Datacomm Final Notes

EXAMPLE 16.6 Consider a (7, 4) code with the generator polynomial $P(X) = X^3 + X^2 + 1$. We have $7 = 2^3 - 1$, so this code is capable of correcting all single-bit errors. Table 16.1a lists all of the valid codewords; note that $d_{\min}$ is 3, confirming that this is a single-error-correcting code. For example, for the data block 1010, we have $D(X) = X^3 + X$ and $X^{n-k}D(X) = X^6 + X^4$. Dividing as in Equation (16.7):

$$
\begin{array}{rl}
 & X^3 + X^2 + 1 \qquad\qquad \leftarrow Q(X) \\
P(X) \rightarrow X^3 + X^2 + 1 \;\big/ & X^6 \qquad X^4 \qquad\qquad \leftarrow 2^3 D(X) \\
 & \underline{X^3 + X^5 + \qquad X^3} \\
 & X^5 + X^4 + \quad X^3 \\
 & \underline{X^5 + X^4 + \qquad X^2} \\
 & X^3 + X^2 \\
 & \underline{X^3 + X^2 + \quad 1} \\
 & 1 \quad \leftarrow C(X)
\end{array}
$$

Then, using Equation (16.6), we have $T(X) = X^6 + X^4 + 1$, which is the codeword 1010001.

For error correction, we need to construct the syndrome table shown in Table 16.1b. For example, for an error pattern of 1000000, $E(X) = X^6$. Using the last line of Equation (16.7), we calculate:

$$
\begin{array}{rl}
 & X^3 + X^2 + X \qquad\qquad \leftarrow Q(X) + B(X) \\
P(X) \rightarrow X^3 + X^2 + 1 \;\big/ & X^6 \qquad\qquad\qquad \leftarrow E(X) \\
 & \underline{X^3 + X^5 + \qquad X^3} \\
 & X^5 + \qquad X^3 \\
 & \underline{X^5 + X^4 + \qquad X^2} \\
 & X^4 + \quad X^3 + X^2 \\
 & \underline{X^4 + \quad X^3 + \qquad X} \\
 & X^2 + X \quad \leftarrow S(X)
\end{array}
$$

Therefore, $S = 110$. The remaining entries in Table 16.1b are calculated similarly. Now suppose the received block is 1101101, or $Z(X) = X^6 + X^5 + X^3 + X^2 + 1$. Using Equation (16.8):

$$
\begin{array}{rl}
 & X^3 \qquad\qquad\qquad \leftarrow B(X) \\
P(X) \rightarrow X^3 + X^2 + 1 \;\big/ & X^6 + X^5 + \qquad X^3 + X^2 + 1 \quad \leftarrow Z(X) \\
 & \underline{X^6 + X^5 + \qquad X^3} \\
 & X^2 + 1 \quad \leftarrow S(X)
\end{array}
$$

Thus $S = 101$. Using Table 16.1b, this yields $E = 0001000$. Then,

$$T = 1101101 \oplus 0001000 = 1100101$$

Then, from Table 16.1a, the transmitted data block is 1100.

EXAMPLE 16.7 Let $t = 1$ and $m = 2$. Denoting the symbols as 0, 1, 2, 3 we can write their binary equivalents as 0 = 00; 1 = 01; 2 = 10; 3 = 11. The code has the following parameters.

$$n = 2^2 - 1 = 3 \text{ symbols} = 6 \text{ bits}$$
$$(n - k) = 2 \text{ symbols} = 4 \text{ bits}$$

This code can correct any burst error that spans a symbol of 2 bits.

Parity-Check Matrix Codes

An important type of FEC (forward error correction) is the low-density parity-check code (LDPC). LDPC codes are enjoying increasing use in high-speed wireless specifications, including the 802.11n and 802.11ac Wi-Fi standards and satellite digital television transmission. LDPC is also used for 10-Gbps Ethernet. LDPC codes exhibit performance in terms of bit error probability that is very close to the Shannon limit and can be efficiently implemented for high-speed use.

Before discussing LDPC, we introduce the more general class of parity-check codes, of which LDPC is a specific example. We discuss LDPC codes in the following section.

