Convolution Code Encoder Design using Particle Swarm Optimization for Constraint Length 6

Kanika Budhwar
Research Scholar
Banasthali Vidyapith, Jaipur , India

Dr. Anuj Sharma
Dean Academic
Om Institute of Technology & Management, Hissar, India

Abstract- Due to channel distortions, transmitted data can be altered. Coding techniques create code words by adding redundant bits to the input data. Convolution code is the most reliable method for transmitting or retrieving the error free data. Convolution code encoder consists of shift registers and mod-2 adders. The performance of convolutional code depends upon the connections between shift registers and mod-2 adders. We are using particle swarm optimization (PSO) technique to find out the best connection as a particular connection combination affects the error correcting capability of convolution code.

Keywords – Convolution code, PSO, Shift Register, BER (bit error rate).

I. INTRODUCTION

For communicating the digital data efficiently, channel coding techniques are used. Convolution code is used as error correcting code in a wide variety of communication and recording systems, including digital video broadcasting, digital audio broadcasting, cellular mobile, and satellite communication. Convolution code is frequently used to correct error in noisy channel. It has good correcting capability and performs well on bad channels (with error probability of about 10-3). It accepts a fixed number of message symbols and produces a fixed number of code symbols. Its computation depends not only on the current set of input symbols but also on some of previous input symbols. Convolution code has many encoder structures (outputs connection with shift registers). We suggest that the PSO algorithm finds the best connections for convolution code encoder.
PSO algorithm has some good features such as good diversity, wide searching area and strong global optimize capability.

II. CONVOLUTION CODE

Convolution code is one technique within the general class of channel codes. Channel codes (also called error-correction codes) permit reliable communication of an information sequence over a channel that adds noise, introduces bit errors, or otherwise distorts the transmitted signal. Elias introduced convolutional codes in 1955. These codes have found many applications, including deep-space communications and voiceband modems. Convolutional codes continue to play a role in low-latency applications such as speech transmission and as constituent codes in Turbo codes.
A convolutional code is a type of code in which each k-bit information to be encoded is transformed into an n-bit symbol. A convolutional code introduces redundant bits into the data stream through the use of linear shift registers as shown in (Figure1). The inputs to the shift registers are information bits and the output encoded bits are obtained by modulo-2 addition of the input information bits and the contents of the shift registers.

2.1 CONVOLUTION CODE ENCODER

Convolutional encoder is used to obtain convolution codes. It is made up of a fixed number of shift registers.

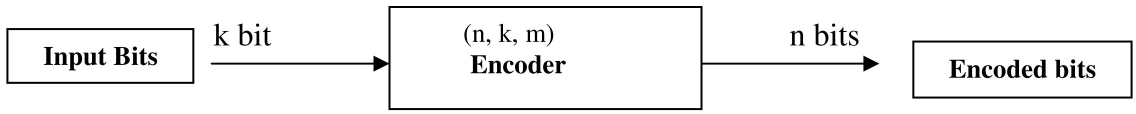


Figure 1: Convolution Code

Each input bit enters a shift register and the output is derived by combining the bits of the shift register. The number of output bits depends on the number of modulo 2-adders used with shift registers**.**

