Clemson University
Electrical and Computer Engineering Department
105 Riggs Hall
Clemson, SC 29631

Turbo Product Codes and Channel Capacity[1]

Jessica Pursley

IEEE Member # 41359253

Submitted for consideration in

Region 3, IEEE

Student Paper Competition

Endorsement of Branch Counselor

The author(s) of this paper are student members of this IEEE student branch and will still be undergraduate students at the time of SoutheastCon. This paper is the only entry of our student branch.

Signature:____

Name: John J. Komo

---

[1] This research was supported in part by MIT Lincoln Laboratory and in part by the Office of Naval Research under grant number N00014-00-1-0565 to Clemson University.

Turbo Product Codes and Channel Capacity[1]

Abstract

An important goal for a new communication system is to transfer information at a high rate while keeping energy consumption low. Shannon's channel capacity sets a limit on the tradeoff between the rate of information transfer and the required energy. In this paper the performance of turbo block coding is compared to Shannon's limit for binary communications over a channel with thermal noise. The turbo product code is a relatively new code that offers a wide range of flexibility in terms of performance, complexity, and code rate. The results of this paper will show that in comparison to a standard convolutional code the turbo product code uses less power and permits a higher data rate for a given bandwidth requirement. One of the turbo product codes discussed in this paper has a rate approximately 0.8, which is a 60% improvement in data rate over the standard convolutional code. It is also shown that the performance of the turbo product code is close to Shannon's capacity limit.

Introduction

A communications system is used to transmit information from one point to another. As an illustration, consider what occurs when two people communicate with each other using cell phones. A conversation begins when one person speaks into the phone. The phone converts the voice message into a binary information signal and sends it over a channel to a base station. The base station is connected to other base stations, one of which is in the area of the receiving party. The cellular system locates the receiving party and routes the information signal to the appropriate base station. An important feature of this system, for purposes of this paper, is the noise that is present in the communication channels connecting the cell phones to the base stations. This noise causes errors in the receptions of the signals and the error control coding discussed in this paper is introduced into the system to make the error rate acceptable.

A digital communications system is made up of three main components: the transmitter, the channel, and the receiver. The information produced by a source must be converted to a digital form

---

[1] This research was supported in part by MIT Lincoln Laboratory and in part by the Office of Naval Research under grant number N00014-00-1-0565 to Clemson University.

suitable for the system. The transmitter then couples the information signal to the channel where it undergoes degradation from the transmitter to the receiver, usually resulting from noise and other undesired signals or interference. The receiver's function is to extract the desired signal at the channel output and convert it to a form suitable for the system user.

There are many techniques to consider when designing a communications system. One important measure of performance for any communications system is the rate at which the information is sent. Obviously, it would be advantageous to create a system which transmits data at a high rate. This would allow for signals to be sent faster, reducing the delay and permitting a higher quality voice signal. Unfortunately there is a limit on the rate at which information can be transmitted through a noisy channel. This upper bound is referred to as the capacity of the channel. This limit, commonly referred to as Shannon's limit [5], states that if the transmission rate R is less than the channel capacity C, then there exists a coding technique that enables transmission over the channel with an arbitrarily small error rate [4]. Thus channel capacity is defined as the maximum rate of reliable information transmission through the channel. In order to achieve this maximum, the signal messages must be coded before they are transmitted over the channel.

Error correcting codes are commonly used in today's communications systems. These codes make it possible to detect and correct errors by adding redundant bits to the message. One form of error correcting code is a block code, which encodes blocks of information symbols into blocks of encoded symbols. Such a code is referred to as an (n,k) code where k is the number of information bits per word and n is the number of encoded bits. The difference $n - k$ is the number of redundant bits that are added. The code rate $R = k/n$ is a measure of the information rate relative to the transmitted bit rate on the channel. The available bandwidth limits the transmitted bit rate, so for a fixed bandwidth the higher the rate of the code the higher the rate at which information is delivered to the destination. Thus, it is desirable to find high rate codes that correct large numbers of errors. In addition to having a high information rate, it is desirable for the communication system to operate at the lowest possible transmitter power. A reduction in the transmitted power results in a decrease in interference to other systems. Such a reduction also decreases energy consumption and preserves battery life in mobile communications equipment.

