**Performance Analysis of BSC.**

**Method 1.**

1.Fix a code(N,K) vary p and get BLER

2.This method resembles how BLER simulation is done for AWGN channels.

3.The rateless paper due to kai chen, does a simulation of similar type

They fix a code 2048,1024 this we call ‘sent rate’=1/2

They use two transmissions and thus get a code achieved rate ¼,

The performance of this meets 2048,512 performance with varying SNR.which proves the point as it achieves the performance of normal polar code under SC decoding.

**Q: this analysis might not require the code to detect the channel ,or to adapt.?**

**The performance of 2048,1024 1 tx is not shown.**

**What to expect if this was shown?**

Guess , the curve would be flat at the top and would start falling after SNR is such that C>½.and would follow 2048,1024 BLER curve for AWGN.

On other hand after 3Tx it would resemble 2048,334 curve

**What happens if we let it run for as many transmissions needed for a given reliability?**

Guess, given the code can detect, the code will adapt , it will be a flat curve which might be touching the performance of the curves mentioned above at the given reliability.ie for a bler of 10^-2 and SNR1 the code will adapt to that achieved rate (or that number of txs) which gives this reliability.

In our simulation we let the code run for as many txs as required.

**Method 2**

1.fix a channel , use this channel to design coding and decoding and vary the coding rate.

2.Arikan, an MS thesis under teletar and hassani etc, have shown plots like this

3.all our simulations have plots like this

How it captures the performance?

The farther we are from capacity the better the code will perform.

How to reach from method 1 to method 2 and vice versa?

In method 1 plot multiple curve for different rates.now fix a SNR or p draw a vertical line, take the intersection points and plot them vs rate.

**Q:The likelihoods are dependent on the channel so while in method 1 each vertical line uses same likelihoods, in method 2 each curve uses same likelihoods?**

The curves from method 2 make “achieving capacity” more prominent are more information theoritic, while the curves in method 1 make “performance of a code” more prominent ,are more useful in communication theory.

Our method of analysis for rateless code.

1.we fix a channel (i,e, p)

2.we send TRY to send a code say 2048,1024, (N,K)

3.but it doesnt go in one tx, or 2 tx, it stops only when it reaches the rate which assures a given reliability , hence achieves this rate.

4.now I’m varying K , i,e trying to send 2048,334 ;2048,512; 2048,1024 say. Over a 2048,512 channel.2048,334 will pass {with (1- false alarm) probab}, 2048,512 will pass 2048,1024 will achieve 2048,512.

**Q:Should i expect sent rate = achieved rate as long as sent rate< C and then a flat line?**

5.presently all the achieved rates for a given sent rate and corresponding BLERs are getting averaged . but BLER depends on achieved rate ONLY, and the achieved rate varies considerably for a given sent rate and hence the estimate of BLER might be wrong.This averaging is not correct for the getting the BLER.Instead **bucketing the BLERs depending on achieved rate might smoothen the curve.**

With a tweak in plotting the overlapping lines are gone, but curves are not smooth.

**Q:Implementation of RX “knows channel”**

Details:

1.use compound channel [0.04,0.15,0.2,0.25], and there capacities

2.try to send data at 0.1 to 1 fraction of the the capacity of 0.04

3.say d fraction is used.if the sent rate >= d fraction of the capacity of any channel , the channel guess is changed to that of a lower channel, and retransmission is done.

Q:IS THIS CORRECT TO DO?

4.the achieved rate vs the sent rate is a straight line.

5.BER curves are smooth

6.BUT they are above the BER curves for LTPT case. Is that OK?