y_n \in B.$$

In the one-sided case only $m = 1$ need be verified. Observe that in general the Markov chain is nonhomogeneous.

A channel $[A, \nu, B]$ is said to be *Markov* if there exists a stationary measurable map $\phi : A^{\mathbb{T}} \rightarrow \mathbf{P}^{\mathcal{I}}$ such that $\nu_x \in \mathcal{M}(\phi(x))$, $x \in A^{\mathbb{T}}$.

Markov channels were introduced by Kieffer and Rahe [98] who proved that one-sided and two-sided Markov channels are AMS. Their proof is not included as it is lengthy and involves techniques not otherwise used in this book. The channels are introduced for completeness and to show that several important channels and codes in the literature can be considered as special cases. A variety of conditions for ergodicity for Markov channels are considered in [69]. Most are equivalent to one already considered more generally here: A Markov channel is ergodic if it is output mixing.

2.18 Finite-State Channels and Codes

The most important special cases of Markov channels are finite-state channels and codes. Given a Markov channel with stationary mapping ϕ , the channel is said to be a *finite-state channel (FSC)* if we have a collection of stochastic matrices $P_a \in \mathbf{P}$; $a \in A$ and that $\phi(x)_n = P_{x_n}$, that is, the matrix produced by ϕ at time n depends only on the input at that time, x_n . If the matrices P_a ; $a \in A$ contain only 0's and 1's, the channel is called a *finite-state code*. There are several equivalent models of finite-state channels and we pause to consider an alternative form that is more common in information theory. (See Gallager [47], Ch. 4, for a discussion of equivalent models of FSC's and numerous physical examples.) An FSC converts an input sequence x into an output sequence y and a state sequence s according to a conditional probability

$$\Pr(Y_k = y_k, S_k = s_k; k = m, \dots, n | X_i = x_i, S_i = s_i; i < m) =$$

$$\prod_{i=m}^n P(y_i, s_i | x_i, s_{i-1}),$$

that is, conditioned on X_i, S_{i-1} , the pair Y_i, S_i is independent of all prior inputs, outputs, and states. This specifies a FSC defined as a special case of a Markov channel where the output sequence above is here the joint state-output sequence $\{y_i, s_i\}$. Note that with this setup, saying the Markov channel is AMS implies that the triple process of source, states,

and outputs is AMS (and hence obviously so is the Gallager input-output process). We will adapt the Kieffer-Rahe viewpoint and call the outputs $\{Y_n\}$ of the Markov channel states even though they may correspond to state-output pairs for a specific physical model.

In the two-sided case, the Markov channel is significantly more general than the FSC because the choice of matrices $\phi(x)_i$ can depend on the past in a very complicated (but stationary) way. One might think that a Markov channel is not a significant generalization of an FSC in the one-sided case, however, because there stationarity of ϕ does not permit a dependence on past channel inputs, only on future inputs, which might seem physically unrealistic. Many practical communications systems do effectively depend on the future, however, by incorporating delay in the coding. The prime example of such look-ahead coders are trellis and tree codes used in an incremental fashion. Such codes investigate many possible output strings several steps into the future to determine the possible effect on the receiver and select the best path, often by a Viterbi algorithm. (See, e.g., Viterbi and Omura [189].) The encoder then outputs only the first symbol of the selected path. While clearly a finite-state machine, this code does not fit the usual model of a finite-state channel or code because of the dependence of the transition matrix on future inputs (unless, of course, one greatly expands the state space). It is, however, a Markov channel.