Efficient communication systems are systems that permit a high rate of information to be communicated with the lowest possible power. In this paper, we are interested in codes that meet these objectives. One type of code that has tremendous potential is the turbo product code. Turbo product codes have been found to improve power efficiency and maximize the rate of data transmission. A standard convolutional code that is used today cuts the data rate in half (that is, $R = 0.5$). One of the turbo product codes described in this paper has a rate approximately 0.8, which is a 60% improvement in data rate over the standard convolutional code. As discussed in this paper the turbo product code also permits a lower transmitter power than the standard convolutional code.

Turbo product codes are error-correction codes that offer a wide range of tradeoffs in performance, complexity, and code rate. While they have some similarities to turbo-concatenated codes that have been incorporated into recent standards for future cellular communication, turbo product codes have different structure and characteristics. Turbo product codes are built from product block codes (e.g., pair of extended Hamming codes). Their performance can be within 1 dB of the Shannon limit. While turbo-concatenated codes are found to have an error floor of approximately 10^{-5}, turbo product codes do not have an error floor. This paper will evaluate turbo product code performance and compare results to the Shannon limit.

Turbo Product Code Construction

A turbo product code [1] is a relatively large code built from smaller code word blocks [4]. For example, consider the (16,11) extended Hamming code. This code takes 11 information bits, computes 5 parity bits, and appends these 5 parity bits to the information bits to create a 16-bit code word. In systematic form a code word has the form

I I I I I I I I I I I P P P P P

Here the symbol I denotes an information bit, and P corresponds to a parity bit. A two-dimensional turbo product code may be constructed using a block of information bits as illustrated in Figure 1. The encoder starts with the first row of information bits, calculates and appends the parity bits, denoted P_H, and then moves on to the second row. This is repeated for each row. Next, the encoder starts with the first column of information bits, calculates and appends the parity bits for that column, denoted by P_V, and moves to the next column. Once the information block is complete, the encoder calculates and appends parity bits onto P_H, denoted P_{VH}. It is important to note that $P_{VH} = P_{HV}$ and that different code lengths may be used for the horizontal and vertical blocks. In addition a two-dimensional turbo product code may be expanded to three dimensions. In the illustration of Figure 1, the (8,4) extended Hamming code yields a (64,16) turbo product code.

I	I	I	I	P_H	P_H	P_H	P_H
I	I	I	I	P_H	P_H	P_H	P_H
I	I	I	I	P_H	P_H	P_H	P_H
I	I	I	I	P_H	P_H	P_H	P_H
P_V	P_V	P_V	P_V	P_{VH}	P_{VH}	P_{VH}	P_{VH}
P_V	P_V	P_V	P_V	P_{VH}	P_{VH}	P_{VH}	P_{VH}
P_V	P_V	P_V	P_V	P_{VH}	P_{VH}	P_{VH}	P_{VH}
P_V	P_V	P_V	P_V	P_{VH}	P_{VH}	P_{VH}	P_{VH}

Figure 1 : Construction of (64,16) Turbo Product Code

Turbo Product Code Decoding

The decoding of a turbo product code is performed one block at a time using iterative decoding. The horizontal blocks are decoded first and then all of the vertical blocks are decoded. Iteration can be done several times to reduce the probability of error. Optimal iterative decoding requires soft-decision decoding. This is based on a soft-decision metric, which is a measure of the likelihood or confidence that the decoder has in each of the bits in the received block. Each decoding iteration builds on the previous decoding performance. Many decoders use the soft-output Viterbi algorithm [5], but there are also other alternatives [3].

Simulation Board

Simulating a turbo product code decoder using soft-decision decoding is a very complicated and time-consuming program to run. Thus, it was proposed to use Efficient Channel Coding's (ECC) Turbo Product Code Evaluation Module [2] to simulate the soft-decision decoder. This board plugs into an ISA slot in a personal computer, and it interacts with the user through a Microsoft Visual C++ program. It encodes and decodes using the AHA 4501 chip which is a turbo product code ASIC that was developed by Advanced Hardware Architectures (AHA) [1].