Consider a simple parity bit scheme used on blocks of n bits, consisting of $k = n - 1$ data bits and 1 parity-check bit, and that even parity is used. Let c_1 through c_n–1 be the data bits and c_n be the parity bit. Then the following condition holds:

$$c_1 \oplus c_2 \oplus \ \dots \ \oplus c_n = 0 \tag{16.10}$$

where addition is modulo 2 (equivalently, addition is the XOR function). Using the terminology from our discussion of block code principles in Chapter 6, there are 2^n possible codewords, of which 2^{n-1} are valid codewords. The valid codewords are those that satisfy Equation (16.10). If any of the valid codewords is received, the received block is accepted as free of errors, and the first n–1 bits are accepted as the valid data bits. This scheme can detect single-bit errors but cannot perform error correction.

We generalize the parity-check concept to consider codes whose words satisfy a set of $m = n - k$ simultaneous linear equations. A **parity-check code** that produces n-bit codewords is the set of solutions to the following equations:

$$\begin{aligned} h_{11}c_1 \oplus h_{12}c_2 \oplus \ \dots \ + h_{1n}c_n &= 0 \\ h_{21}c_1 \oplus h_{22}c_2 \oplus \ \dots \ \oplus h_{2n}c_n &= 0 \\ &\vdots \\ h_{m1}c_1 \oplus h_{m2}c_2 \oplus \ \dots \ \oplus h_{mn}c_n &= 0 \end{aligned} \tag{16.11}$$

where the coefficients h_{ij} take on the binary values 0 or 1.

The $m \times n$ matrix $\mathbf{H} = [h_{ij}]$ is called the **parity-check matrix**. Each of the m rows of $\mathbf{H}$ corresponds to one of the individual equations in (16.11). Each of the n columns of $\mathbf{H}$ corresponds to one bit of the codeword. If we represent the codeword by the row vector $\mathbf{c} = [c_j]$, then the equation set (16.11) can be represented as:

$$\mathbf{Hc^T} = \mathbf{cH^T} = \mathbf{0} \tag{16.12}$$

An (n, k) parity-check code encodes k data bits into an n-bit codeword. Typically, and without loss of generality, the convention used is that that leftmost k bits of the codeword reproduce the original k data bits and the rightmost $(n - k)$ bits are the check bits (Figure 16.5). This form is known as a **systematic code**. Thus, in the parity-check matrix $\mathbf{H}$, the first k columns correspond to data bits and the remaining columns to check bits. To repeat what was said in Chapter 6, with an (n, k) block code, there are 2^k valid codewords out of a total of 2^n possible codewords. The ratio of redundant bits to data bits, $(n - k)/k$, is called the **redundancy** of the code, and the ratio of data bits to total bits, k/n, is called the **code rate**. The code rate is a measure of how much additional bandwidth is required to carry data at the same data rate as without the code.

The fundamental constraint on a parity-check code is that the code must have $(n - k)$ linearly independent equations. A code may have more equations, but only $(n - k)$ of them will be linearly independent. Without loss of generality, we can limit H to have the form:

$$\mathbf{H} = [\mathbf{A}\ \mathbf{I}_{n-k}]$$

where $\mathbf{I}_{n-k}$ is the $(n - k) \times (n - k)$ identity matrix, and $\mathbf{A}$ is a $k \times k$ matrix. The linear independence constraint is satisfied if and only if the determinant of $\mathbf{A}$ is nonzero.[1] With this constraint, the k data bits may be specified arbitrarily in the equation set (16.11). The set of equations can then be solved for the values of the check bits. Put another way, for each of the 2^k possible sets of data bits, it is possible to uniquely solve equation set (16.11) to determine the $(n - k)$ check bits.

Consider a (7, 4) check code defined by the equations:

$$\begin{aligned}
c_1 \oplus c_2 \oplus c_3 \oplus c_5 &= 0 \\
c_1 \oplus c_3 \oplus c_4 \oplus c_6 &= 0 \\
c_1 \oplus c_2 \oplus c_4 \oplus c_7 &= 0
\end{aligned} \tag{16.13}$$

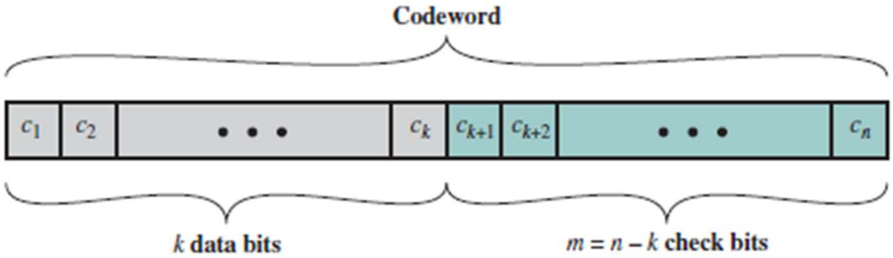


Figure 16.5 Structure of a Parity Check Codeword

Using the parity-check matrix, we have:

$$\underbrace{\begin{bmatrix} 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}}_{\mathbf{H}} \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \\ c_5 \\ c_6 \\ c_7 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{16.14}$$

