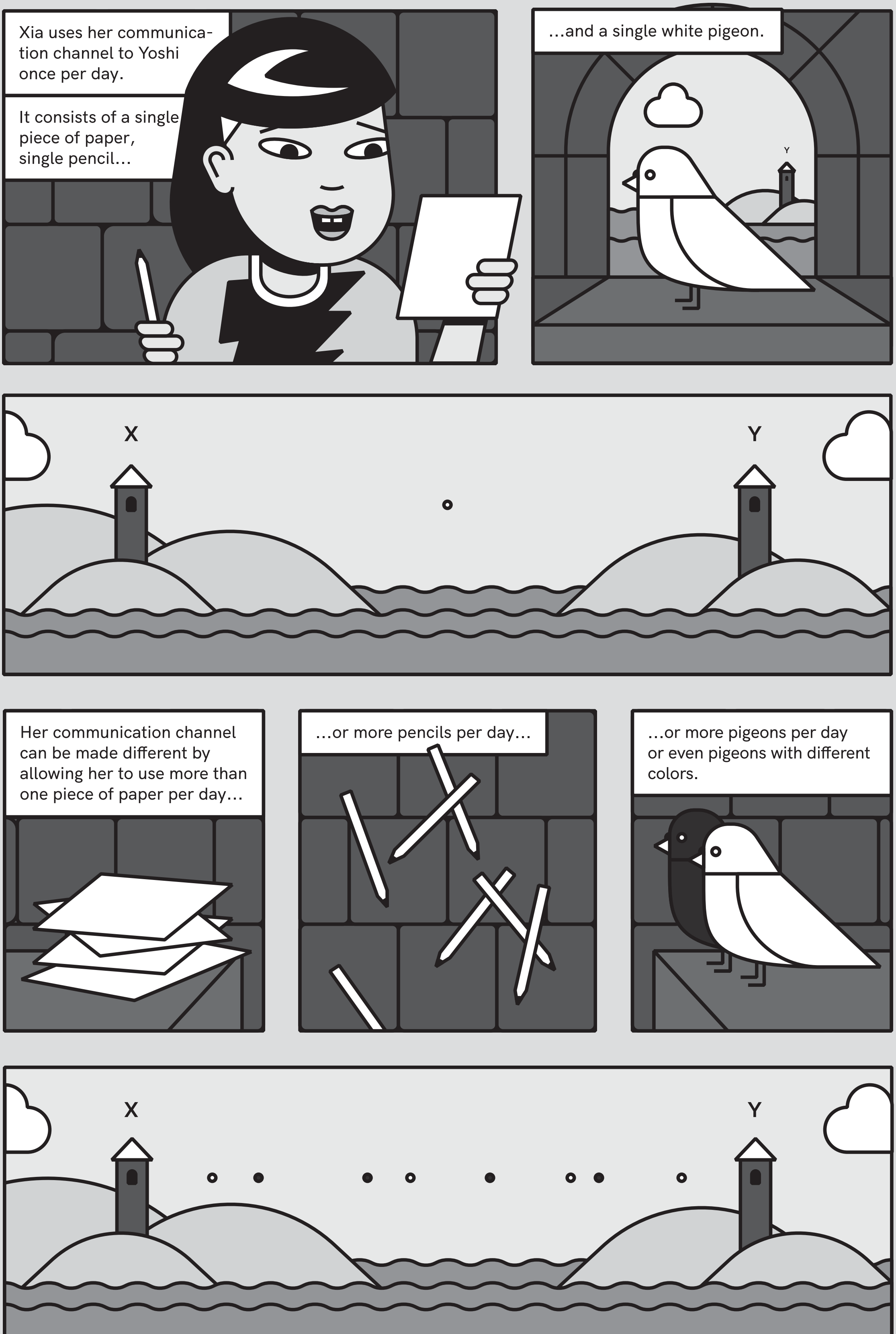


Story by Petar Popovski / Art by Peter Gregson

6

A Mathematical View on a Communication Channel

The communication subsystem involving the TXbaseband and the RXbaseband defines a communication channel in a sense of a *mathematical model for communication*, as formulated by Shannon in his groundbreaking paper from 1948. The schematic diagram of the general communication system considered by Shannon is shown in Figure 6.1. One of the most succinct descriptions of the models and methods used in communication and information theory is stated as follows[1]:

> *A channel is that part of the communication system that one is "unwilling or unable to change".*

This simple definition can also be extended to describe the layering principle, presented in Chapter 4, since a layer is a black box with well defined interfaces, which we are "unwilling or unable to change". In Shannon's model in Figure 6.1 it can be noticed that the system model features two layers, precisely as the one we have used to introduce layering in Figure 4.1. Specifically, at the higher layer the information source communicates with the destination; at the lower layer the transmitter communicates with the receiver. The noise source is part of the channel between the transmitter and the receiver and it represents all random disturbances that can affect the reception of the transmitted signal. On the other hand, once the transmitter and the receiver are fixed, then the transmitter, noise source, and the receiver can be grouped together and considered as a channel that is defined between the information source and the destination.

In the following discussion, we will see the importance of the above definition in order to identify what constitutes a communication channel at a given layer/part of the communication system.

6.1 A Toy Example: The Pigeon Communication Channel

In order to introduce the basic idea behind channel models as well as the terminology, we take an example of a channel that is rather unorthodox in communication engineering. This channel is illustrated in the cartoon at the beginning of this chapter. Xia and Yoshi are

1 This lucid statement has often been used by the great information theorist James L. Massey, but he himself attributed the definition to J. L. Kelly.

Wireless Connectivity: An Intuitive and Fundamental Guide, First Edition. Petar Popovski.

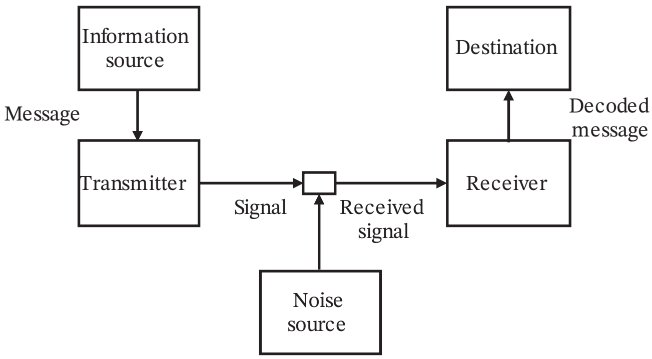


Figure 6.1 The general model of communication system considered by Shannon.

imprisoned in two towers that are several kilometers apart. Every morning at Xia's window there is a pigeon that can carry a single sheet of paper from Xia to Yoshi. After the pigeon delivers the paper to Yoshi, it returns to Xia, without getting any paper or message from Yoshi. Xia uses a pencil and handwriting in order to compose the messages to Yoshi. Xia does have a radio tuner in her tower, but Yoshi does not and the only way to get news updates is through the messages sent by Xia. The question is: *how much news per day can Yoshi learn?* Or, considering that the news is eventually encoded into bits and bytes, the question becomes: *what is the data rate, expressed in bits per day or bytes per day, at which Xia can supply news to Yoshi?*

A basic term used in this context is *channel use*: the smallest, atomic unit of communication that can be sent from the transmitter to the receiver. In the example, the channel use is a single piece of paper and we are interested in the maximal amount of information that can be sent by having the pigeon carry the paper from Xia to Yoshi. Note that the fact that the paper is sent once per day *is not part of the channel*. Even if the pigeon can fly between Xia and Yoshi several times during the day, the description of the channel use stays the same: flying more times per day may increase the data rate per day, but not the data rate per single channel use. We are thus separating the question "how frequently can you send data from Xia to Yoshi?" from the question "how many data bits can be sent in a single channel use?". Therefore, it is meaningful to speak about *bits per channel use*, in short bits/c.u.

We note that the communication channel between Xia and Yoshi is still not completely defined, as we have not put any constraint on how the messages are written and read. Here is an example how to define such *channel constraints and costs*. Assume that Xia can use up to one whole pencil per day: every day Xia returns the pencil remaining from the previous day and gets a new one. The pencil has a finite stroke width and the total amount of writing that can be done with a pencil, such as the total length of the lines, is limited. For this specific example of a channel, a higher cost does not necessarily result in a higher number of bits per channel use. To see this, we can think of the paper and the pencil being dimensioned in such a way that, if the whole pencil is used, then the paper becomes completely gray, not carrying any information.

On the way from Xia to Yoshi, the pigeon is subject to external disturbances that can, for example, damage the piece of paper or make the text less readable. These disturbances fall under the general notion of *noise*. Let us denote by x the original text that Xia sends. This is an *input* to the pigeon channel. What the pigeon delivers to Yoshi is the *output* of the channel. The noise impacts the paper in a random way, such that if two pieces of paper with identical writing are sent at two different times, then Yoshi may observe two different outputs or, in other words, read two different texts from the two received sheets. An essential property of a communication channel is that, for any possible input there is a set of possible output values, such that a particular value y occurs with a conditional probability $p(y|x)$. That is, for a given input text on the sheet sent by Xia, there are multiple options about what Yoshi can receive and the actual output observed by Yoshi is a random outcome.

6.1.1 Specification of a Communication Channel

The communication channel between Xia and Yoshi is specified by the following:

- The set $\mathcal{X}$ of possible input symbols x that Xia can transmit. One possible input symbol is represented by a text that is possible to write on a single piece of paper, such that the number of input symbols is equal to the number of possible texts. Since the number of readable letters that can be written on a piece of paper is finite, we can assume that the number of possible input symbols is finite.
- The set $\mathcal{Y}$ of possible output symbols y that can be received by Yoshi. It can happen that one, more or even all of the input symbols from $\mathcal{X}$ can appear unaltered as output symbols. This happens, for example, if the sheet is not affected in any way while being transported from Xia to Yoshi. However, there can be other possible output symbols. For example, part of the paper may be torn apart during the pigeon flight and consequently missing when the paper arrives at Yoshi. A damaged paper is an output symbol y that is different from any of the input symbols in $\mathcal{X}$. Another type of noise could be manifested in that the pigeon loses the paper along the way and nothing is delivered to Yoshi, which results in a channel output that is an empty symbol.
- The probabilities $p(y|x)$ for all possible input/output pairs. This is the most essential part of the communication channel, as it describes the external and uncontrollable factors that affect the transmission, sublimed under the term *noise*. Clearly, we would like to have a perfect channel where, for each input symbol $x \in \mathcal{X}$ we have $p(y = x|x) = 1$ and $p(y \neq x|x) = 0$, but this is rarely the case. In the mathematical theory of communication we always assume that $p(y|x)$ is given and we are unwilling or unable to change it.
- The cost and constraints imposed on how to use the channel. It should be noted that in this particular example the single-pencil-per-day constraint is already reflected in the possible input symbol $\mathcal{X}$.