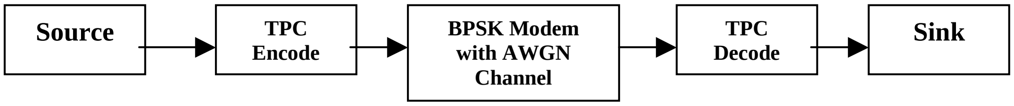

Figure 2 : Block Diagram of BPSK Simulation

Simulation

The modulation used for transmission of the binary data is phase shift keying (PSK). Figure 2 shows the general setup for the binary PSK (BPSK) simulation. Information is generated at the source. It is then sent to the encoder which uses turbo product encoding as explained earlier. The encoded bits are then modulated using BPSK to communicate over a channel in which Gaussian white noise is added. The received block is decoded using soft-decision decoding and sent to the sink which has already received the original information bits from the source. The sink compares the two blocks and calculates the errors.

BLOCK CONFIGURATION	BLOCK SIZE (bits)	DATA SIZE	RATE	CODE STRUCTURE
(64,57)X(64,57)	**4096**	**3249**	**0.793**	**2D**
(32,26)X(32,26)X(4,3)	4096	2028	0.495	3D
(16,11)X(16,11)X(16,11)	**4096**	**1331**	**0.325**	**3D**
(64,57)X(32,26)	2048	1482	0.724	2D
(32,26)X(16,11)X(4,3)	2048	858	0.419	3D
(32,26)X(32,26)	**1024**	**676**	**0.66**	**2D**
(16,11)X(16,11)X(4,3)	1024	363	0.354	3D

Table 1: Code Types Supported by AHA4501

Table 1 shows the block configuration, block size, data size, rate, and code structure for various codes that are considered. The important thing to notice is the wide range of code rates that turbo product codes can provide. Today's standard code has a code rate of 0.5. Three codes in particular have been highlighted and will be referred to by their rates. The performance of the rate 0.793 code which is configured from two (64,57) extended hamming codes, the 0.325 code configured from three (16,11) extended hamming codes, and the 0.66 code which is configured from two (32,26) extended hamming codes will be compared to channel capacity.

The main aspect to notice about the performance of these codes shown in Figure 3 is that they achieve very low error probability at what communication engineers consider to be a very low signal-to-noise ratio. The curve for uncoded binary phase shift keying illustrates that the performance is unacceptable without coding. Without coding more than 1 bit error will occur in every 100 bits, while only 1 error in every 100,000 bits will occur with the rate 0.325 code at a signal-to-noise ratio of 2dB.