For a parity-check code, as with any FEC, there are three functions we need to perform:

- **Encoding:** For a given set of k data bits, generate the corresponding n-bit codeword.
- **Error detection:** For a given codeword, determine if there are one or more bits in error.
- **Error correction:** If an error is detected, perform error correction.

ENCODING For our example, to form a codeword, we first choose values for data bits c_1, c_2, and c_3; for example, $c_1 = 1$, $c_2 = 1$, $c_3 = 0$, $c_4 = 0$. We then solve equation set (16.13) by rewriting them so that we show each check bit as a function of data bits:

$$\begin{aligned} c_5 &= c_1 \oplus c_2 \oplus c_3 \\ c_6 &= c_1 \oplus c_3 \oplus c_4 \\ c_7 &= c_1 \oplus c_2 \oplus c_4 \end{aligned} \tag{16.15}$$

Thus the codeword is 1100010. With three information bits, there are a total of 16 valid codewords out of the $2^7 = 128$ possible codewords. We can solve for each of the 16 possible combinations of data bits to calculate these codewords. The results are shown in Table 16.4.

A more general approach to encoding is to create a $k \times n$ **generator matrix** for the code. Using equation set (16.13), we can write:

$$[c_1 \quad c_2 \quad c_3 \quad c_4 \quad c_5 \quad c_6 \quad c_7] = [c_1\ c_2\ c_3\ c_4] \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{bmatrix}}_{\mathbf{G}} \tag{16.16}$$

By our convention, the first k bits of $\mathbf{c}$ are the data bits. Let us label the data bits of $\mathbf{c}$ as $\mathbf{u} = [u_i]$ where $u_i = c_i$, for $i = 1$ to k. Then, the codeword corresponding to a data block is determined by:

$$\mathbf{c} = \mathbf{uG} \tag{16.17}$$

FİNAL İÇİN SORU ÇÖZÜMÜ

Q1)

Suppose users share a 500 Mbps link and each user requires 25 Mbps when transmitting, but each user is active (transmits) only 20 percent of the time.

a) Suppose circuit switching is used, how many users can be supported?

b) Suppose packet switching is used. There are 40 users.

i) Find the probability that a given user is transmitting.

ii) What fraction of the link capacity will be used when one user is transmitting?

iii) Find the probability that a given user is and the remaining users are not transmitting?

iv) Find the probability that one user (anyone among 40 users) is and the remaining users are not transmitting?

v) Find the probability that there are 20 users (among 40 users) transmitting simultaneously and the remaining users are not transmitting.

vi) Find the probability that there are 21 or more users transmitting simultaneously. Discuss the effectiveness of utilization of circuit switching and packet switching.

vii) If each user transmits only 50 percent of the time, find the probability that there are 21 or more users transmitting simultaneously. Discuss the effectiveness of utilization of circuit switching and packet switching taking into consideration the results of option **f)**.

Q3)

The reference loss values are given below for two different transmission media. For Medium-1, there is a constant loss per km. For Medium-2, we assume that only free space loss is the destructive effect. Please calculate missing loss values in the following table. **Please demonstrate your calculations.**