A question that can be asked is: can we improve the capacity of the channel by using, for example, higher quality paper? This is similar to asking if we can improve the capacity of a wireless channel by changing the antenna. While the capacity to communicate in the system, understood in broader terms, is indeed increased, the question does not make sense if we consider the capacity per channel use that can be achieved once the communication channel is given as above. Namely, the communication channel is defined once the paper quality (or the antenna) is fixed and represents the part of the channel that we are unwilling or unable to change. The specific choice of paper or antenna will affect the specification of the probability $p(y|x)$.

Another question could be: can we improve the capacity by finding a faster pigeon or let the pigeon carry two/more pieces of paper? Again, the overall data rate of the system will be increased, but neither the pigeon speed nor the number of paper pieces is a part of the communication channel that we have defined. The speed of the pigeon corresponds to how often there is a transmission of a channel use. Recalling our baseband model, the capacity of a channel use describes the maximal number of bits that can be sent through a single baseband symbol. A different question is how often can we send that symbol and, as stated previously in this chapter, in our discussion these two questions are separated.

As illustrated in the cartoon, there are different ways to create new channels, for example, by allowing Xia to use more items per day, such as more than one piece of paper or more than one pencil. In a more extended example, Xia may be able to use multiple pigeons per day. Assume that Xia can freely decide to send a white or a black pigeon; then the choice of the pigeon color also carries information to Yoshi. For example, white pigeon stands for "0" and black pigeon stands for "1". Of all these channel extensions, in the remainder of this section we look how the constraint on the number of pencils per day affects the ability to communicate through the channel.

6.1.2 Comparison of the Information Carrying Capability of Mathematical Channels

The constraint of a single-pencil-per-day is rather simple and let us refer to that channel as 1pen-channel for brevity. Let us define two new channels by changing the constraint on the used pencils, while still assuming that the pigeon flies once per day:

- $\frac{1}{3}$pen-channel: one pencil is given every third day.
- 3pen-channel: three pencils are given every third day.

Which of these three channels is able to carry more information per channel use?

Let us at first compare the 1pen-channel and the $\frac{1}{3}$pen-channel. If one pencil is sufficient to have a sheet completely colored gray, then with the 1pen-channel we can create more different input symbols as compared to the $\frac{1}{3}$pen-channel. In order to see this, observe three consecutive days. If one pencil is available per day, then on any day Xia can create any symbol she wishes on the piece of paper. However, if one pencil is given per three days, then the symbols that can be created on the third day depend on how much of the pencil has been spent in the first two days. Consequently, anything that could be sent, in three consecutive days, over the $\frac{1}{3}$pen-channel, can be also sent over the 1pen-channel, but not vice versa. Thus, we expect that the 1pen-channel is more capable of carrying information than the $\frac{1}{3}$pen-channel.

Now let us compare the 1pen-channel with the 3pen-channel. Both channels use, on average, one pencil per day. Any input that can be created by the 1pen-channel can also be made by the 3pen-channel by deciding to use one pencil per day for the latter. However, there are other inputs that can be created in the 3pen-channel and not in the 1pen-channel. To see this, assume that a single pencil is not sufficient to make the paper fully gray, such that there can be advantage of using more than one pencil per day. In that case, the 3pen-channel can create inputs that are based on, for example, using more than one pencil the first day, but less than a single pencil in the second day. These inputs cannot be created by the 1pen-channel.

It can be concluded that the 3pen-channel is more capable of carrying information than the 1pen-channel.

In order to relate this discussion to a wireless example, consider two different baseband communication channels. For the first channel, each transmitted symbol has to have a power of at most P. The following is an example of a set of symbol powers that can be used in a sequence of symbols sent over that channel:

$$P \qquad 0.8P \qquad P \qquad 0.7P \qquad 0.5P \qquad P. \tag{6.1}$$

Let us now look at another, second channel, where the *average power* used over a group of three symbols can be at most P. An example of allowable symbol sequence through that channel is:

$$2P \qquad P \qquad 0 \qquad 1.8P \qquad 0.5P \qquad 0.7P. \tag{6.2}$$

The reader should note that the first channel is a *special case* of the second channel, as fixing the power of each symbol to P is *only one way* in which we can ensure that the average power used over three symbols is at most P. Thus, any combination of symbols that can be sent over the first channel, can also be sent over the second channel, but not vice versa. In other words, the sequence of input symbols (6.1) can appear in both channels, but the sequence (6.2) only in the second channel. We conclude that the amount of information that can be carried by the second channel can always be made at least as high as the one carried by the first channel. Observations of this type are essential in order to understand how the channel is defined in a particular situation and whether we are losing/gaining something from the ways in which the channel constraints are specified.

6.1.3 Assumptions and Notations

In this chapter we will always refer to point-to-point channels and links, where Xia transmits to Yoshi, unless explicitly stated otherwise. Recalling the terminology from Chapter 4, we will observe the communication channel in a connection oriented mode, after Xia and Yoshi have already agreed to communicate. Specifically, we assume that all the signaling has been exchanged and each channel use carries only pure data. The channel uses are numbered $1, 2, \ldots$. For example, one packet transmission can consist of L channel uses and we can also refer to the ith channel use, where $i = 1, 2, \ldots$.

6.2 Analog Channels with Gaussian Noise

We use the phrase *analog channels* for communication channels in which both the input and the output symbols are continuous, that is, complex or real numbers. A single channel use for the baseband channel, described in Chapter 5, is represented as:

$$y = hx + n. \tag{6.3}$$

The input symbol is x, the output symbol is y. The core parameters of this channels are h and n. We need to specify what Xia and Yoshi know about these parameters, as well as what constraints/costs are imposed on the usage of x. The channel coefficient h can be given as an exact value, learned before the communication channel starts to get used, or specified through a statistical characterization. In the latter case it can also be seen as a multiplicative noise, as it contributes to the random disturbances that define the channel.

6.2.1 Gaussian Channel

Let us now define a standard *AWGN (additive white Gaussian noise) channel*, often called simply a *Gaussian channel*. Xia and Yoshi have agreed that one packet transmission consumes L channel uses. During all the L channel uses, the value of h is constant $h = h_0$. For the ith channel use, n_i is an independent Gaussian random variable with a mean value zero and a variance of σ^2. The average power of the transmitted signal x is P, such that the following must be satisfied:

$$\frac{1}{L}\sum_{i=1}^{L}|x_i|^2 \leq P \tag{6.4}$$

where x_i is the symbol sent by Xia in the ith channel use. Both Xia and Yoshi know h_0, σ^2, P and L through some prior communication and/or measurement.

An important property of the Gaussian channel (6.3) is that it is *memoryless*: for a given ith channel use, the output y_i depends only on the input x_i and the noise n_i, but not on the previous inputs $x_{i-1}, x_{i-2}, \ldots$. Since the noise n_i is independent of the noise that occurs in the other channel uses, it follows that y_i does not depend on noise instances that affect the jth channel use, where $j \neq i$. Unless stated otherwise, it will be always assumed that we speak about memoryless channels.

The Gaussian channel is illustrated in Figure 6.2(a). The channel specification given above may seem exaggerated, as the researchers and engineers often refer to it as "Gaussian channel with an SNR equal to γ", which in our case would be $\gamma = \frac{|h_0|^2 P}{\sigma^2}$. The reason is that such a shortened specification makes a number of implicit assumptions, such as the knowledge of the channel coefficient, variance, etc. In order to see the impact of these assumptions, note that for the Gaussian channel Xia knows the exact mean value of the received signal y, as it is dependent on the symbol x she is transmitting. In other words, for a fixed x, Xia observes y as a Gaussian random variable with known mean value, equal to hx, and known variance σ^2. Based on that, Xia can use appropriate coding and modulation methods in order to achieve, for example, the highest goodput (recall the adaptive modulation from the previous chapter). However, if Xia does not know h, then Xia cannot treat y as a Gaussian random variable with known mean, but a random variable with a completely different statistics, which also depends on the statistics of h. Hence, when Xia does not know h, one cannot guarantee that she should use the same coding/modulation methods as for the case where the channel h is known. We will make a more detailed example of this later in this chapter.