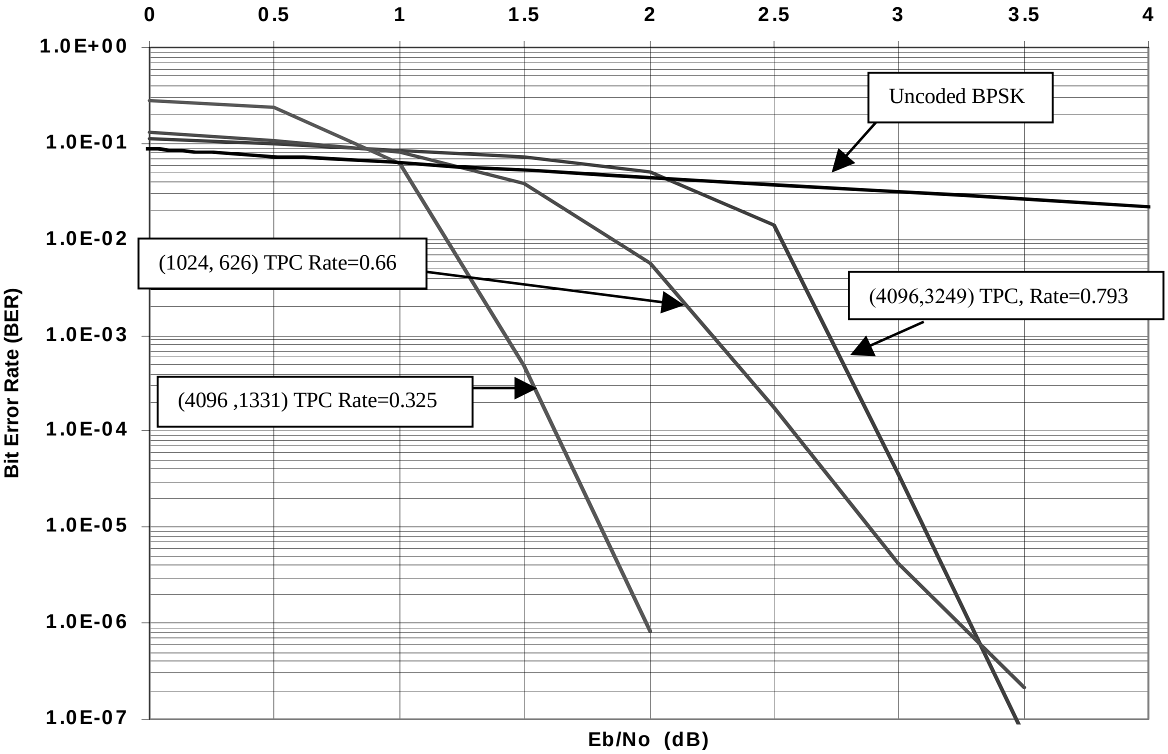

Figure 3 : Performance of Turbo Product Codes

Performance Bounds on Error Correcting Codes

It is well known that forward error correction (FEC) is a valuable technique to increase power and spectrum efficiency and thus has an important role in many systems. However, the development of FEC with increased coding gain and decreased redundancy (i.e., high rate) does have a limit. This limit arises from Shannon's channel capacity theorem that states that the code rate must be less than the capacity of the channel. No code (or concatenation of codes) can perform better than this theoretical maximum.

The capacity of the AWGN channel, if restricted to binary antipodal signaling, i.e. the normalized output signal is either $+\sqrt{E_s}$ or $-\sqrt{E_s}$, is then (for equally likely inputs) [6]

$$C = 1 - \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-\frac{1}{2}(y-\beta)^2} \log_2(1+e^{-2y\beta})dy \qquad (1)$$

where $\beta = \sqrt{\frac{2\mathrm{E}_s}{\mathrm{N}_0}}$, and $E_s = E_b R$. E_s is the energy per transmitted signal, R is the code rate, and E_b is the energy per bit. This integral cannot be evaluated analytically, so finding a solution requires numerical integration. As an additional complication, standard numerical integration routines are not usable because as y becomes increasingly more negative, the exponential term approaches zero and the logarithm term approaches infinity. To resolve this issue, it is necessary to truncate the interval to avoid the regions where the integrand is poorly behaved. As will be shown, it is not necessary to evaluate the integral over infinite limits. Instead it will be shown that the integral can be taken from –5 to 20.

If we divide the integral over three regions we obtain

$$C = 1 - L_1 - D - L_2 \tag{2}$$

where

$$L_1 = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{y_0} e^{-\frac{1}{2}(y-\beta)^2} \log_2(1+e^{-2y\beta})dy, y_0 < 0 \tag{3}$$

$$D = \frac{1}{\sqrt{2\pi}} \int_{y_0}^{z_0} e^{-\frac{1}{2}(y-\beta)^2} \log_2(1+e^{-2y\beta})dy, y_0 < 0, z_0 > 0 \tag{4}$$

$$L_2 = \frac{1}{\sqrt{2\pi}} \int_{z_0}^{\infty} e^{-\frac{1}{2}(y-\beta)^2} \log_2(1+e^{-2y\beta})dy, z_0 > 0 \tag{5}$$

In what follows we show that if y_0 and z_0 are selected properly, L_1 and L_2 can be ignored. First observe that L_1, D, and L_2 are each positive since their integrands are positive. In order to show that $C \approx 1 - D$, it is necessary to prove that $L_1 << D$ and $L_2 << D$.

Let's first look at the case where y << 0, then we notice that $e^{-2y\beta}$ >>1, in which case

$$1 + e^{-2y\beta} \approx e^{-2y\beta}$$

and $\log_2(1+e^{-2y\beta}) \approx \log_2(e^{-2y\beta}) = \frac{\ln(e^{-2y\beta})}{\ln(2)} = \frac{-2y\beta}{\ln(2)}$.