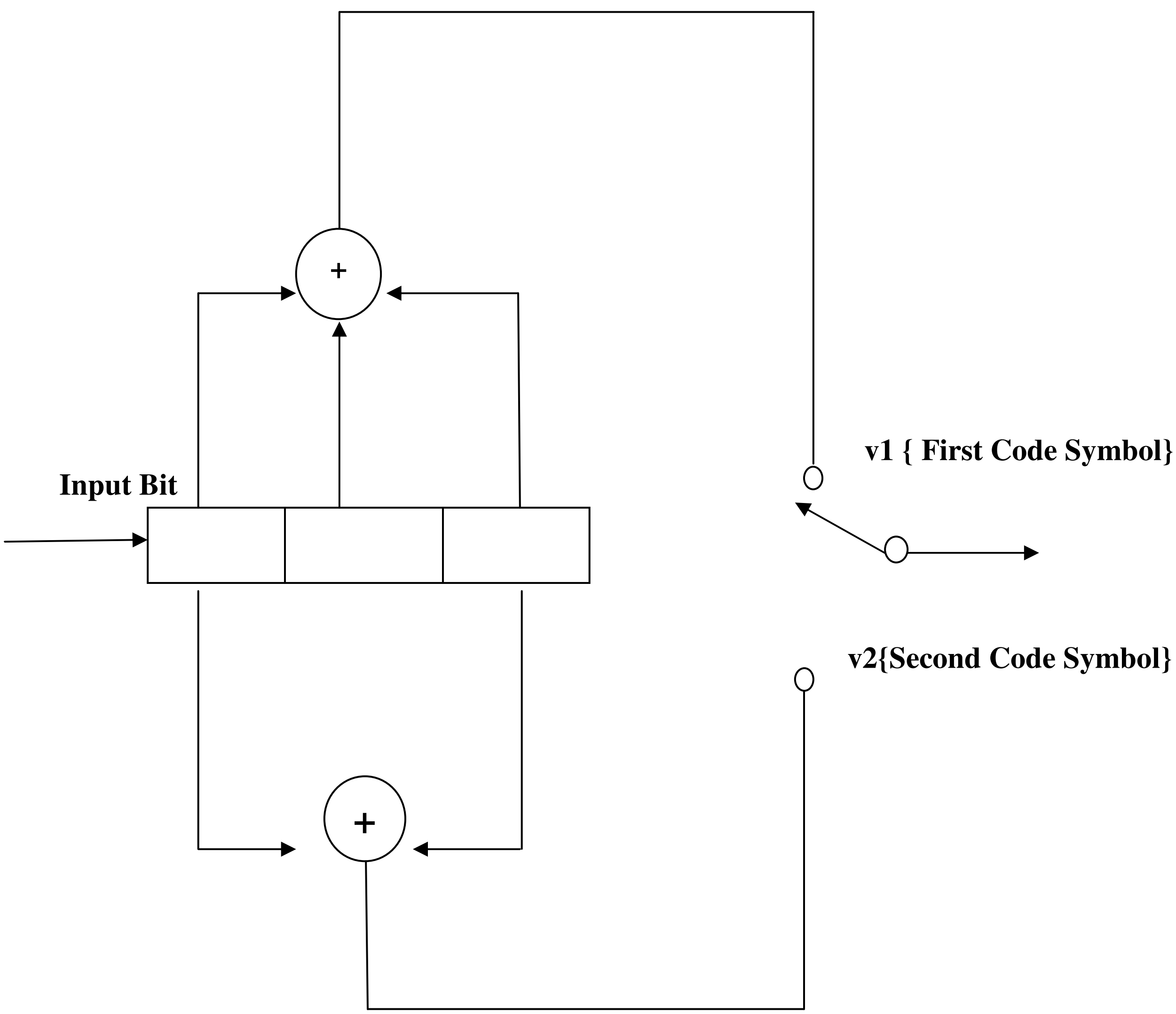


Figure 2: Convolution Code Encoder

2.1.1 Encoder Parameters
Convolutional codes are primarily described by three parameters (n ,k ,m) where,

n = number of output bits

k = number of input bits
m = number of memory registers

A convolutional encoder is characterized by two parameters, namely code rate (r) and constraint length (K).

The code rate is defined as the ratio of the number of message bits (k) to the number of encoded bits (n).

$$\mathbf{r = k/n}$$

The constraint length (K) is defined as the number of shifts over which a single message bit can affect the encoder output.

III. PARTICLE SWARM OPTIMIZATION

Particle swarm optimization technique provides an evolutionary based search. It was developed in 1995 by James Kennedy and Russell Eberhart. PSO algorithm is especially useful for parameter optimization in continuous, multi-dimensional search space. It is developed from swarm intelligence and is based on the research of bird and fish flock movement behaviour. The connection to search and optimization problem is made by assigning direction vector and velocity to each point in multi-dimensional search space. It uses a number of agents (particles) that constitute a swarm moving with a particle speed toward the best particle found so far by particular heuristic including their experience from past generation.

3.1 PSO Algorithm

The PSO algorithm consists of following steps, which are repeated until some stopping condition is met:
1. Initialize the population, location and velocity.
2. Evaluate the fitness of the individual particle (Pbest).
3. Keep track of the individual highest fitness (Gbest).
4. Modify velocity based on Pbest and Gbest location.
5. Update the particle position.
6. Terminate if condition is met.
7. Go to step 3.

After finding the two best values, the particle updates its velocity and position with following equation:

$$v(t+1) = w.v(t) + c1.rp.(plb\text{-}x(t)) + c2.rg.(pgb\text{-}x(t)) \quad (1)$$
$$x(t+1) = x(t) + v(t+1) \quad (2)$$

The $v(t)$ & $x(t)$ is the velocity and position of the particle at time t. The value w, $c1$ and $c2$ ($0 \le w \le 1.2$, $0 \le c1 \le 2$ and $0 \le c2 \le 2$) are user supplied co-efficient. The values of rp and rg ($0 \le rp \le 1$ and, $\le rgd \le 1$) are random value regenerated for each velocity update.