6.2.2 Other Analog Channels Based on the Gaussian Channel

For completeness, we remark here that in the usual model of a Gaussian channel, Yoshi observes y perfectly, or more formally, taken from Yoshi's perspective, there is no uncertainty about y, while there is an uncertainty about x and, of course, n. We can also model

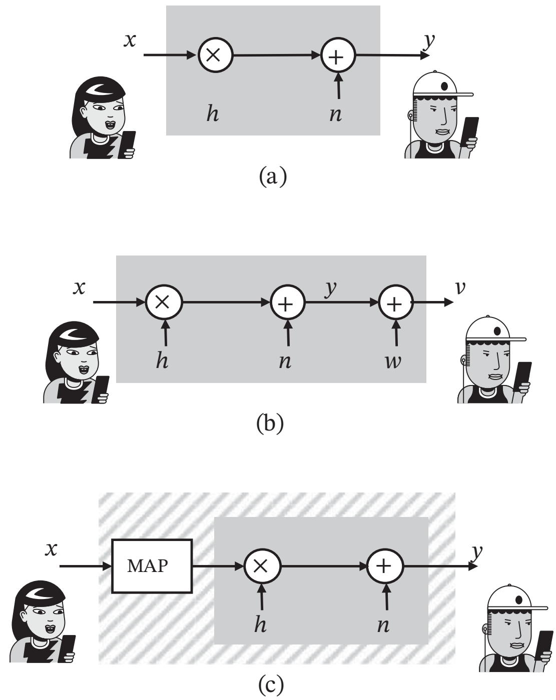


Figure 6.2 Three baseband communication channels between Xia and Yoshi with additive Gaussian noise. (a) Gaussian channel with known channel coefficient h at the transmitter and the receiver. (b) Channel with an imperfect observation at the receiver. The imperfect observation is modeled as additional Gaussian noise w. (c) Channel with additive Gaussian noise in which the inputs are constrained to belong to a discrete constellation, e. g. 8-PSK.

the situation in which Yoshi cannot perfectly observe y, but this would constitute a different channel. In order to illustrate this point, assume that Yoshi can perfectly observe v, which is a randomly altered version of y and specified through a probability density function $p(v|y)$. The reader can now think as if y is sent through a different communication channel, specified by $p(v|y)$; however, Xia does not have direct control over the input y of that channel. As an example, the channel y–v may be again a channel with Gaussian noise:

$$v = y + w \tag{6.5}$$

where w is a Gaussian noise, independent of the noise n and with variance σ_w^2 that is, in general, different from σ^2. As Figure 6.2(b) illustrates, now the channel between Xia and Yoshi is a *cascade* of two channels x–y and y–v. Ultimately, the channel between Xia and Yoshi is x–v, as we can hide y in the model and take into account only its statistical impact. For example, if the imperfect observation is modeled as an added noise, see (6.5), then the channel between Xia and Yoshi is represented as

$$v = hx + n + w. \tag{6.6}$$

If both w, n are Gaussian, then the channel $x - v$ is again a Gaussian channel, but with higher noise power $\sigma^2 + \sigma_w^2$. The equivalent channel is represented in Figure 6.2(b).

For the model of Gaussian channel that we have defined, we are allowed to design any possible modulation method that satisfies the power constraint (6.4) within the prescribed value of L symbols. A *specific* way to satisfy the constraint would be, for example, to use 8-PSK modulation, such that each of the eight constellation points has a power of P (recall the discussion with the pencils for the pigeon channel). By constraining that each transmitted symbol must belong to a 8-PSK modulation, we create a new channel that *overlays* the original Gaussian channel, as shown in Figure 6.2(c) and this channel is referred to as 8PSK_P. We can think of that channel as being constructed by adding a gadget that generates 8-PSK symbols over the original Gaussian channel and we are "unable or unwilling to change it". In a similar way, we can define a BSPK_P channel by using the Gaussian channel to send only two possible symbols, $-\sqrt{P}$ and $\sqrt{P}$.

It should be noted that whenever we fix the power of the modulation constellation to P and use the same constellation for each transmitted symbol, we are using the Gaussian channel in a sub-optimal way. Recall the discussion from Section 6.1.1 on the use of the constraints/cost in specifying a communication channel. Let us call the set of symbols sent in the L channel uses a *super-symbol*. For example, if the constellation contains S points, then there are S^L different super-symbols that can be transmitted within L channel uses. On the other hand, if Xia only respects the constraint (6.4), then she is allowed to send many more different super-symbols. To see why this is the case, let the power of that ith symbol that is part of the lth super-symbol be P_{li}. Then P_{li} can be any non-negative real number as long as:

$$\frac{1}{L}\sum_{i=1}^{L} P_{li} \leq P. \tag{6.7}$$

Since we have the freedom to select L real numbers to constitute a symbol, then the number of possible super-symbols that can be sent is uncountably infinite. Let us now assume that we want to be able to send S^L different super-symbols; then fixing the constellation of S points that needs to be the same for all L symbols is *only one possible way* to make that selection of S^L super-symbols. However, if we are allowed to select S^L from all super-symbols that satisfy (6.7), then the best set, selected in such a way, will be at least as good as the one that uses a fixed Sary constellation for all the symbols in the super-symbol. Here "goodness" is measured according to a communication criterion, such as probability of error in sending the entire super-symbol.

This argument shows the general principle that restricting the freedom at the channel input can only decrease the capability of a communication channel to carry information.

6.3 The Channel Definition Depends on Who Knows What

Changes in the assumptions about h can lead to significant deviations of the channel model from the model of a Gaussian channel. In the context of wireless communications, knowing what each variation brings is essential, as the channel coefficient is subject to random change due to *fading*, which summarizes the effects of the propagation environment, antennas, etc. More on the physics behind those changes is given in Chapter 10.

Here we consider a rather artificial, but sufficiently illustrative model of random *binary fading*, where the channel can only be in two states. Assume that, in each channel use, the coefficient h is selected randomly to be $h = 0.5$, with probability $p = 0.625$, and $h = -1.5$, with probability 0.375. The standard terminology used to describe this situation is to say that the channel changes its *state*, represented by h, randomly from one channel use to another. In this section we will be concerned with the knowledge of h that is available to

Table 6.1 A random sample of five consecutive channel uses.

Channel use number	**1**	**2**	**3**	**4**	**5**
Channel state, h	0.5	0.5	−1.5	0.5	−1.5

Xia and Yoshi before the actual channel use occurs. If Xia has that knowledge, we say that the channel state information (CSI) is available to the transmitter; if Yoshi knows h, we say that the CSI is available at the receiver.

In order to make the example more concrete, let us assume that that the transmission power is $P = 1$. If not explicitly stated otherwise, Xia uses BPSK modulation, where −1 stands for bit value 0 and 1 for bit value 1. The average received power at Yoshi is:

$$0.625 \cdot 0.5^2 \cdot 1 + 0.375 \cdot 1.5^2 \cdot 1 = 1 \tag{6.8}$$

that is, identical on average to the case in which the coefficient is constant $h = 1$ for all channel uses. Finally, we observe a random sample of five channel uses, whose channel states are depicted in Table 6.1.

We have the following four cases of CSI knowledge under which the system can operate:

- *Both Xia and Yoshi have CSI*. This falls back to the Gaussian channel as we have described it above. Figure 6.3(a) shows the probability density of the output signal when $h = 0.5$ and Xia sends 1, while Figure 6.3(b) shows the density when Xia sends −1. The decision rule at Yoshi's side is 1 when for positive signals, −1 for negative signals. According to the illustration, when the channel is in state $h = 0.5$, the noise power is sufficiently high to cause a significant probability of error, as Yoshi gets a negative signal when Xia sends 1. This probability is much lower when $h = -1.5$, as can be seen from Figure 6.3(c) and Figure 6.3(d). However, note that now, receiving a positive signal indicates that −1 has been sent. Since both Xia and Yoshi know the CSI in that channel use, Yoshi can apply the correct decision rule and invert the sign.

  If we remove the assumption that BPSK is used for transmission, then Xia can apply adaptive modulation in order to take advantage of the stronger channel. Specifically, Xia has two communication strategies, denoted by strategy 1 and 2, respectively. Strategy 1 uses a lower-order modulation, for example QPSK, and is applied when the channel is weaker with $h = 0.5$. Strategy 2 relies on a higher-order modulation, such as 8-PSK, and is used when when the channel has $h = -1.5$.

  Referring to Table 6.1, let us at first optimistically assume that Xia and Yoshi know the sequence of five channel states in advance, before the first channel use. Then Xia uses strategy 1 in the channel uses 1, 2, 4 and strategy 2 in uses 3, 5. Since Yoshi knows which strategy is used when, this is equivalent to the following case: the channel in state 0.5 for three consecutive uses and Xia and Yoshi communicate with strategy 1; then the channel is in state −1.5 for two consecutive uses and strategy 2 is used. Thus, the overall channel consists of two time-multiplexed Gaussian channels.

  Let us now look at the case in which Xia and Yoshi learn the channel state just before the channel use in which it is applicable. We still need one strong assumption, namely that Xia and Yoshi know before the first channel use that the state $h = 0.5$ will occur three

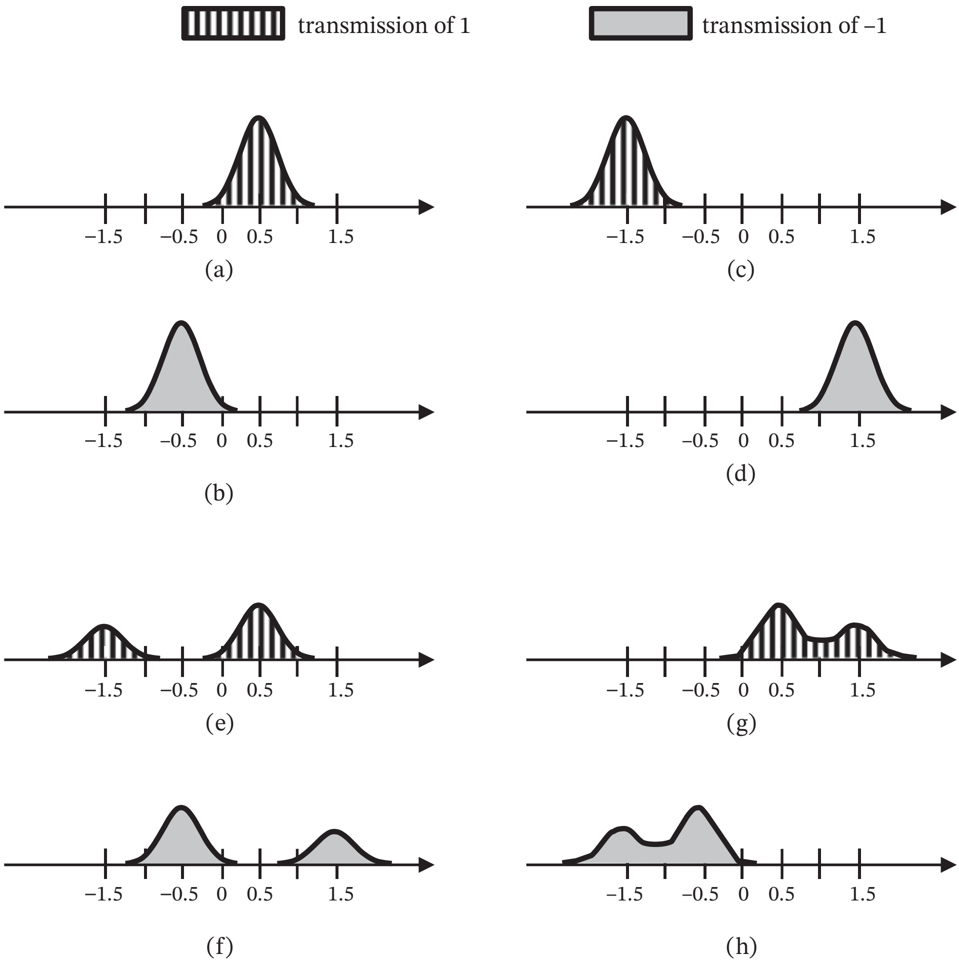


Figure 6.3 Distribution of the received/output signal at Yoshi's side. (a) Known $h = 0.5$ and 1 is sent. (b) Known $h = 0.5$ and -1 is sent. (c) Known $h = -1.5$ and 1 is sent. (d) Known $h = -1.5$ and -1 is sent. (e) Unknown h and 1 is sent. (f) Unknown h and -1 is sent. In (g) and (h), Xia knows h, Yoshi does not, and Xia represents the bit value 1 with 1 when $h = 0.5$ and -1 when $h = -1.5$. (g)/(h) shows the output distribution as seen by Yoshi when bit value 1/0 is sent, respectively.