We select y_0 to ensure that y < y_0 is enough for

$$1 + e^{-2y\beta} \approx e^{-2y\beta}$$

for all y < y_0. Then the integral over the region in which $e^{-2y\beta}$ >>1 (i.e. y < y_0) is approximately

$$L_1 = \frac{1}{\ln(2)\sqrt{2\pi}} \int_{-\infty}^{y_0} (-2y\beta\, e^{-\frac{1}{2}(y-\beta)^2})dy\,.$$

This integral can be evaluated in terms of the cumulative Gaussian distribution function, which is defined by

$$\Phi(x) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}}\, e^{-\frac{y^2}{2}} dy\,, \text{ or } Q(x) = \int_{x}^{\infty} \frac{1}{\sqrt{2\pi}}\, e^{-\frac{y^2}{2}} dy\,.$$

First we observe that

$$L_1 = \frac{2\beta}{\ln(2)\sqrt{2\pi}} \left[-\int_{-\infty}^{y_0} (y-\beta)\, e^{-\frac{1}{2}(y-\beta)^2} dy - \int_{-\infty}^{y_0} \beta\, e^{-\frac{1}{2}(y-\beta)^2} dy \right]$$

$$= \frac{2\beta}{\ln(2)\sqrt{2\pi}} \left[e^{-\frac{1}{2}(y_0-\beta)^2} - \int_{-\infty}^{y_0} \beta\, e^{-\frac{1}{2}(y-\beta)^2} dy \right]$$

$$= \frac{2\beta}{\ln(2)} \left[\frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(y_0-\beta)^2} - \beta\Phi\left(y_0-\beta\right) \right].$$

Since $Q(x) = \Phi(-x)$ for all x, then

$$L_1 = \frac{2\beta}{\ln(2)} \left[\frac{1}{\sqrt{2\pi}} e^{-\frac{1}{2}(\beta-y_0)^2} - \beta Q\left(\beta-y_0\right) \right].$$

Next observe that $\Phi(x) \geq 0$, so $Q(x) \geq 0$, and therefore

$$L_1 \leq \frac{2\beta}{\ln(2)\sqrt{2\pi}} \left[e^{-\frac{1}{2}(\beta-y_0)^2} \right].$$

For efficient communications we are only interested in values of E_s/N_o less than 17dB. In fact, our primary interest is in values of E_s/N_o less than 5dB, but the 17dB limit is adequate for our analogies. If E_s/N_o is less than 17dB, then $\beta \leq 10$ and $\frac{2\beta}{\ln(2)\sqrt{2\pi}} \leq 12$ thus $L_1 < 12e^{-\frac{1}{2}(\beta-y_0)^2}$. In addition, we are only interested in rates less than 0.99, so $C \leq 0.99$. Since $L_1 > 0$ and $L_2 > 0$, it follows from (2) that $D \geq 0.01$. To make L_1 negligible, we need to guarantee that $L_1 << 0.01$. Suppose we want $L_1 < 5\times10^{-5}$, then we need $12e^{-\frac{1}{2}(\beta-y_0)^2} < 5\times10^{-5}$ which requires

$$-(\beta - y_0)^2 < 2\ln\left(\frac{5\times10^{-5}}{12}\right)$$

which implies $\beta - y_0 > \sqrt{24.78} \approx 4.98$. This is guaranteed if $-y_0 = 5$ since $\beta \geq 0$. In other words, we can ignore L_1 if we let $y_0 = -5$ in (3) and (4).

For the bounding of L_2 we see from (5) that the integrand is

$$e^{-\frac{1}{2}(y-\beta)^2} \log_2(1+e^{-2y\beta})dy$$

and $y > z_0 > 0$. Notice that we can find an upper bound for the logarithm. Since $e^{-2y\beta} \leq 1$, then $1+e^{-2y\beta} \leq 2$. This results in $\log_2(1+e^{-2y\beta}) \leq \log_2(2) = 1$, which gives

$$L_2 \leq \frac{1}{\sqrt{2\pi}} \int_{z_0}^{\infty} e^{-\frac{1}{2}(y-\beta)^2} dy .$$

From our earlier calculations, we recognize this as $L_2 \leq \frac{1}{\sqrt{2\pi}} \int_{z_0-\beta}^{\infty} e^{-\frac{u^2}{2}} du,$ where u = $y - \beta$.