2.19 Cascade Channels

We will often wish to connect more than one channel in cascade in order to form a communication system, e.g., the original source is connected to a deterministic channel (encoder) which is connected to a communications channel which is in turn connected to another deterministic channel (decoder). We now make precise this idea. Suppose that we are given two channels $[A, \nu^{(1)}, C]$ and $[C, \nu^{(2)}, B]$. The cascade of $\nu^{(1)}$ and $\nu^{(2)}$ is defined as the channel $[A, \nu, B]$ given by

$$\nu_x(F) = \int_{C^{\mathbb{T}}} \nu_u^{(2)}(F) d\nu_x^{(1)}(u).$$

In other words, if the original source sequence is X , the output to the first channel and input to the second is U , and the output of the second channel is Y , then $\nu_x^{(1)}(F) = P_{U|X}(F|x)$, $\nu_u(G) = P_{Y|U}(G|u)$, and $\nu_x(G) = P_{Y|X}(G|x)$. Observe that by construction $X \rightarrow U \rightarrow Y$ is a Markov chain.

Lemma 2.10. *A cascade of two stationary channels is stationary.*

Proof: Let T denote the shift on all of the spaces. Then

$$\nu_x(T^{-1}F) = \int_{C^{\mathbb{T}}} \nu_u^{(2)}(T^{-1}F) d\nu_x^{(1)}(u) = \int_{C^{\mathbb{T}}} \nu_u^{(2)}(F) d\nu_x^{(1)}T^{-1}(u).$$

But $\nu_x^{(1)}(T^{-1}F) = \nu_{Tx}^{(1)}(F)$, that is, the measures $\nu_x^{(1)}T^{-1}$ and $\nu_{Tx}^{(1)}$ are identical and hence the above integral is

$$\int_{C^{\mathbb{T}}} \nu_u^{(2)}(F) d\nu_{Tx}^{(1)}(u) = \nu_{Tx}(F),$$

proving the lemma. $\square$

2.20 Communication Systems

A *communication system* consists of a source $[A, \mu]$, a sequence encoder $f : A^{\mathbb{T}} \rightarrow B^{\mathbb{T}}$ (a deterministic channel), a channel $[B, \nu, B']$, and a sequence decoder $g : B'^{\mathbb{T}} \rightarrow \hat{A}^{\mathbb{T}}$. The overall distribution r is specified by its values on rectangles as

$$r(F_1 \times F_2 \times F_3 \times F_4) = \int_{F_1 \cap f^{-1}(F_2)} d\mu(x) \nu_{f(x)}(F_3 \cap g^{-1}(F_4)).$$

Denoting the source by $\{X_n\}$, the encoded source or channel input process by $\{U_n\}$, the channel output process by $\{Y_n\}$, and the decoded process by $\{\hat{X}_n\}$, then r is the distribution of the process $\{X_n, U_n, Y_n, \hat{X}_n\}$. If we let X, U, Y , and $\hat{X}$ denote the corresponding sequences, then observe that $X \rightarrow U \rightarrow Y$ and $U \rightarrow Y \rightarrow \hat{X}$ are Markov chains. We abbreviate a communication system to $[\mu, f, \nu, g]$.

It is straightforward from Lemma 2.10 to show that if the source, channel, and coders are stationary, then so is the overall process.

A key topic in information theory, which is a mathematical theory of communication systems, is the characterization of the optimal performance one can obtain for communicating a given source over a given channel using codes within some available class of codes. Precise definitions of *optimal* will be based on the notion of the quality of a system as determined by a measure of *distortion* between input and output to be introduced in Chapter 5.

2.21 Couplings

So far in this chapter the focus has been on combining a source $[A, \mu]$ and a channel $[A, \nu, B]$ or a code which together produce a pair or input/output process $[A \times B, \pi]$, where $\pi = \mu\nu$. The pair process in turn

induces an output process, $[B, \eta]$. Given a pair process $[A \times B, \pi]$, the induced input process $[A, \mu]$ and output process $[B, \eta]$ can be thought of as the *marginal processes* and μ and η the *marginal distributions* of the pair process $[A \times B, \pi]$ and its distribution. From a different viewpoint, we could consider the two marginal processes $[A, \mu]$ and $[B, \eta]$ as being given and define a *coupling* or *joining* of these two processes as any pair process $[A \times B, \pi]$ having the given marginals. Here we can view π as a coupling of μ and η .