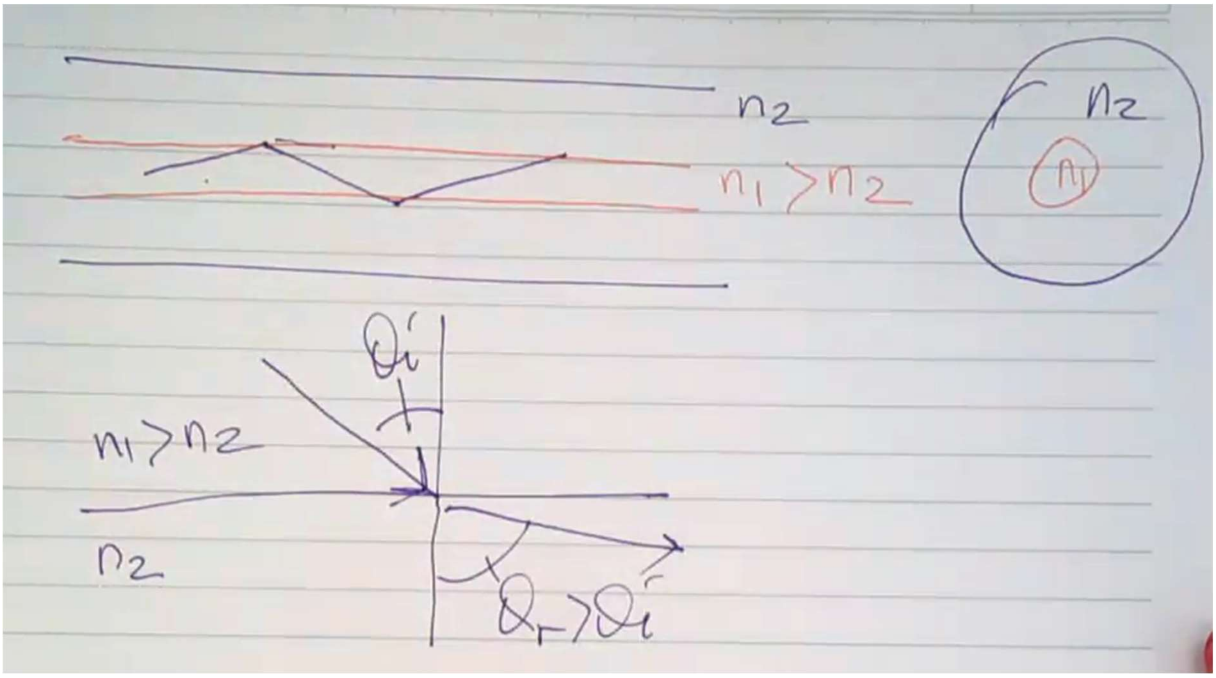
Distance	1 km	4 km	8 km	32 km
Medium-1 (dB)	**-2**			
Medium-2 (dB)	**-4**			

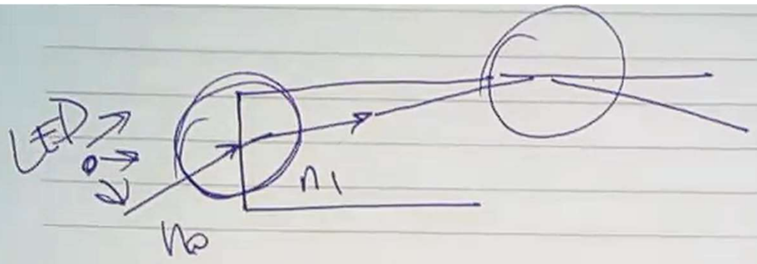
Q2)

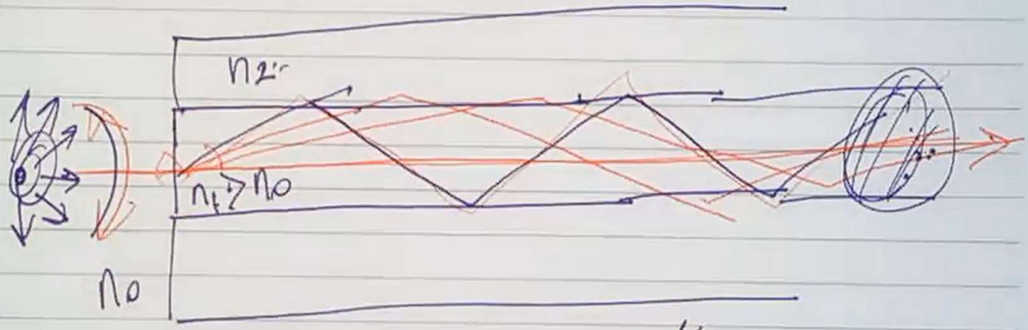
Capacity in additive white Gaussian channel (AWGN) is given by
$C = B\log_2(1 + P/(N_0B))$. The bandwidth (B)1 MHz and noise power spectral density (PSD) $N_0/2 = 0.5x10^{-10}$ W/Hz are given. The transmit power is equal to 1W for the wireless channel with isotropic antennas and a carrier frequency 100 MHz. The distance between transmitting and receiving antennas is equal to 10 m.

a) Find the value of P.
b) Calculate the capacity.
c) How much does capacity increase by doubling the transmit power?
d) How much does capacity increase by doubling the channel bandwidth?
e) Find capacity in the limit of infinite bandwidth $B \to \infty$ as a function of P.