This gives $L_2 \leq Q(z_0 - \beta)$. Suppose we want $L_2 \leq 5\times10^{-5}$, then we need $Q(z_0 - \beta) \leq 5\times10^{-5}$. Thus, it is sufficient if $z_0 - \beta \leq 5$. If we take $\beta \leq 10$ (because we are only interested in values of E_s/N_0 less than 17dB) we have $z_0 \leq 15$. Just to be conservative, we choose $z_0=20$ as the lower limit for the integral L_2.

Thus the integral can be approximated as

$$C \approx 1 - D = 1 - \frac{1}{\ln(2)\sqrt{2\pi}} \int_{-5}^{20} (e^{-\frac{1}{2}(y-\beta)^2} \log_2(1+e^{-2y\beta}))dy \tag{6}$$

where $\beta = \sqrt{\frac{2\mathrm{E}_s}{\mathrm{N}_0}}$, and $E_s = E_b R$. The advantage is that the integral in (6) is easily evaluated numerically using standard software such as Matlab or Maple.

The graph for the capacity is obtained as follows. For each value of β, R is determined from (6), since we are communicating at a rate equal to capacity (i.e., R=C). For each β and its corresponding value of R, we obtain E_b/N_o. This is one point on the graph. Then β is incremented and the same method is used to obtain another point on the graph. The graph of the E_b/N_o required to achieve capacity is shown in Figure 4.

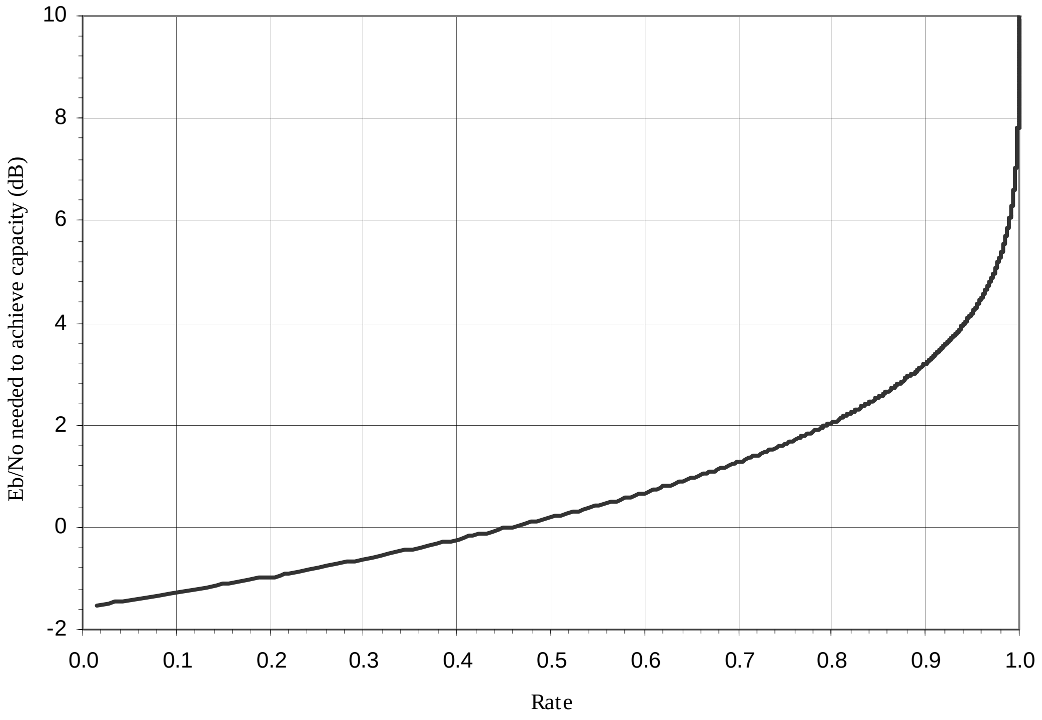

Figure 4 : Capacity of the Binary-Input Channel

Figure 5 shows the capacity curve in comparison with the simulation results from three turbo product codes. As mentioned before Shannon's theorem states that none of these codes can perform better than capacity. The 0.793 code is approximately 1.3 dB from capacity, the 0.66 rate code is about 2.3 dB from capacity, and the 0.325 code performs 2.1 dB from capacity.