In general, given any two processes $[A, \mu]$ and $[B, \eta]$, let $\mathcal{P}(\mu, \eta)$ denote the class of all pair process distributions corresponding to couplings of the two given distributions. This class is not empty because, for example, we can always construct a coupling using product measures. This corresponds to the pair process with the given marginals where the two processes are mutually independent or, in other words, the example of the completely random channel given earlier.

When it is desired to place emphasis on the names of the random processes rather than the distributions, we will refer to a pair process distribution $\pi_{X,Y}$ with marginals π_X and π_Y . If we begin with two separate processes with distributions μ_X and μ_Y , say, then $\mathcal{P}(\mu_X, \mu_Y)$ will denote the collection of all pair processes with marginals $\pi_X = \mu_X$ and $\pi_Y = \mu_Y$. Occasionally $\pi_{X,Y} \in \mathcal{P}(\mu_X, \mu_Y)$ will be abbreviated to $\pi_{X,Y} \Rightarrow \mu_X, \mu_Y$.

If one is given two sources and forms a coupling, then in the case of processes with standard alphabets the coupling implies a channel since the joint process distribution and the input process distribution together imply a conditional distribution of output sequences given input sequences, and this conditional distribution is a regular conditional distribution and hence describes a channel.

Couplings can also be defined for pairs of random vectors rather than random processes in a similar manner.

2.22 Block to Sliding-Block: The Rohlin-Kakutani Theorem

The punctuation sequences of Section 2.14 provide a means for converting a block code into a sliding-block code. Suppose, for example, that $\{X_n\}$ is a source with alphabet A and γ_N is a block code, $\gamma_N : A^N \rightarrow B^N$. (The dimensions of the input and output vector are assumed equal to simplify the discussion.) Typically B is binary. As has been argued, block codes are not stationary. One way to stationarize a block code is to use a procedure similar to that used to stationarize a block independent channel: send long sequences of blocks with occasional random spacing to make the overall encoded process stationary. Thus, for example, one could use a sliding-block code to produce a punctuation sequence $\{Z_n\}$ as in Corollary 2.1 which produces isolated 0's followed by KN 1's

and occasionally produces 2's. The sliding-block code uses γ_N to encode a sequence of K source blocks $X_n^N, X_{n+N}^N, \dots, X_{n+(K-1)N}^N$ if and only if $Z_n = 0$. For those rare times l when $Z_l = 2$, the sliding-block code produces an arbitrary symbol $b^* \in B$. The resulting sliding-block code inherits many of the properties of the original block code, as will be demonstrated when proving theorems for sliding-block codes constructed in this manner. This construction suffices for source coding theorems, but an additional property will be needed when treating the channel coding theorems and other applications. The shortcoming of the results of Lemma 2.6 and Corollary 2.1 is that important source events can depend on the punctuation sequence. In other words, probabilities can be changed by conditioning on the occurrence of $Z_n = 0$ or the beginning of a block code word. In this section we modify the simple construction of Lemma 2.6 to obtain a new punctuation sequence that is approximately independent of certain prespecified events. The result is a variation of the Rohlin-Kakutani theorem of ergodic theory [157] [83]. The development here is patterned after that in Shields [164]. See also Shields and Neuhoff [167].

We begin by recasting the punctuation sequence result in different terms. Given a stationary and ergodic source $\{X_n\}$ with a process distribution μ and a punctuation sequence $\{Z_n\}$ as in Section 2.14, define the set $F = \{x : Z_N(x) = 0\}$, where $x \in A^\infty$ is a two-sided sequence $x = (\dots, x_{-1}, x_0, x_1, \dots)$. Let T denote the shift on this sequence space. Restating Corollary 2.1 yields the following.