times and $h = -1.5$ two times; however, Xia and Yoshi do not know in advance which state will be applicable in a particular channel use. This is a strong assumption, but is used here only for illustrative purposes. If, before the first channel use, both Xia and Yoshi know how many times will each channel state occur, but not when it will occur, then learning the channel state just before the channel use is as good as knowing the complete channel state sequence before the first channel use. The reason is that Xia again prepares for three channel uses with $h = 0.5$ and two with -1.5, and, more specifically, prepares three symbols created according to strategy 1 and two symbols according to strategy 2. When Xia learns that the next channel state will be, for example, $h = 0.5$, she picks the next symbol to be transmitted with strategy 1 and sends it. Since Yoshi has also learned the channel state, he knows that what is arriving is the next transmission symbol created according to strategy 1.

In summary, the common knowledge that Xia and Yoshi have about the states of the channel with binary fading allows them to use time-multiplexing between two communication strategies.

- *Both Xia and Yoshi do not have the CSI*. Figure 6.3(e) shows the distribution of the output signal when Xia sends 1. The channel h is a random factor, unknown to both Xia and Yoshi, such that it has essentially the same effect as a random noise, except that it appears as a multiplicative rather than an additive factor in (6.3). Figure 6.3(f) shows the distribution of the output signal when Xia sends -1. The distributions in Figures 6.3(e) and (f) are valid for all five channel uses from Table 6.1. Note that the height of the bell curve at 1.5 is lower than at 0.5, thus reflecting the fact that $h = 0.5$ occurs with a higher probability. With the distributions in Figures 6.3(e) and (f), we cannot use the same decision rule, as when 1 is sent it is likely to get positive as well negative value at Yoshi's side.

  One can think of what a good decision rule would be, but what is more important is to rethink the transmission strategy. Namely, having an average symbol power of 1 and having two input symbol, there is a better strategy than selecting $\{-1, 1\}$ as transmitted symbols. One possibility could be to use symbols with different power, e.g. power 0 when 0 is sent and power 2 when 1 is sent, such that the average power stays equal to 1. This is similar to the strategies for non-coherent communication discussed in the previous chapter; the reader is encouraged to use the examples from Figures 6.3(e) and (f) and try to devise a suitable decision criterion.
- *Xia has the CSI, but Yoshi does not*. If Xia sends 1, then, according to Xia, the distribution of the output signal is either given by Figure 6.3(a) or by Figure 6.3(c), since the distribution is conditioned on the fact that Xia knows h. But this is not true for Yoshi and thus the distribution of the output signal for Yoshi is given in Figure 6.3(e).

  However, this is not the same situation as when neither Xia nor Yoshi has the CSI and, in fact, Yoshi can still use the same simple decision rule. Since Xia knows h, then she can control her transmission and can thereby control the distribution of the output signal. A possible communication strategy could be as follows. If Xia would like to send the bit value 1, then she looks at the channel state and if $h = 0.5$, she transmits 1, otherwise she transmits -1. Using such a transmission strategy, whenever Xia wants to send the bit value 1, the distribution of the output signal for Yoshi is given by Figure 6.3(g). The output distribution that corresponds to bit value 0 is shown in Figure 6.3(h). Hence, now Yoshi can decide that Xia wants to send a bit value 1 whenever he observes a positive signal. Note that here Yoshi does not decide on the actual symbol sent by Xia, but the bit value behind it, and Yoshi is ignorant of whether the actual analog value sent by Xia is -1 or 1. In our example from Table 6.1, the bit value 0 is represented by -1 in the channel uses 1, 2, 4 and by 1 in the channel uses 3, 5.
- *Xia does not have the CSI, but Yoshi does*. In this case, if Xia transmits 1, then the distribution of the output signal to Xia looks as in Figure 6.3(e). On the other hand, Yoshi knows what the channel state is, such that to him, the output distribution is either according to Figure 6.3(a) or according to Figure 6.3(c). Yoshi can again apply the simple decision rule for positive/negative signals, since he knows h and can act exactly the same as in the case when both Xia and Yoshi know the CSI. What is different, though, is that Xia cannot

have two different strategies, 1 and 2, each to be used with a different channel state. In other words, by not having the CSI, Xia cannot adapt to the channel in the sense of using adaptive modulation and she has to use a fixed transmission scheme.

6.4 Using Analog to Create Digital Communication Channels

In introducing modulation and baseband symbols in the previous chapter, we did not think of the data as being directly represented by analog baseband symbols. The data is originally represented through bits and a group of bits is mapped onto a specific constellation point, thereby resulting in a modulated baseband symbol. This naturally puts forward the *binary channel* as one of the most fundamental communication channels, with a single bit as an input symbol and a single bit as an output symbol.

We can create a discrete binary channel by using a baseband module that sends a BPSK channel over a channel with AWGN. This is illustrated in Figure 6.4(a). The module "BPSK map" maps 0/1 to $-\sqrt{P}/\sqrt{P}$, respectively. At the output of the channel polluted with Gaussian noise, there is another block that makes a decision whether the transmitted signal has been positive or negative. We assume that the receiver Yoshi knows h, applies a matched filter, and obtains the decision variable (recall the discussion on BPSK from the previous chapter):

$$|h|^2 x + h^* n. \tag{6.9}$$

The next block outputs 1 if the decision variable is positive and 0 otherwise. As discussed in the previous chapter, by fixing the decision threshold, we also determine the error probabilities $P(1|0)$ and $P(0|1)$, which completely specifies the digital binary channel, as we can find the remaining probabilities as follows:

$$P(0|0) = 1 - P(1|0) \qquad P(1|1) = 1 - P(0|1).$$

Since the noise variance in the underlying Gaussian channel does not depend on the symbol that has been transmitted, the probability of error is identical for the two possible transmitted symbols $P(1|0) = P(0|1) = p$. This is one of the most frequently used channel models; it is known as the *binary symmetric channel (BSC)* and is commonly represented as shown in Figure 6.4(b). Note that the impact of the noise from the underlying Gaussian channel is indirect, as it determines the probability of error. In general, any communication channel with a discrete set of inputs and a discrete set of outputs can be described by specifying the probability $P(y|x)$ for all possible pairs (x, y), as depicted in Figure 6.4(a). We remark that the binary symmetric channel is memoryless.

6.4.1 Creating Digital Channels through Gray Mapping

Let us now consider the digital channel that is obtained by putting the message bits through a baseband module that uses QPSK symbols and sends them over a channel with AWGN. Let us fix the power of the QPSK constellation and select the constellation points from the set $\mathcal{S}_{\text{QPSK}} = \{s_1, s_2, s_3, s_4\}$, as depicted in Figure 6.4(c). At each channel use, Xia provides two bits to the TXbaseband module, while the RXbaseband module outputs two bits to Yoshi.

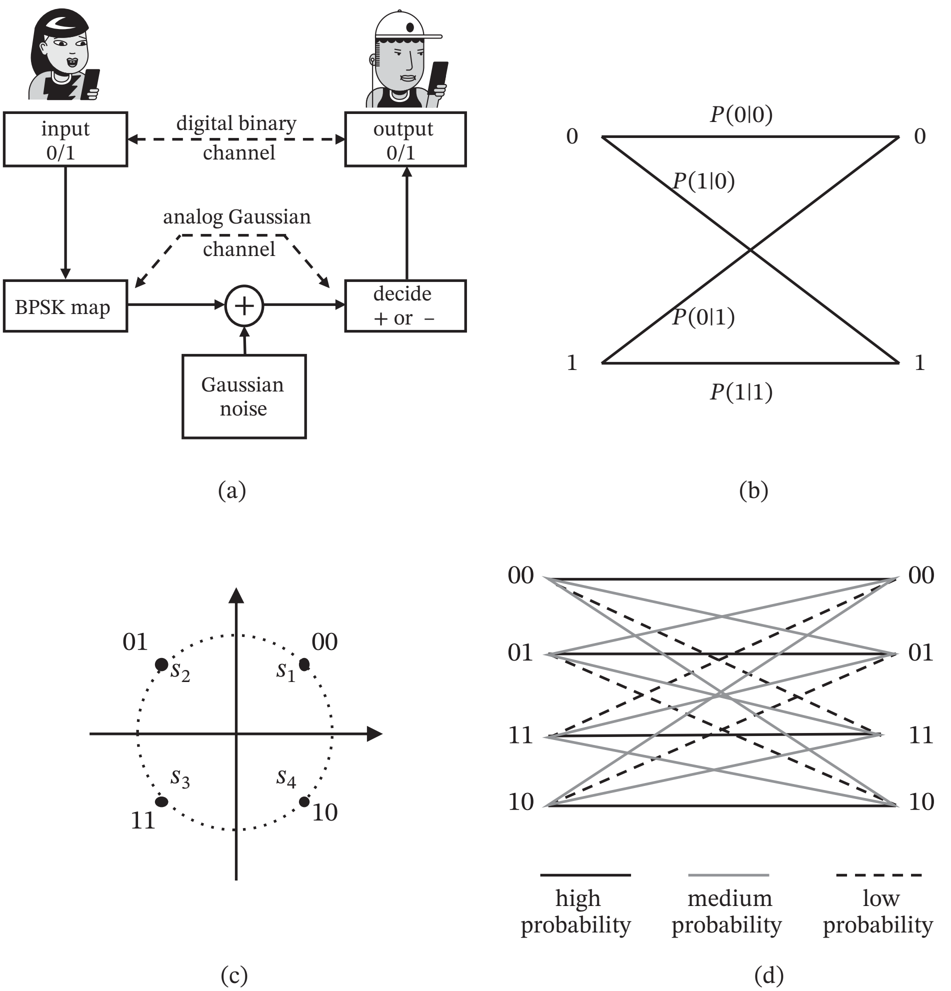


Figure 6.4 Creating digital channels from Xia to Yoshi by using BPSK or QPSK modulation. (a) System blocks are used to create a digital binary symmetric channel (BSC) that uses an underlying channel with Gaussian noise. (b) Common representation of a BSC. (c) QPSK modulation with Gray mapping of the bit pairs. (d) Equivalent digital channel obtained with the QPSK mapping from (c) and a symbol-by-symbol decision.