3.1.1 Advantages of the basic Particle Swarm Optimization Algorithm.

(1) PSO is based on the intelligence. It can be applied into both scientific research and engineering use.

(2) PSO have no overlapping and mutation calculation. The search can be carried out by the speed of the particle. During the development of several generations, only the most optimist particle can transmit information onto the other particles, and the speed of the researching is very fast.

(3) The calculation in PSO is very simple. Compared with the other developing calculations, it occupies the bigger optimization ability and it can be completed easily.

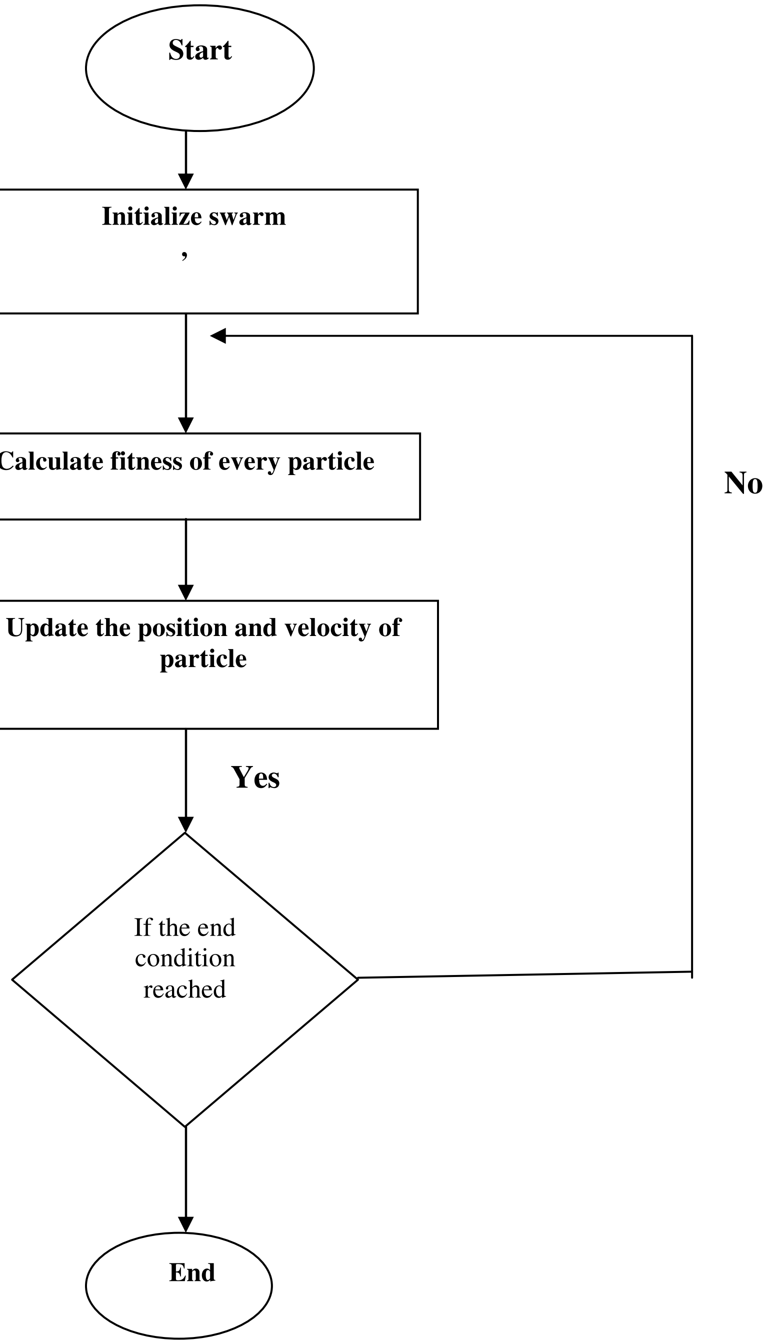


Figure 3: PSO Algorithm

(4) PSO adopts the real number code, and it is decided directly by the solution. The number of the dimension is equal to the constant of the solution.

3.1.2 Disadvantages of the basic Particle Swarm Optimization Algorithm:

(1) The method easily suffers from the partial optimism, which causes the less exact at the regulation of its speed and the direction.