Lemma 2.11. *Given $\delta > 0$ and an integer N , an L sufficiently large and a set F of sequences that is measurable with respect to $(X_{-L}, \dots, X_L)$ with the following properties:*

- (A) *The sets $T^i F$, $i = 0, 1, \dots, N-1$ are disjoint.*
 (B)

$$\frac{1 - \delta}{N} \leq \mu(F) \leq \frac{1}{N}.$$

(C)

$$1 - \delta \leq \mu\left(\bigcup_{i=0}^{N-1} T^i F\right).$$

So far all that has been done is to rephrase the punctuation result in more ergodic theory oriented terminology. One can think of the lemma as representing sequence space as a “base” F together with its disjoint shifts $T^i F$; $i = 1, 2, \dots, N-1$, which make up most of the space, together with whatever is left over, a set $G = A^\infty - \bigcup_{i=0}^{N-1} T^i F$, a set which has probability less than δ which will be called the *garbage set*. This picture is called a *tower* or *Rohlin-Kakutani tower*. The basic construction is pictured in Figure 2.3.

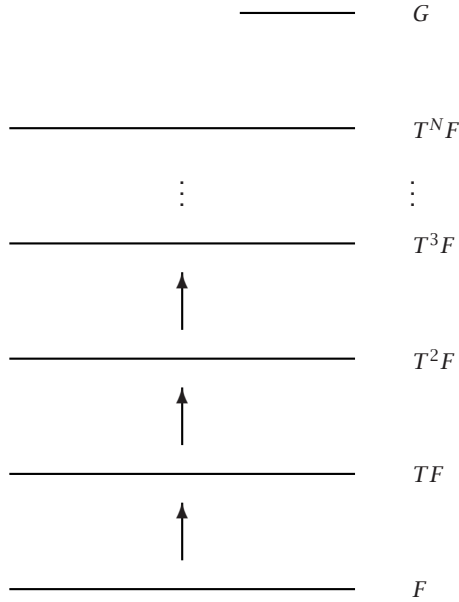


Fig. 2.3 Rohlin-Kakutani Tower

We can relate a tower to a punctuation sequence by identifying the base of the tower, the set F , as the set of sequences of the underlying process which yield $Z_0 = 0$, that is, the punctuation sequence at time 0 yields a 0, indicating the beginning of an N -cell.

Partitions

We now add another wrinkle — consider a finite partition $\mathcal{P} = \{P_i; i = 0, 1, \dots, \|\mathcal{P}\| - 1\}$ of A^∞ . One example is the partition of a finite-alphabet sequence space into its possible outputs at time 0, that is, $P_i = \{x : x_0 = a_i\}$ for $i = 0, 1, \dots, \|A\| - 1$. This is the zero-time partition for the underlying finite-alphabet process. Another possible partition would be according to the output of a sliding-block coding of x , the zero-time partition of the sliding-block coding (or the zero-time partition of the encoded process). In general there is a finite collection of important events that we wish to force to be approximately independent of the punctuation sequence and $\mathcal{P}$ is chosen so that the important events are unions of atoms of $\mathcal{P}$.

Given a partition $\mathcal{P}$, we define the *label* function

$$\text{label}_{\mathcal{P}}(x) = \sum_{i=0}^{\|\mathcal{P}\|-1} i 1_{P_i}(x),$$

where as usual 1_P is the indicator function of a set P . Thus the label of a sequence is simply the index of the atom of the partition into which it falls.