According to Figure 5.1, the user of the baseband channel is the DataSender that has to map the message/packet bits into baseband symbols x.

In a single channel use, two bits can be mapped to one input symbol. One common way to map the bits is shown in Figure 6.4(c) and is referred to as Gray mapping, introduced in the previous chapter. By assuming that the QPSK symbols are sent over a Gaussian channel and fixing the Gray mapping, we define a new digital channel that has four inputs and four outputs. We can represent the input and the output as two-bit vectors:

$$\mathbf{x}_i, \mathbf{y}_i \in \mathcal{B} = \{00, 01, 10, 11\} \tag{6.10}$$

and the digital channel is specified through the conditional probabilities:

$$P(\mathbf{y}_i|\mathbf{x}_i) \tag{6.11}$$

for all $\mathbf{x}_i, \mathbf{y}_i \in \mathcal{B}$. Note that these probabilities will change if the bits-to-symbol mapping is not Gray mapping, thus leading to a definition of a *different* discrete channel with four inputs and four outputs.

The probabilities in (6.11) can be computed using the distribution of the Gaussian noise and referring to Figure 6.4(c). Let us assume that $\mathbf{x}_i = (00)$ has been sent, which corresponds to the baseband transmission of the symbol $x_i = s_1$, then:

$$P(\mathbf{y}_i = 00|\mathbf{x}_i = 00) = P(00|00) = P(s_1|s_1)$$
$$P(01|00) = P(s_2|s_1) \qquad P(10|00) = P(s_4|s_1) \qquad P(11|00) = P(s_3|s_1). \tag{6.12}$$

Following the properties of the Gaussian noise, such as the symmetry and the fact that it is more likely that the noise will keep the received signal closer to the original point rather than to a more distant constellation point, it follows that the probabilities from (6.12) can be ordered as:

$$P(00|00) > P(01|00) = P(10|00) > P(11|00). \tag{6.13}$$

This is indicated in Figure 6.4(d), which shows the equivalent digital communication channel with four inputs and four outputs. To create this channel we can refer again to Figure 6.4(a), with the following changes: (1) "input 0/1" is replaced with "input 00/01/10/11"; (2) "BPSK map" is replaced with "QPSK map"; (3) "decide + or −" is replaced with "decide s_1, s_2, s_3 or s_4", and (4) "output 0/1" is replaced with "output 00/01/10/11". Note that when $\mathbf{x}_i = 00$ is sent, then if $\mathbf{y}_i = 01$ there is only one bit error, but receiving $\mathbf{y}_i = 11$ leads to two bit errors. From this the main idea behind Gray mapping becomes evident: it minimizes the average number of bit errors that occur in a single symbol transmission. It is desirable to have a lower number of bit errors as those are more likely to be repairable by using error correction codes; this will be discussed in the next chapter.

Another property of QPSK is that each of the two bits mapped to the same symbol experiences errors that are independent of the value of the other bit. If we denote the input/output of the ith bit by x_i/y_i, where $i = 1, 2$, then we can write:

$$P(y_1y_2|x_1x_2) = P(y_1|x_1) \cdot P(y_2|x_2). \tag{6.14}$$

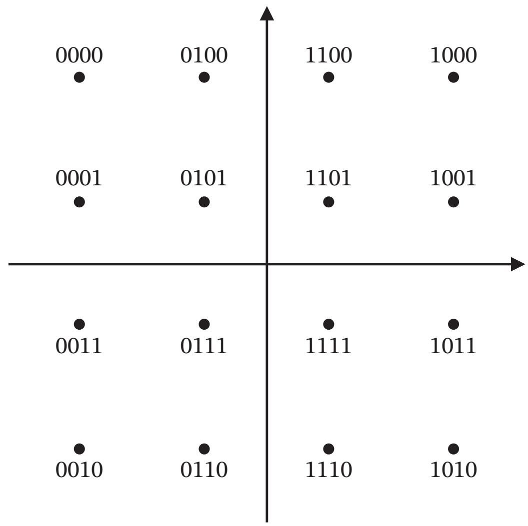
Therefore, a single use of a digital channel that is defined over a QPSK baseband transmission is equivalent to a simultaneous use of two independent binary symmetric channels that have identical bit error probabilities.

The latter property is not transferred to higher order complex constellations that use Gray mapping. For example, a 16-QAM with Gray mapping is depicted in Figure 6.5. It should be noted that there are other maps that have the properties of Gray mapping and this is only one of them. Each use of the digital channel defined over the 16-QAM baseband module has 4-bit input/output, denoted as follows:

$$\mathbf{z} = x_1x_2x_3x_4 \qquad \mathbf{y} = y_1y_2y_3y_4 \tag{6.15}$$

where each x_i and y_j corresponds to a bit value 0 or 1. Regardless of what kind of bit-to-symbol mapping is used, the probability of error does depend on the actual

Figure 6.5 16-QAM constellation with Gray mapping.



transmitted symbol. For example, when the transmitted symbol corresponds to one of the corner constellation points, labeled $0000, 1000, 0010, 1010$ in Figure 6.5, the probability of symbol error is lower compared to any other from the remaining twelve symbols. This is because the corner points have the least number of neighboring constellation points and thus a lower chance of the noise bringing the received signal into a decision region that belongs to another symbol. Another important difference with QPSK is that the four bits that constitute a single channel use are not equally reliable and do not have the same error probabilities. For example, for the mapping used in Figure 6.5, it can be checked that:

$$P(y_2 = 1|x_2 = 0) > P(y_3 = 1|x_3 = 0). \tag{6.16}$$

Finally, the transmission of four bits through the Gray-mapped constellation in Figure 6.5 *does not* correspond to a simultaneous transmission of four bits through four independent binary symmetric channels. This means that, in general:

$$P(y_1y_2y_3y_4|x_1x_2x_3x_4) \neq P(y_1|x_1)P(y_2|x_2)P(y_3|x_3)P(y_4|x_4). \tag{6.17}$$

The implication of this is that, for example, the probability of error when 0001 is sent is not the same with the probability of error when 0010 is sent, which can be checked by the reader. On the other hand, 0010 is a permutation of 0001 and if those four bits are sent through four identical binary symmetric channels, then, for example, the following should hold: $P(0110|0010) = P(0101|0001)$ and, overall, the error probability when 0001 is sent should be identical to the one when 0010 is sent.

6.4.2 Creating Digital Channels through Superposition

A digital channel can also be obtained when the high-order constellations, such as 16-QAM are created by superposing symbols of lower-order constellations. A simple example of a constellation created by superposition is 4-PAM (pulse amplitude modulation).

Figure 6.6(a) shows a 4-PAM constellation with power P. Once this constellation is fixed, then, strictly speaking, the channel is no longer Gaussian, but a 4-PAM input constrained analog channel with Gaussian noise. The type of digital channel that one can get by using this 4-PAM analog channel depends on the type of bit-to-symbol mapping. Figure 6.6(b) shows a possible Gray mapping for this constellation.

However, the same 4-PAM constellation can be obtained by superposing two binary 2-PAM signals, as shown in Figure 6.6(c). Each of the constituent 2-PAM signals is modulated independently with a single bit. It can be seen that the resulting bit-to-symbol mapping is not a Gray mapping, which is a consequence of the independent modulation of the constituent binary streams. As discussed in the previous chapter, no mapping of the constituent binary signals can result in a Gray mapping.

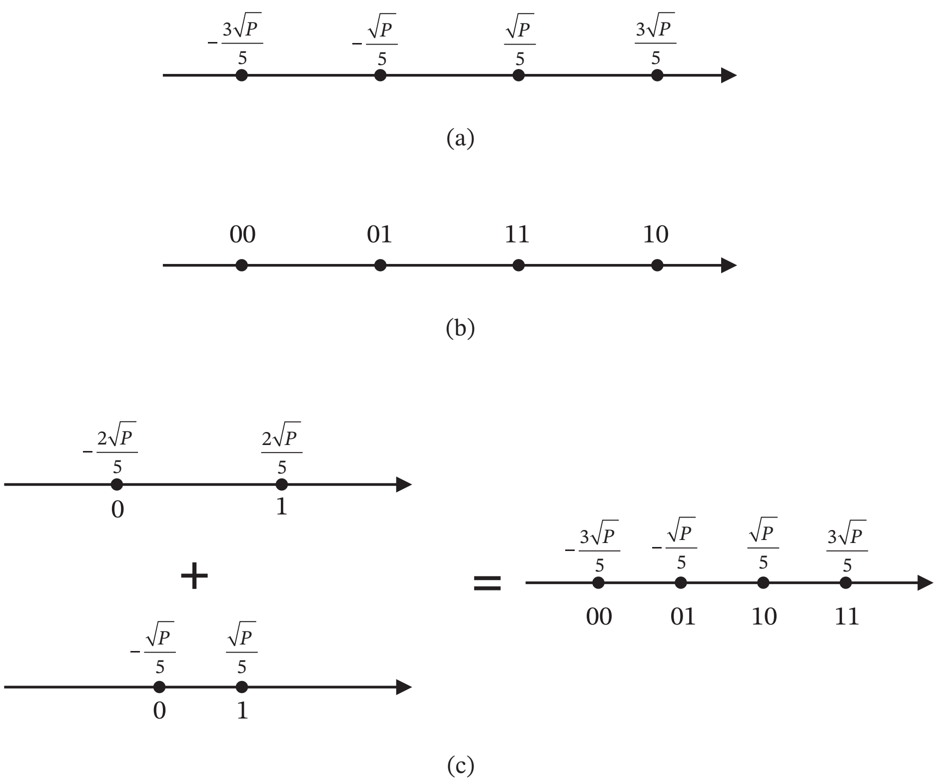


Figure 6.6 Difference between Gray mapping and the bit-to-symbol mapping obtained when a higher-order constellation is obtained by superposition of lower-order modulations. (a) 4-PAM modulation with power P. (b) A possible Gray mapping for the 4-PAM modulation. (c) Synthesis of a 4-PAM modulation by superposition of two 2-PAM modulated signals; the result is not a Gray mapping.