(2) The method cannot work out the problems of scattering and optimization.

(3) The method cannot work out the problems of non-coordinate system, such as the solution to the energy field and the moving rules of the particles in the energy field.

3.8 Convolutional Code Optimization using PSO

Optimization is the mechanism by which one finds the maximum or minimum value of a function or process. Optimization can refer to either minimization or maximization.

PSO is limited in optimization solution, especially in solving discrete optimization solutions. So there are some improved algorithms.

Step1: Generate polynomial

A Polynomial description of convolution encoder describes the connection among shift registers and modulo -2 adders. Build a binary number representation by placing a 1 in each connection line from shift feed into the adder and 0 elsewhere. Convert this binary representation into an octal representation.

Step2: Draw the trellis

A trellis description of a convolutional encoder shows how each possible input of encoder influences both the output and state transition of encoder. Start with a polynomial description of the encoder and use poly2trellis function to convert it to valid structure.

Step3: Calculate BER

Calculate bit error rate using octal code and trellis structure. To decode convolutional code use the vitdec function with the flag hard and with binary input data. Because the output of convenc is binary, hard decision decoding can use the output of convenc directly. After convec adds white Gaussian noise to the code with AWGN.

Step4：Update particle's position and velocity

At each time, all particles have an update. At iteration t, the element in the vector is updated. Particle's position is decided by velocity as equation(2). At the decoding process, the update of vi (t +1) and xi (t) update must act up to transfer rule of encoder state. Select lowest value of bit error rate as fitness function.

Step 5 ：Update personal best position and the global best position.

Update personal best position and the global best position after all particles position have been updated.

Step 6：Ending condition

When iteration t=L, all particle's position have been updated for L times and reached
the grids ending.