As $\mathcal{P}$ partitions the input space into which sequences belong to atoms of $\mathcal{P}$, $T^{-i}\mathcal{P}$ partitions the space according to which shifted sequences $T^i x$ belong to atoms of $\mathcal{P}$, that is, $x \in T^{-i}P_l \in T^{-i}\mathcal{P}$ is equivalent to $T^i x \in P_l$ and hence $\text{label}_{\mathcal{P}}(T^i x) = l$. The join

$$\mathcal{P}^N = \bigvee_{i=0}^{N-1} T^{-i}\mathcal{P}$$

partitions the space into sequences sharing N labels in the following sense: Each atom Q of $\mathcal{P}^N$ has the form

$$\begin{aligned} Q &= \{x : \text{label}_{\mathcal{P}}(x) = k_0, \text{label}_{\mathcal{P}}(Tx) = k_1, \dots, \text{label}_{\mathcal{P}}(T^{N-1}x) = k_{N-1}\} \\ &= \bigcap_{i=0}^{N-1} T^{-i}P_{k_i} \end{aligned}$$

for some N tuple of integers $\mathbf{k} = (k_0, \dots, k_{N-1})$. In the ergodic theory literature $\mathbf{k}$ is called the $\mathcal{P}$ - N -name of the atom Q . For this reason we index the atoms of $\mathcal{P}^N = \mathcal{Q}$ as $Q_{\mathbf{k}}$. Thus $\mathcal{P}^N$ breaks up the sequence space into groups of sequences which have the same labels for N shifts.

Gadgets

In ergodic theory a *gadget* is a quadruple $(T, F, N, \mathcal{P})$ where T is a transformation (for us a shift), F is an event such that $T^i F$; $i = 0, 1, \dots, N-1$ are disjoint (as in a Rohlin-Kakutani tower), and $\mathcal{P}$ is a partition of $\bigcup_{i=0}^{N-1} T^i F$. For concreteness, suppose that $\mathcal{P}$ is the zero-time partition of an underlying process, say a binary IID process. Consider the partition induced in F , the base of the gadget, by $\mathcal{P}^N = \{Q_{\mathbf{k}}\}$, that is, the collection of sets of the form $Q_{\mathbf{k}} \cap F$. By construction, this will be the collection of all infinite sequences for which the punctuation sequence at time zero is 0 ($Z_0 = 0$) and the $\mathcal{P}$ - n label of the next N outputs of the process is $\mathbf{k}$, in the binary example there are 2^N such binary N -tuples since $\|\mathcal{P}\| = 2$. The set $Q_{\mathbf{k}} \cap F$ together with its $N-1$ shifts (that is, the set $\bigcup_{i=0}^{N-1} T^i(Q_{\mathbf{k}} \cap F)$) is called a *column* of the gadget.

Gadgets provide an extremely useful structure for using a block code to construct a sliding-block code. Each atom $Q_{\mathbf{k}} \cap F$ in the base partition

contains all sequences corresponding the next N input values being a given binary N -tuple following a punctuation event $Z_0 = 0$.

Strengthened Rohlin-Kakutani Theorem

Lemma 2.12. *Given the assumptions of Lemma 2.11 and a finite partition $\mathcal{P}$, L and F can be chosen so that in addition to properties (A)-(C) it is also true that*

(D)

$$\mu(P_i|F) = \mu(P_i|T^l F); \quad l = 1, 2, \dots, N-1, \quad (2.9)$$

$$\mu(P_i|F) = \mu(P_i| \bigcup_{k=0}^{N-1} T^k F) \quad (2.10)$$

and

$$\mu(P_i \cap F) \leq \frac{1}{N} \mu(P_i). \quad (2.11)$$

Comment: Eq. (2.11) can be interpreted as stating that P_i and F are approximately independent since $1/N$ is approximately the probability of F . Only the upper bound is stated as it is all we need. Eq. (2.9) also implies that $\mu(P_i \cap F)$ is bounded below by $(\mu(P_i) - \delta)\mu(F)$.