6.5 Transmission of Packets over Communication Channels

6.5.1 Layering Perspective of the Communication Channels

In the previous section we saw that it is possible to create a new channel by taking an existing communication channel and putting certain gadgets at the input and/or the output of that channel. Figure 6.4(a) can be interpreted as follows: the BSC operates by relying on the service provided by the BPSK/Gaussian channel. This irresistibly reminds us of the layering paradigm, described in Chapter 4, and we can thus extend the paradigm and built more channels in a hierarchical manner. For the example in Figure 6.7, channel 2 uses the black-box service provided by channel 1 and provides a black-box service to channel 3. In terms of layering, as discussed in Chapter 4, we can treat channel 1 as representing the lowest layer and channel 3 a representing the highest layer.

An important property that can be used in creating a new channel through the layered approach can be described as follows. For Figure 6.7, a single channel use of channel 2 can consist of one or more channel uses of channel 1. As an example, assume that the channels have the following inputs/outputs:

- Channel 1 is a BSC with inputs/outputs $\{0, 1\}$.
- Channel 2 has three inputs/outputs $\{A, B, C\}$.
- Channel 3 has eight inputs/outputs $Z_1, Z_2, \dots Z_8$.

Since the number of inputs for channel 2 is larger than two, the input block of channel two should map A, B, C to more than one binary symbol. A possible mapping can be the following:

$$A \mapsto 0, B \mapsto 10, C \mapsto 11.$$

In a similar way, each input symbol of channel 3 should be mapped to more than one symbol of channel two, for example:

$$Z_1 \mapsto AA, Z_2 \mapsto AB, Z_3 \mapsto AC, Z_4 \mapsto BA$$

$$Z_5 \mapsto BB, Z_6 \mapsto BC, Z_7 \mapsto CA, Z_8 \mapsto CB.$$

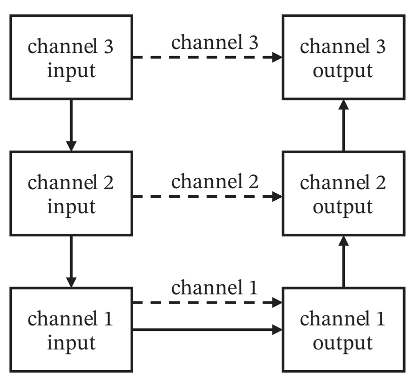


Figure 6.7 Layered representation of communication channels.

If each use of channel 1 is independently affected by noise, then we can compute the transition probabilities for channel 2 and channel 3 using the probabilities specified for channel 1.

It should be noted that the sequence $Z_3Z_5Z_8$ sent over channel 3 will result in a binary sequence sent over channel 1 that looks as follows: 01110101110, taking 11 bits. We have on purpose introduced three inputs in channel 2 in order to introduce inefficiency in the way the symbols from one channel are mapped to another. Note that, had the channel 3 been directly connected to channel 1, we could have represent each symbol Z_i with three bits and thus represent $Z_3Z_5Z_8$ with 9 bits. This is an example of *cross-layer optimization*, already introduced in the previous chapters.

In general, let us consider a communication channel that resides at a lower layer (LL), termed LLChannel, and its duty is to transfer each of its input symbols to the output, independently of all the other symbols before or after that symbol. Let LLChannel have S input/output symbols. We can create a channel, termed HLChannel, that resides at a layer that is higher than the LLChannel and each channel use of the HLChannel is mapped to L channel uses of the LLChannel. HLChannel can be seen as a black box that has S^L possible inputs/outputs and a single symbol for the HLChannel is in fact a *packet* that consists of L Sary symbols of the LLChannel. If the time between two channel uses for the LLChannel is T, then a single channel use of the HLChannel has a duration of $L \cdot T$. However, in order to devise a communication strategy over the HLChannel, we do not need to consider the timing (recall that timing is not part of the definition of a channel use) and we can thus think of HLChannel as a regular communication channel, with a remark that it has a large number of inputs and outputs.

6.5.2 How to Obtain Throughput that is not Zero

Bundling multiple channel uses of the LLChannel to represent a single use of the HLChannel is the essential operation towards enabling reliable communication, which will be elaborated in details in the next chapter. In order to see why this operation is essential, let us look at a LLChannel defined as a BSC that has a probability of error $p > 0$. Let Xia try to send data to Yoshi by making a single use of the LLChannel: she picks the next data bit, which can have value 0 or 1 with probability 0.5, and sends it. The receiver will get either 0 or 1, but has no way of knowing whether the bit is correct or not. Therefore, the throughput that can be achieved by sending one-by-one bit through this channel is *strictly zero*.

Let us now enhance the communication channel and allow the following type of operation: after receiving a single bit from Xia, then Yoshi immediately feeds back to Xia the bit value that he has received. The feedback from Yoshi is ideal and without errors. For the moment we are not concerned how this feedback is implemented, but we just introduce it to see how it can be used to get a non-zero throughput. The main observation is that Xia can still not achieve a positive throughput by making a single use of the LLChannel. This is because the feedback can only help Xia to understand that error has occurred, but Yoshi still lacks the knowledge on which bits are correct.

The problem with the BSC is that there is nothing certain about the data transmission that takes place in a single channel use. In order to see how this can be circumvented, let us assume that the LLChannel is a *binary erasure channel*, which completely changes the

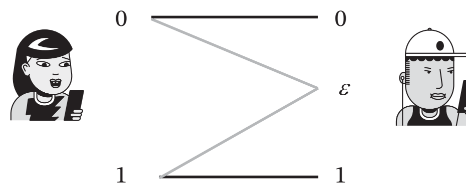


Figure 6.8 The binary erasure channel where Yoshi knows for sure if an error has occurred.

situation. The erasure channel is depicted in Figure 6.8 and the probability that a bit will get correctly to the destination is $(1 - p)$, while it is erased with probability p. If Yoshi receives 0 or 1, he knows that this is a correct value; however, Xia cannot know whether the bit she sent was erased or not. One may argue that in this case the throughput of a single channel use is $(1 - p)$, however, we somehow want to be sure that the information arrives at Yoshi and Xia knows that it has happened. If we now enhance the channel model and allow ideal feedback from Yoshi to Xia, then we can certainly say that the throughput is $(1 - p)$ bits per channel use: if an error occurs, both Xia and Yoshi know it and Xia can retransmit the same bit value in the next channel use.

The main conclusion from the previous discussion is that one can get a positive throughput by having a reliable detection of erasure on the receiver side. This brings us back to the importance of creating a new communication channel by grouping multiple channel uses of another, LLChannel, and ensuring that there is detectable erasure in the newly created channel. Let us assume that Xia again uses BSC to communicate to Yoshi, but now takes $b > 1$ bits to send them in a packet. Xia appends c bits for error detection to those b bits and thus obtains a packet of $(b + c)$ bits. These c check bits are uniquely determined by the other b bits. We have thus created a new channel that has 2^b possible inputs, denoted by $\omega_1, \omega_2, \ldots \omega_{2^b}$, which correspond to the possible patterns of the b data bits. The packet is sent by making $(b + c)$ channel uses of the BSC, such that there are 2^{b+c} possible outputs. Among them, 2^b outputs are equal to the inputs $\omega_1, \omega_2, \ldots \omega_{2^b}$. The remaining $(2^{b+c} - 2^b)$ outputs can only appear if at least one bit error has occurred in the packet.

Ideally, we would like the error check to operate as follows. If Xia sends ω_i, then Yoshi either receives the same output and declares a correctly received message or he receives another packet from the set of $(2^{b+c} - 2^b)$ and declares an erasure (packet error). However, this ideal operation assumes that it is impossible that an input ω_i produces an output equal to another valid input ω_j, where $\omega_j \neq \omega_i$. This is denoted as an undetected error. Nevertheless there is always a non-zero probability for undetected error as long as the LLChannel is a channel in which an input sequence can produce any output sequence with non-zero probability. For example, the BSC has this property: the combination of bit errors can produce any output bit sequence, regardless of which input bit sequence has been sent, though not all with equal probability (this latter property will turn out to be the key point in attaining reliable communication). In other words, when $(b + c)$ bits are sent, then any of the possible 2^{b+c} bit sequences can be received with non-zero probability. It is thus impossible to construct an ideal error check when the LLChannel is a BSC and $(b + c)$ is a finite number, since there may always be a bit error pattern that tricks Yoshi into falsely assuming that he correctly decodes ω_j when Xia actually sends ω_i.