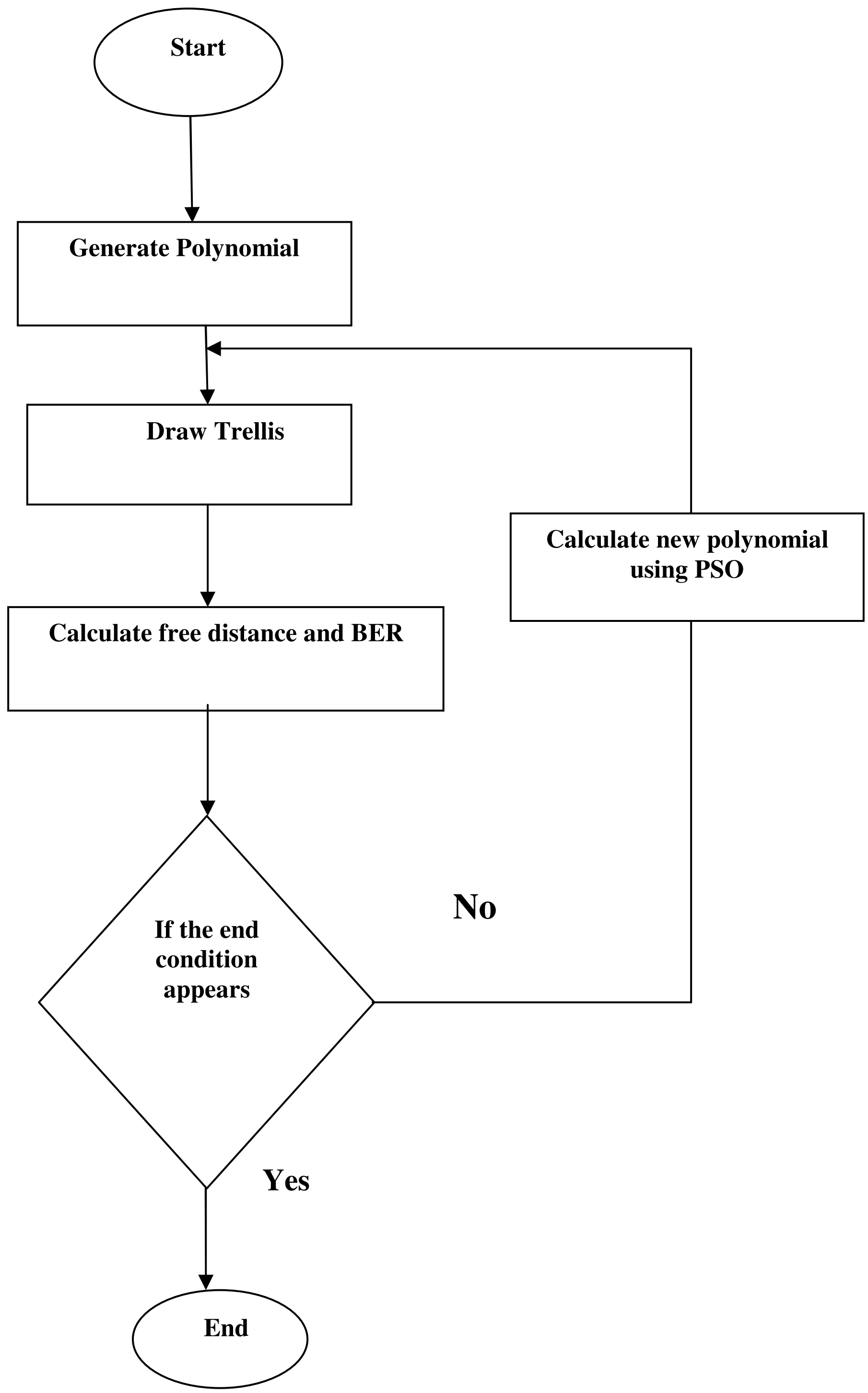
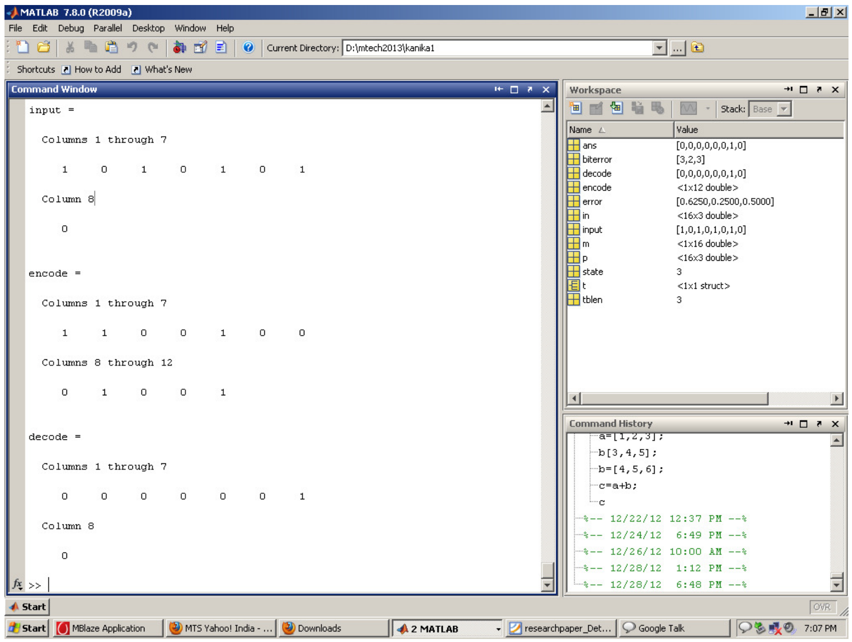
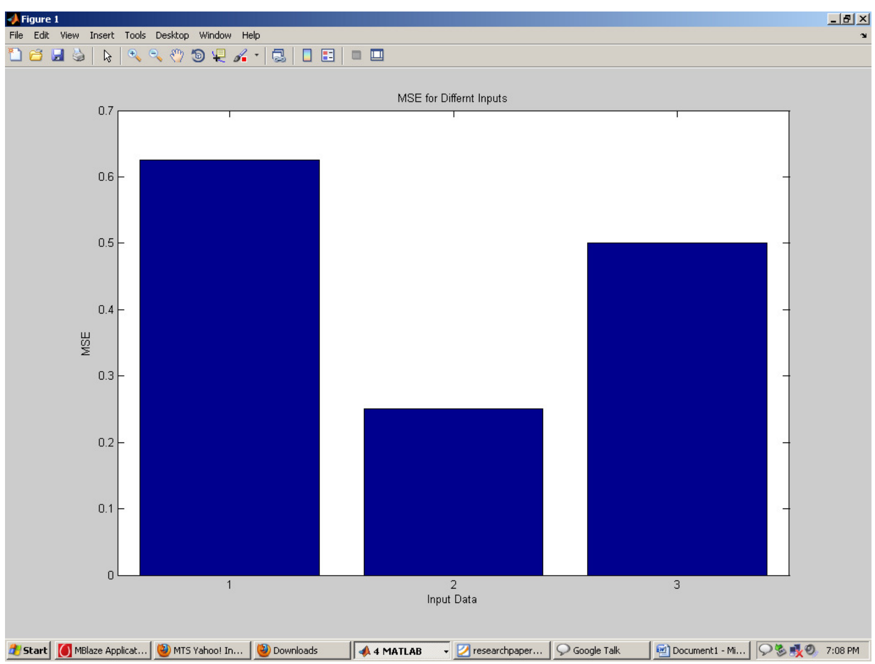