Proof: Eq. (2.10) follows from (2.9) since

$$\begin{aligned} \mu(P_i| \bigcup_{l=0}^{N-1} T^l F) &= \frac{\mu(P_i \cap \bigcup_{l=0}^{N-1} T^l F)}{\mu(\bigcup_{l=0}^{N-1} T^l F)} = \frac{\sum_{l=0}^{N-1} \mu(P_i \cap T^l F)}{\sum_{l=0}^{N-1} \mu(T^l F)} \\ &= \frac{\sum_{l=0}^{N-1} \mu(P_i|T^l F)\mu(T^l F)}{N\mu(F)} = \frac{1}{N} \sum_{l=0}^{N-1} \mu(P_i|T^l F) \\ &= \mu(P_i|F) \end{aligned}$$

Eq. (2.11) follows from (2.10) since

$$\begin{aligned} \mu(P_i \cap F) &= \mu(P_i|F)\mu(F) = \mu(P_i| \bigcup_{k=0}^{N-1} T^k F)\mu(F) \\ &= \mu(P_i| \bigcup_{k=0}^{N-1} T^k F) \frac{1}{N} \mu(\bigcup_{k=0}^{N-1} T^k F) \\ &= \frac{1}{N} \mu(P_i \cap \bigcup_{k=0}^{N-1} T^k F) \leq \frac{1}{N} \mu(P_i) \end{aligned}$$

since the $T^k F$ are disjoint and have equal probability, The remainder of this section is devoted to proving (2.9).

We first construct using Lemma 2.11 a huge tower of size $KN \gg N$, the height of the tower to be produced for this lemma. Let S denote the base of this original tower and let ϵ be the probability of the garbage set. This height KN tower with base S will be used to construct a new tower of height N and a base F with the additional desired property. First consider the restriction of the partition $\mathcal{P}^N$ to F defined by $\mathcal{P}^N \cap F = \{Q_{\mathbf{k}} \cap F; \text{ all } KN\text{-tuples } \mathbf{k} \text{ with coordinates taking values in } \{0, 1, \dots, \|\mathcal{P}\| - 1\}\}$. $\mathcal{P}^N \cap F$ divides up the original base according to the labels of NK shifts of base sequences. For each atom $Q_{\mathbf{k}} \cap F$ in this base partition, the sets $\{T^l(Q_{\mathbf{k}} \cap F); k = 0, 1, \dots, KN - 1\}$ are disjoint and together form a *column* of the tower $\{T^l F; k = 0, 1, \dots, KN - 1\}$. A set of the form $T^l(Q_{\mathbf{k}} \cap F)$ is called the *lth level* of the column containing it. Observe that if $y \in T^l(Q_{\mathbf{k}} \cap F)$, then $y = T^l u$ for some $u \in Q_{\mathbf{k}} \cap F$ and $T^l u$ has label k_l . Thus we consider k_l to be the label of the column level $T^l(Q_{\mathbf{k}} \cap F)$. This complicated structure of columns and levels can be used to recover the original partition by

$$P_j = \bigcup_{l, \mathbf{k}: k_l = j} T^l(Q_{\mathbf{k}} \cap F) \cap (P_j \cap G), \quad (2.12)$$

that is, P_j is the union of all column levels with label j together with that part of P_j in the garbage. We will focus on the pieces of P_j in the column levels as the garbage has very small probability.

We wish to construct a new tower with base F so that the probability of P_i for any of N shifts of F is the same. To do this we form F dividing each column of the original tower into N equal parts. We collect a group of these parts to form F so that F will contain only one part at each level, the N shifts of F will be disjoint, and the union of the N shifts will almost contain all of the original tower. By using the equal probability parts the new base will have conditional probabilities for P_j given T^l equal for all l , as will be shown.