FIBER OPTIC PART







$$NA = n_0 \sin\theta_i = (n_1^2 - n_2^2)^{1/2}$$

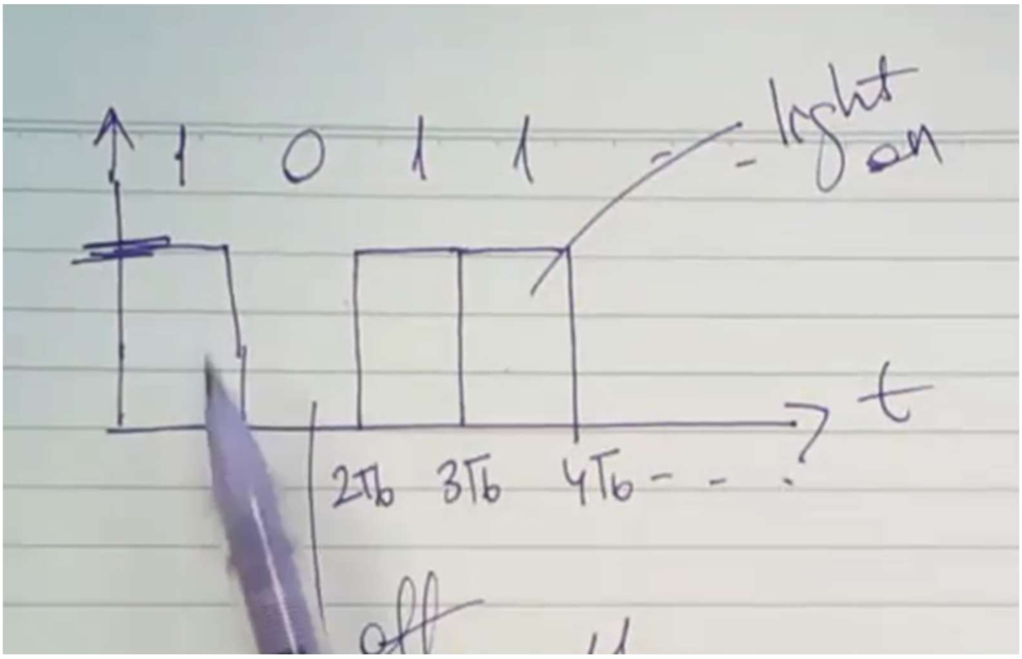
$$\Delta T = \frac{\left[\left(\frac{L}{\sin\phi_c}\right) - (L)\right] n_1}{c} \quad [s]$$

$$v = \frac{1}{\sqrt{\varepsilon\mu}}$$

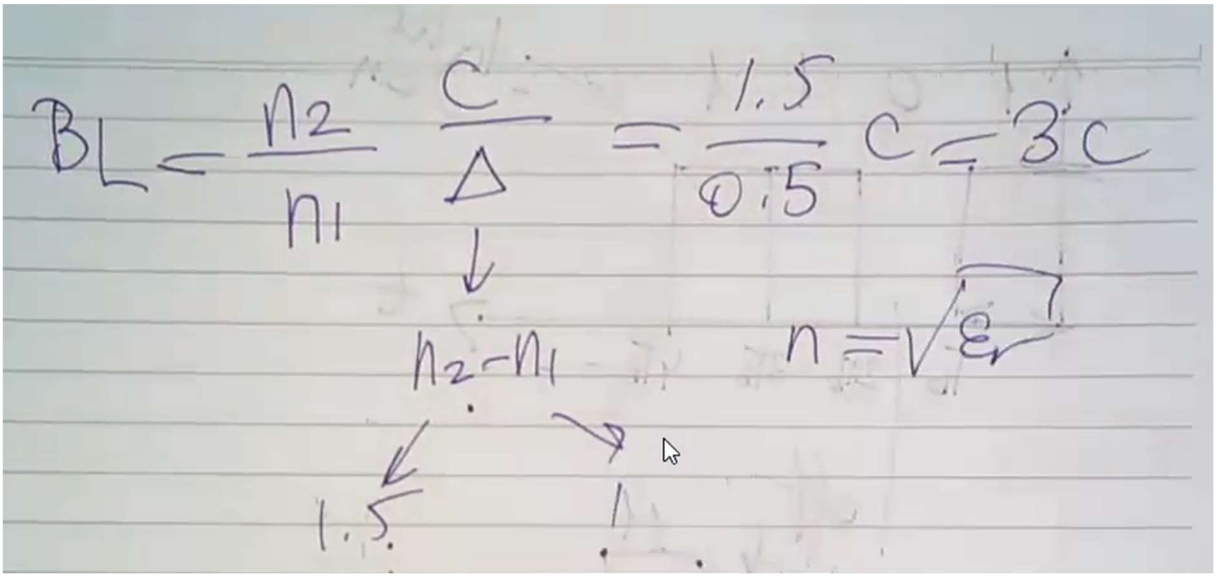
$$v = \frac{c}{\sqrt{\varepsilon_{r_1}}} \; n_1$$