Figure 4: Convolutional Encoder using PSO

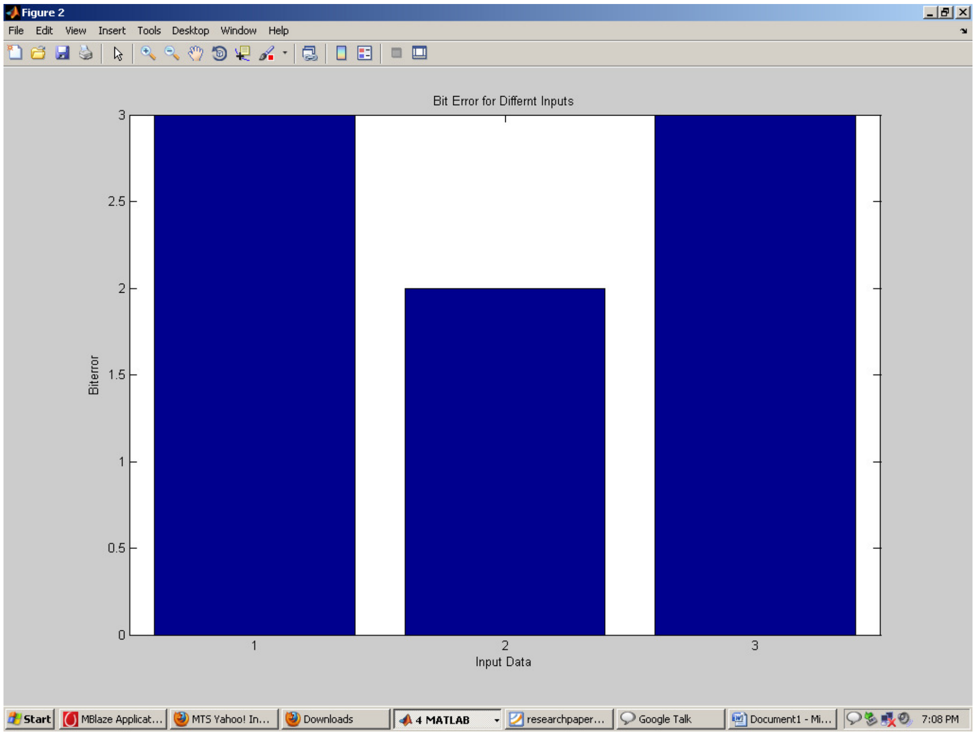
4. RESULT



We have used 8 bit as input and encoded it into 12 bits.



The bar graph is showing the error rate generated during the encoding process. Here the error is given in the form MSE (Mean Square error). Three Bars represents error in three different inputs. The bar graph is driven for 8 bit input.



The bar graph is showing the Bit Error Rate (BER) generated during the encoding process. Three bars represent errors in three different inputs. The bar graph is driven for 8 bit input.

REFERENCES

[1] Schwartz M. (2005), Mobile Wireless Communications, Cambridge University Press, U.K
[2] Hamming, R. "Error detecting and correcting codes", Bell Sys.Tech .J., vol. 29 ,pp.147-160 , 1960.
[3] R. Johannesson and K. Sh. Zigangirov, *Fundamentals of Convolutional Coding*.
[4] E. R. Berlekamp, ed., Key Papers in the Development of Coding Theory.
[5] Richard D. Wesel, "Convolution Code University of California at Los Angeles, California 2002.
[6] Xiaoling Huang, Yujia Zhang, Jinxue Xu and Yongfu Wang, "Fast Decoding of Convolution Codes Based on Particle Swarm Optimization".
[7] Angeline P J. (1999). Using selection to improve Particle Swarm Optimization. Proceedings of the 1999 Congress on Evolutionary Computation. Piscataway. NJ:IEEE Press,1999:84-89.
[8] Sanjay Sharma , " Digital Communications", Pearson Education, Sixth edition.
[9] Ranjan Boss, "Information Theory and Coding", Tata McGraw-Hill, Second Reprint 2008.