More formally, let P_d be the probability that a packet is erased, that is, an error is successfully detected. Let P_u denote the probability of undetected error, described as an event in which Xia sends ω_i, Yoshi receives ω_j with $j \neq i$. Strictly speaking, and in line with the discussion above, no positive throughput can be achieved over the BSC. However, in practice, P_u can be made to be several orders of magnitude lower than P_d, such that we can approximate the error detection to operate as an ideal one. Given that, Yoshi can map all the erased packet to a single symbol, thus creating a channel that has a set of $2^b + 1$ outputs, out of which 2^b outputs correspond to the valid packets and a single output that stands for an erased packet. We have thus obtained a generalization of the binary erasure channel, where a packet will be erased if there is at least one error in the $(b + c)$ bits, such that the probability of erasure is:

$$P_\mathrm{e} = 1 - (1 - p)^{b+c} = P_\mathrm{d} + P_\mathrm{u} \approx P_\mathrm{d} \tag{6.18}$$

since $P_\mathrm{u} \approx 0$. The *nominal data rate* of this channel, expressed in terms of the channel uses of the BSC, corresponds to the data rate that Xia obtains if there are no errors in the channel. This allows us to find the ideal goodput, calculated only by using the contribution from the information carrying bits:

$$R = \frac{b}{b + c} \quad \text{(bits/c.u.)}. \tag{6.19}$$

If we assume that Xia gets an ideal feedback from Yoshi and therefore learns if the packet has been erased or not, then a fraction of $(1 - P_\mathrm{e})$ packets goes through, such that the goodput is:

$$\mathrm{G} = R(1 - P_\mathrm{e}) = \frac{b}{b + c}(1 - P_\mathrm{e}) \quad \text{(bits/c.u.)}. \tag{6.20}$$

Coming back to the layering discussion, we can think of the packet erasure channel created in this way as being an HLChannel that is based on $(b + c)$ uses of the BSC LLChannel.

In reality, P_u is not zero and it can happen that an error goes undetected. The approach to tackling this is to have extra error checks at the higher layers; this implies that the b data bits are carrying less than b bits of information, as a fraction of them are error detection bits applied by the higher layer to check the integrity of the data. As a general rule, whenever error detection is put towards the lower layers, an overhead is introduced for every packet sent at the lower layer, which increases the overall overhead. Nevertheless, the advantage is that an error is detected with a lower delay. Conversely, when error detection is placed at the higher layers, the receiver needs to wait to receive all the data parts and, if an error is detected and there is an ideal feedback to the sender, then all of the data needs to be retransmitted.

The LLChannel can be generalized to the case in which it has Sary input and Sary output; for example, $S = 4$ for QPSK. Since a single channel use can carry $\log_2 S$ bits, the number of channel uses required to send a packet with b bits and c check bits is $L = \frac{b+c}{\log_2 S}$. The nominal data rate is:

$$R = \frac{b}{L} = \frac{b}{b + c}\log_2 S \quad \text{(bits/c.u.)}. \tag{6.21}$$

The HLChannel constructed in this way has $S^L = 2^{b+c}$ different outputs. By using an ideal error check, only 2^b of those outputs are accepted as correct data packets, while all the remaining received symbols are mapped into the erased symbol.

6.5.3 Asynchronous Packets and Transmission of "Nothing"

The communication with packets, as described in the previous section, makes an implicit assumption that Xia and Yoshi are synchronized with respect to the packet transmission. In other words, Xia and Yoshi have a prior agreement at which channel use the new packet transmission starts and how many channel uses belong to the same packet. We will not go through all the details of link establishment and how Xia and Yoshi agree on the communication parameters; the reader can use Chapter 1 in order to get an idea about the control packets that are necessary for this process. Here we will address a more fundamental issue, which sets the basis for relating the definition of a communication channel to the problem of sending a particular type of control messages and packets.

A common view on wireless packet transmission is that if the sender is not transmitting, the receiver gets "nothing" and observes an idle channel. This mode of operation is not supported by the model of synchronous packet transmission, which underlies the discussion from the previous section, since each channel use over the binary symmetric channel is a valid transmitted symbol and it is not possible to transmit "nothing". This follows from Shannon's model of communication from Figure 6.1, which involves the tacit assumption that Xia and Yoshi have already agreed to communicate and every symbol that is sent over the channel is intentionally sent by Xia, there is no unintentional transmission of an idle symbol. Then, how can we model the asynchronous packet transmission, in which the start instant of a packet is not predefined? How to define a transmission of "nothing", which can be interrupted by a packet transmission? In that case, the receiver first needs to detect existence of a packet and then decode the information bits carried in the packet.

We use the layered framework depicted in Figure 6.9. Xia has the DataSender as an entity that is the user of the packet channel and, in a given channel use, DataSender passes b data bits to the PacketSender. The role of the PacketSender is to create a packet by supplementing the bits received from DataSender with r bits to denote the packet start/stop and c bits for error checking. After that, PacketSender passes on $r + c + b$ bits for transmission over the

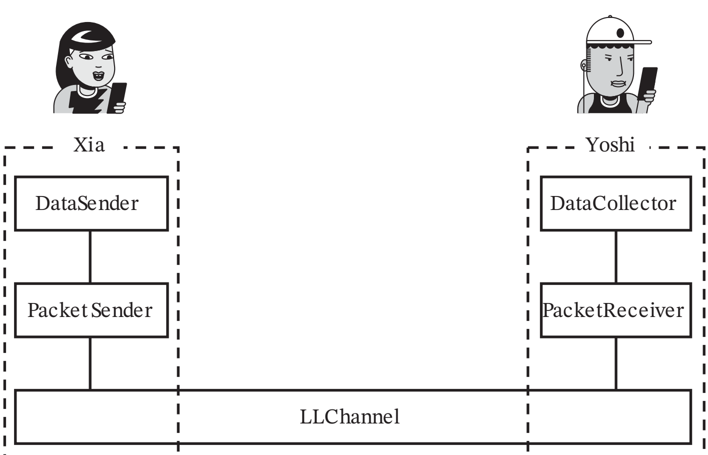


Figure 6.9 Layered model for asynchronous packet transmission.

LLChannel. Yoshi has a PacketReceiver that needs to detect if there is a packet and, if yes, collects the $c + b$ bits, runs an error check and, if the packet is correctly received, it passes the data bits to the DataCollector. If the packet is not correct, then the PacketReceiver passes on an erasure signal to the DataCollector. In the following we will investigate how to enable the PacketSender to send packets to the PacketReceiver asynchronously, when requested by the DataSender. This boils down to the choice of the r bits that need to mark the packet existence and duration. Note that, if the packet duration is fixed and known in advance, then the r bits are only required to denote the start of the packet; those bits are often termed *preamble*.

Let us at first assume that the low layer channel (LLChannel) is a binary channel, not necessarily symmetric. We can think of a channel based on non-coherent transmission: the symbol 0 is sent during a given channel use by staying silent, while power transmission in that channel use corresponds to sending 1. The symbol value 0 has a dual role: it can represents an idle state (no ongoing packet transmission) as well as the bit value 0. In other words, when there is no packet transmission and there are no errors in the channel, the receiver gets 000 …. The next transmitted packet interrupts this zero sequence and this should be detected by Yoshi. If there are no errors in the LLChannel, a preamble consisting of a single 1 is sufficient to reliably interrupt the idleness (transmission of “nothing”) and mark the packet start. However, if there are errors in the channel, then the receiver may experience false positives and incorrectly start to receive a packet. Therefore, in practice, the preamble consists of multiple symbols in order to reliably delimit the packet start, but it is clear that it is always possible to have transmission errors and thereby an imperfect preamble detection.

The problem of detecting existence/start of a packet is known as *frame synchronization*. A simple variant of the problem is illustrated in Figure 6.10. It is assumed that a packet

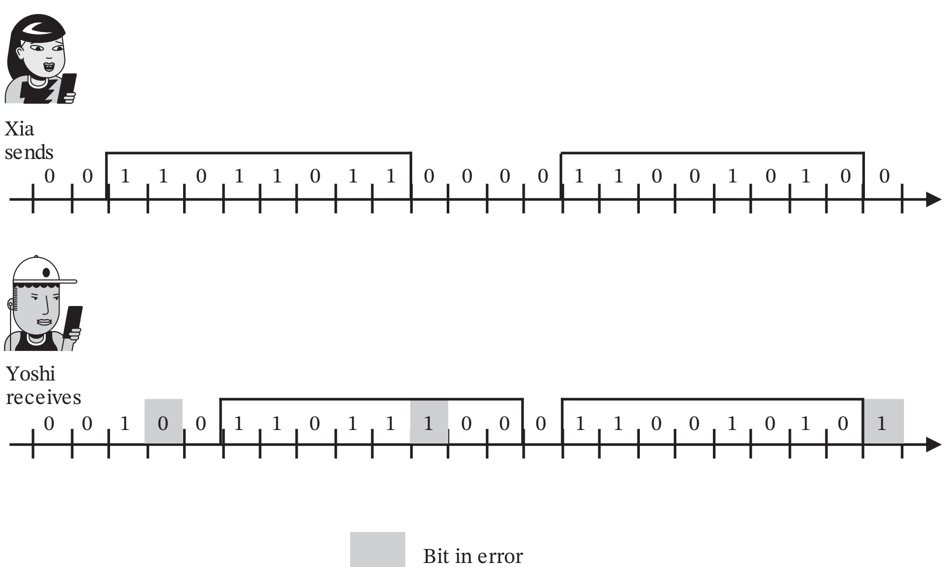


Figure 6.10 An illustration of the frame synchronization problem with packets of length 8 that include a preamble 11, five data bits, and a parity check bit.

sent by Xia has a predefined length of 8 bits, such that there is no need for a group of bits within the packet that describes the packet length. At this point it is useful to recall the discussion on making packets in the discussion in Section 3.5. The 8 bits include: $r = 2$ bits for the preamble 11, $b = 5$ data bits passed on by the DataSender and a $c = 1$ parity check bit for error detection. For the example in Figure 6.10, the first packet 11011011 of Xia is not detected correctly by Yoshi; instead Yoshi detects a "phantom" packet with content 11011100; however, the group of bits 011100 will not pass the parity check and will be discarded or Yoshi will request a retransmission through feedback to Xia. Nevertheless, if the probability to receive 1 if 0 is sent is $P(1|0) > 0$, then there is always a chance that an error goes undetected and an erroneous packet is delivered to the DataCollector. Reliability can be improved if additional error check is used at the higher layer at which the DataSender communicates with the DataCollector. In that case, the total data that needs to be sent by the DataSender is supplemented by an error check, such that the erroneous data packet delivered from the PacketReceiver to the DataCollector will be detected when the DataCollector runs an error check over the total received data.