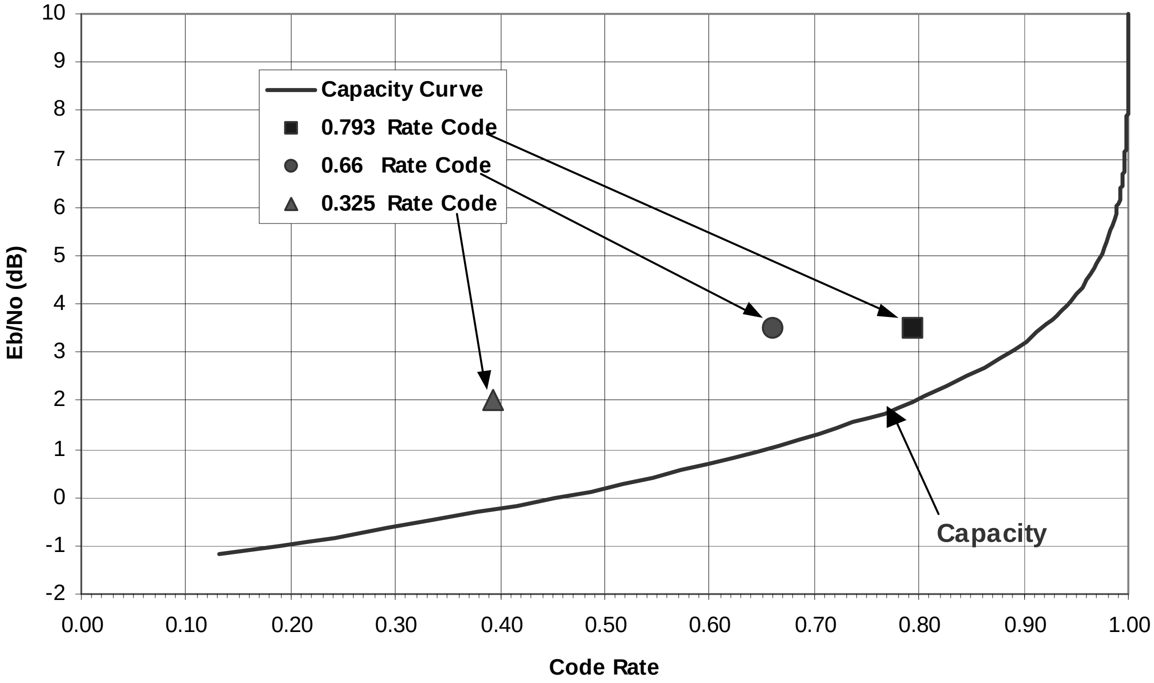

Figure 5 : Turbo Product Code Results Compared to Capacity

Conclusion

Turbo product codes are an effective way to increase power efficiency in satellite or wireless communication systems. They offer a wide range of flexibility in terms of performance, complexity, and code rates. In addition they perform fairly close to capacity. When compared to the standard rate 0.5 convolutional code which requires a signal-to-noise ratio of 3.5dB in order to achieve a bit error probability of 10^{-5}, a rate 0.793 turbo product code was found to provide a saving of 0.4 dB. This is a 9.6% decrease in energy power and a 60% increase in data rate. We can conclude that the turbo product code requires less energy and gives a higher data rate for the same bandwidth requirement.

References

[1] Technical Description of Turbo Product Codes, www.aha.com

[2] Efficient Channel Coding, www.eccincorp.com

[3] J. Hagenauer, E.Offer, L.Papke, “Iterative Decoding of Binary Block and Convolutional Codes”, *IEEE Transactions on Information Theory*, Vol. 42, No. 2, March 1996.

[4] W.W. Peterson, E.J.Weldon, *Error Correcting Codes*, The MIT Press, Cambridge Mass., 1972.

[5] B. Sklar, *Digital Communications*, Prentice Hall, Englewood Cliffs, New Jersey, 1988.

[6] W. E. Stark, unpublished course notes, University of Michigan, 2000