Consider the atom $Q = Q_{\mathbf{k}} \cap S$ in the partition $\mathcal{P}^N \cap S$ of the base of the original tower. If the source is aperiodic in the sense of placing zero probability on individual sequences, then the set Q can be divided into N disjoint sets of equal probability, say $W_0, W_1, \dots, W_{N-1}$. Define the set F_Q by

$$\begin{aligned} F_Q &= \left(\bigcup_{i=0}^{(K-2)N} T^{iN} W_0 \right) \bigcup \left(\bigcup_{i=0}^{(K-2)N} T^{1+iN} W_1 \right) \bigcup \dots \left(\bigcup_{i=0}^{(K-2)N} T^{N-1+iN} W_{N-1} \right) \\ &= \bigcup_{l=0}^{N-1} \bigcup_{i=0}^{(K-2)N} T^{l+iN} W_l. \end{aligned}$$

F_Q contains $(K-2)N$ shifts of W_0 , of $TW_1, \dots$ of $T^l W_l, \dots$ and of $T^{N-1} W_{N-1}$. Because it only takes N -shifts of each small set and because

it does not include the top N levels of the original column, shifting F_Q fewer than N times causes no overlap, that is, $T^l F_Q$ are disjoint for $j = 0, 1, \dots, N-1$. The union of these sets contains all of the original column of the tower except possibly portions of the top and bottom $N-1$ levels (which the construction may not include). The new base F is now defined to be the union of all of the $F_{Q_k} \cap S$. The sets $T^l F$ are then disjoint (since all the pieces are) and contain all of the levels of the original tower except possibly the top and bottom $N-1$ levels. Thus

$$\begin{aligned} \mu\left(\bigcup_{l=0}^{N-1} T^l F\right) &\geq \mu\left(\bigcup_{i=N}^{(K-1)N-1} T^i S\right) = \sum_{i=N}^{(K-1)N-1} \mu(S) \\ &\geq K - 2 \frac{1-\epsilon}{KN} = \frac{1-\epsilon}{N} - \frac{2}{KN}. \end{aligned}$$

By choosing $\epsilon = \delta/2$ and K large this can be made larger than $1 - \delta$. Thus the new tower satisfies conditions (A)-(C) and we need only verify the new condition (D), that is, (2.9). We have that

$$\mu(P_i | T^l F) = \frac{\mu(P_i \cap T^l F)}{\mu(F)}.$$

Since the denominator does not depend on l , we need only show the numerator does not depend on l . From (2.12) applied to the original tower we have that

$$\mu(P_i \cap T^l F) = \sum_{j, \mathbf{k}: k_j = i} \mu(T^j(Q_{\mathbf{k}} \cap S) \cap T^l F),$$

that is, the sum over all column levels (old tower) labeled i of the probability of the intersection of the column level and the l th shift of the new base F . The intersection of a column level in the j th level of the original tower with any shift of F must be an intersection of that column level with the j th shift of one of the sets $W_0, \dots, W_{N-1}$ (which particular set depends on l). Whichever set is chosen, however, the probability within the sum has the form

$$\begin{aligned} \mu(T^j(Q_{\mathbf{k}} \cap S) \cap T^l F) &= \mu(T^j(Q_{\mathbf{k}} \cap S) \cap T^j W_m) \\ &= \mu((Q_{\mathbf{k}} \cap S) \cap W_m) = \mu(W_m), \end{aligned}$$

where the final step follows since W_m was originally chosen as a subset of $Q_{\mathbf{k}} \cap S$. Since these subsets were all chosen to have equal probability, this last probability does not depend on m and hence on l and

$$\mu(T^j(Q_{\mathbf{k}} \cap S) \cap T^l F) = \frac{1}{N} \mu(Q_{\mathbf{k}} \cap S)$$

and hence

$$\mu(P_i \cap T^l F) = \sum_{j, \mathbf{k}: k_j = i} \frac{1}{N} \mu(Q_{\mathbf{k}} \cap S),$$

which proves (2.9) since there is no dependence on l . This completes the proof of the lemma. $\square$

Data Driven e-Science

Use Cases and Successful Applications of Distributed
Computing Infrastructures (ISGC 2010)

Lin, S.C.; Yen, E. (Eds.)

2011, XXII, 517 p., Hardcover

ISBN: 978-1-4419-8013-7