6.5.4 Packet Transmission over a Ternary Channel

Another model that can be used to treat asynchronous data transmission is to have TXbaseband for which the signals used for 0 and idle are different. For example, when TXbaseband sends data, it uses uses non-zero power and BPSK modulation with input symbols $\{-1, 1\}$. When the TXbaseband is idle, it sends zero power, which corresponds to a baseband symbol 0. Now the analog channel that resides in the lowest layer has a *ternary* input $x \in \{-1, 0, 1\}$, while the output is obtained after adding Gaussian noise to the input. Following the discussion in Section 6.4 about making a digital channel by using an underlying Gaussian channel, the receiver of Yoshi uses a detecting device that operates as follows. For each received symbol, the detecting device produces a discrete output y_{d} with the following possible values:

- $y_{\mathrm{d}} = 0$, if the detecting device believes that -1 has been sent
- $y_{\mathrm{d}} = 1$, if the detecting device believes that 1 has been sent
- $y_{\mathrm{d}} = \epsilon$, if the detecting device believes that the channel is idle.

The question now is how to carry out frame synchronization for a channel with ternary input and output. One may conjecture that this case is somewhat easier compare to the case in which 0 is interpreted as idle ϵ, as now there is a dedicated input symbol to denote an idle channel. A straightforward way to use this channel for sending packets asynchronously would be to set the input to ϵ whenever there is no packet and send either 0 or 1 when there is a packet. However, note that the detecting device is looking only at a single received symbol y at a time and it is agnostic to the fact whether that symbol belongs to a packet or to an idle period. For example, noise can cause y to be detected as ϵ; this results in a channel use that is detected as idle in the middle of a packet. Conversely, the receiver may detect 0/1 when the channel is idle.

The latter observation leads to another idea to create a different type of a discrete communication channel on top of the Gaussian channel. For convenience, let us also put a mapping device at the input of the baseband channel in Xia's transmitter. The discrete input to this

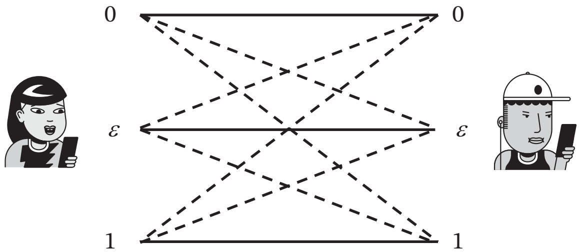


Figure 6.11 Description of the channel with ternary inputs and outputs, where the idle channel symbol is also a valid, information carrying input.

Table 6.2 Possible mapping of three data bits into two ternary symbols.

Bit combination	Transmitted symbols
000	00
001	01
010	0ϵ
011	10
100	11
101	1ϵ
110	$\epsilon 0$
111	$\epsilon 1$

device is denoted by x_d and it can get three possible values $x_d \in \{0, 1, \epsilon\}$, where ϵ is a valid discrete input symbol that also has the role of an idle symbol. The mapping between x_d and the baseband symbol x is given as follows:

$$x_d \to x: \quad 0 \to -1; \quad 1 \to 1; \quad \epsilon \to 0. \tag{6.22}$$

The new insight offered by this channel is that Xia can use the idle symbol ϵ as a legitimate symbol to send data. A single use of the channel is described in Figure 6.11; the transition probabilities can be determined in a way that is similar to the one we have used to create digital channels from analog channels with Gaussian noise. Since ϵ is now also an information carrying symbol, we need to use a preamble for frame synchronization. For simplicity, let us assume that the packet preamble is 11 and it is followed by five information bits and a parity check bit, as for the example in Figure 6.10. If only BPSK is used for packet transmission, then Xia needs to send 6 inputs 0/1. However, if ϵ may also be used to send information, then each group of three bits can be encoded into two ternary symbols $0, 1, \epsilon$ by following Table 6.2. Therefore, a single packet, including the preamble, now consumes a total of 6 channel uses instead of 8, provided that ϵ is also a legitimate data carrying symbol.

The latter example illustrates the conceptual separation between the data networking community and the information–theoretic view on the communication channel. While in data networking the approach is always to use only the "legal" symbols to transmit data packets, the approach in information theory is to use all the possibilities provided by a given communication model. Keeping the erasure symbol only to denote an idle channel could be more practical from the viewpoint of implementation, energy consumption, or communication architecture. However, if our goal is to send the maximal amount information over the communication channel, one needs to treat the idle symbol as another modulation/input symbol for sending data. More on the information–theoretic notion of channel capacity in Chapter 8.

6.6 Chapter Summary

The notion of a communication channel can be made mathematically precise by specifying the channel inputs, outputs, the probabilistic relation between them, as well as any constraint imposed on the way the channel is used. This chapter has emphasized the importance of understanding a communication channel as a mathematical object that can be defined at any layer of the communication system, as long as there is an agreement what we, as system designers, are unwilling or unable to change. This chapter has also illustrated the following property of a communication channel: the way the channel is defined is tightly related to the assumptions about which knowledge is available at the transmitter and the receiver. This aspect is an essential bridge between the proper engineering assumptions about the system in question and the mathematical model used to design and optimize communication schemes. A communication channel at a higher layer can be constructed by using one or more channel uses of a channel defined at a lower layer. An example of this is the creation of a BSC with binary inputs and outputs by using an underlying Gaussian channel. The chapter has illustrated other possible mappings between the data bits and various baseband modulation schemes. The last part of the chapter makes a connection between the communication channel, as a mathematical object, and the data packets, as main entities in the communication protocols.

6.7 Further Reading

A must-read for understanding the fundamentals and the idea behind the mathematical models used in communication engineering is the first paper by Shannon [1948]. For the readers that are interested to explore further the information–theoretic treatment of communication channels, two standard references are Gallager [1968] and Cover and Thomas [2012]. The latter also covers mathematical models for multi-user channels, which have not been discussed here. A very insightful reading on communication channels and information theory can be found in the Lecture Notes by Massey in Massey [1980] (Chapter 4 discusses the channel as an entity that one is “unwilling or unable to change”). A primer on a definition of a channel based on the underlying packets (which already carry data from other communication channels) is given in Anantharam and Verdu [1996], while a wider discussion on the relation between mathematical models for communication channels and communication networks is given in Ephremides and Hajek [1998]. Communication channels are universal in nature and technology; the reader is referred to Nakano et al. [2012] for the more recent interest in molecular communication and nano-communication channels.

6.8 Problems and Reflections

1. *Using all channel outputs versus combining the outputs*. Zoya transmits through a wireless communication channel to Walt. She transmits one complex symbol z every T_s s. There are two different receiver configurations for Walt:

   Configuration 1. Every T_s s Walt gets a noisy version of the symbol sent by Zoya, such that for the ith transmitted symbol of Zoya, Walt gets the ith received symbol $w_i = z_i + n_i$, where n_i is a Gaussian noise.

   Configuration 2. At Walt's receiver there is an additional module that combines two noisy received symbols into one, such that Walt gets one new received symbol every $2T_s$ s. Thus, if Zoya transmits $z_1, z_2, z_3, \ldots$, Walt gets:

   $$\begin{aligned} w_1 &= z_1 + z_2 + n_1 + n_2 \\ w_2 &= z_3 + z_4 + n_3 + n_4 \\ &\vdots \; \vdots \; \vdots \\ w_{i/2} &= z_{i-1} + z_i + n_{i-1} + n_i. \end{aligned}$$

   Compare the two channels and argue which one has a larger capability to carry information.
2. *Communication channel based on packets*. Zoya transmits data to Yoshi through data packets. Each packet consists of D bits of data and a header. Each transmitted packet is either received without errors by Yoshi or is lost and never arrives at Yoshi. The packets that arrive at Yoshi arrive in order, such that if Zoya sends packet l before the packet $l + 1$, then it cannot happen that Yoshi receives first the packet $l + 1$ and then the packet l. Discuss the differences between the two communication channels that can be defined for the following two cases:
   (a) The header of a packet contains "Yoshi" as a destination as well as the ID of the transmitted packet.
   (b) The header of a packet only contains "Yoshi" as a destination, but there is no packet ID transmitted.

   Hint: The number of inputs for the communication channel is 2^D.
3. *Communication with timed packets*. This is similar to the previous problem, but here Zoya and Yoshi agree that she sends one packet with D bits each T_p s. Each packet either arrives immediately to Yoshi or is lost. Discuss the differences between the two communication channels that can be defined for the following two cases, as well as the difference with the two channels from the previous problem:
   (a) The header of a packet contains "Yoshi" as a destination as well as the ID of the transmitted packet.
   (b) The header of a packet only contains "Yoshi" as a destination, but there is no packet ID transmitted.

   Hint: Zoya can get an additional channel input by deciding not to send a packet. If the packet ID is available, think of how Zoya can use that by intentionally not sending the packets in the proper order.
4. *Signaling with antenna activation*. Xia has two transmit antennas, denoted by TXA_1 and TXA_2, while Yoshi has two receive antennas denoted by RXA_1 and RXA_2, respectively.

At a given time, Xia can only transmit through one of the antennas. When Xia transmits through TXA_1, only RXA_1 receives a signal, while there is no signal at RXA_2. When Xia transmits through TXA_2, both antennas of Yoshi receive signals. Establish a mathematical model for this communication channel and devise communication strategies for the following cases:

(a) Each of the receiving antennas can only detect if there is power or not. Assume that there is no noise.

(b) Each transmission of Xia is a QPSK modulated signal, while the received signal has an added Gaussian noise. When Xia sends x through TXA_1, then the received signal at RXA_1 is $x + n_1$, while RXA_2 stays idle. When Xia sends x through TXA_2, then the received signal at RXA_1 is $x + n_1$ and the received signal at RXA_2 is $x + n_2$. Here n_i is a Gaussian noise.

5. *Bit reliabilities with 16-QAM*. Consider the Gray mapped 16-QAM depicted in Figure 6.5. For a fixed SNR, study the reliability of each of the 4 bits transmitted with this modulation. Verify the statement given in the text that the four binary channels created with a 16-QAM transmission are not four identical and independent binary symmetric channels.