$$c = \frac{1}{\sqrt{\varepsilon_0 \mu_0}}$$

$$\Delta T = \frac{n_1}{c}\left(\frac{L}{\sin\phi_c} - L\right) = \frac{L}{c}\frac{n_1^2}{n_2}\,\Delta.$$



EXAMPLE



$n_2 = \cancel{1.5}$ $n_1 = 1.5$

$$BL = \frac{1}{1.5} \cdot \frac{c}{0.5} = 1.33c \quad \rightarrow 3 \cdot 10^8 \, m/s$$

$$=$$

$$BL = 400 \, Mbps \cdot \cancel{mi} = 0.4 \, Mbps \cdot km$$

$$B = \frac{B \cdot L}{L} = \frac{0.4 \, Mbps \cdot km}{10 \, km} = 0.04 \, Mb/s$$

$$B = 40 \, kbps$$

$n_1 = 1.5$ $\Delta = 0.01$

$$BL < \frac{8c}{n_1 \Delta^2} = \frac{8 \cdot 3 \cdot 10^8 \, m/s}{1.5 \cdot 10^{-4}} = 16 \, Gbps \cdot km \quad / \, 10 \, km$$

$$BL < 16 \cdot 10^{12} \, bps \cdot m$$

$$\boxed{B = 1.6 \, Gbps}$$

EXAMPLE 2

$n_1 = 1.5$ $\quad \Delta = 0.01$ $\quad n_2 = 1.49$

$$BL < \frac{c}{n_1 \Delta^2} = \frac{3 \cdot 10^8 \ m/s}{1.5 \cdot 10^{-4}} = 16 \ Gbps \cdot km$$

$$BL < 16 \cdot 10^{12} \ bps \cdot m$$

10km

$$\boxed{B = 1.6 \ Gbps}$$

Spot Size

Since the field distribution given in Eq. (2.2.36) is cumbersome to use in practice, it is often approximated by a Gaussian distribution of the form

$$E_x = A \exp(-\rho^2/w^2) \exp(i\beta z), \tag{2.2.38}$$

Spot Size

Since the field distribution given in Eq. (2.2.36) is cumbersome to use in practice, it is often approximated by a Gaussian distribution of the form

$$E_x = A \exp(-\rho^2/w^2) \exp(i\beta z), \tag{2.2.38}$$

where w is the *field radius* and is referred to as the *spot size*. It is determined by fitting the exact distribution to the Gaussian function or by following a variational procedure [19]. Figure 2.7 shows the dependence of w/a on the V parameter. A comparison of the actual field distribution with the fitted Gaussian is also shown for $V = 2.4$. The quality of fit is generally quite good for values of V in the neighborhood of 2. The spot size w can be determined from Figure 2.7. It can also be determined from an analytic approximation accurate to within 1% for $1.2 < V < 2.4$ and given by [19]

$$w/a \approx 0.65 + 1.619 V^{-3/2} + 2.879 V^{-6}. \tag{2.2.39}$$

$$\Gamma = \frac{P_{core}}{P_{total}} = 1 - \exp\left\{-2\left(\frac{a}{w}\right)^2\right\}$$

%80

$$e^{-2(a/w)^2} = 0.2$$

$$a/w = 0.89706$$

$$w/a = 1.1147$$

$$w/a \simeq 0.65 + 1.619V^{-3/2} + 2.879V^{-6}$$

$$\boxed{V = 2.356}$$

$\boxed{V = 2.356} < 2.4$ single mode

$$V = \left(\frac{2\pi}{\lambda}\right) a \cdot n_1 \sqrt{2\Delta}$$

λ: 1.3nm

n_1: 1.5

Δ: $(n_1 - n_2)$, 1.10^{-4}

$a = 0.0229$

$= 22.9$ nm