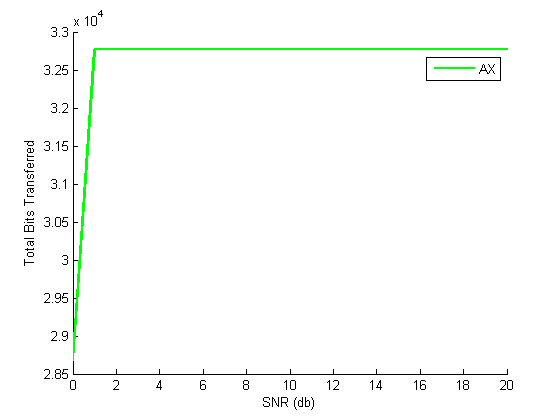
**Communication Theory Project #1: Team Mosquito**

Xiaowen Zhang, Xiangling Kong, Jeffrey Shih

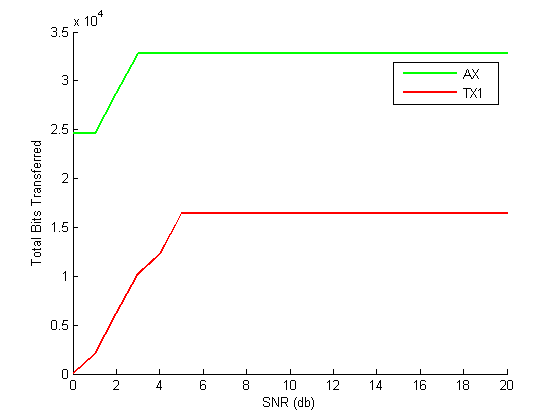
The general strategy consists of two parts, transmission and jamming. For transmission, we decided to use a low power signal across all channels and perform a max-vote on the results. The modulation scheme we are using is PSK with an M value of either 2 or 4 depending on the channel conditions. The selection process for M is similar to branch prediction. We start with an M value of 4. If the results are incorrect, we move to an M value of 2. If we are successful in transmitting with this M value twice, we move back up to 4. This system allows for high bit rates at both low and high SNR. When the SNR is low, we would be mostly stuck at the M = 2 level. However at high SNR, even if we get one packet incorrect, we are still able to recover back to M = 4 very quickly. In addition, we flip the bits on every other channel so that that if the error is lower on the inverted bits, there is a higher chance to pick those bits in the max-vote. In terms of jamming, we detect the opposing signal by checking the standard deviation in each channel on the receiving end. Since we transmit the same signal in every channel (except the channel we are blocking), the channel with the most power is most likely the channel the opponent is transmitting on. Thus, on the next cycle, we create a random 4-QAM signal and put it through that particular channel instead of putting our signal through it.

We considered using error encoding but eventually decided against it. This is because the max-vote serves as a type of error encoding. Our system trades off higher power (and thus lower error rate) for more of the same signal. Even if the error rate is high, say for every bit, there is a 30% chance that we get it wrong, since we are broadcasting on 15 channels, that’s only 30% of the signals received were incorrect. This means that if we max-vote, as long as the error is below 50%, we should be able to correctly ascertain the sequence. This avoids the issue of even with low error rates, there is always the probability that one bit is wrong and mess the entire system up. Since over one thousand bits are being transmitted at once, the probability of one bit being wrong is pretty high and since we need to throw out the entire packet even if one bit is wrong, that error is not tolerable.

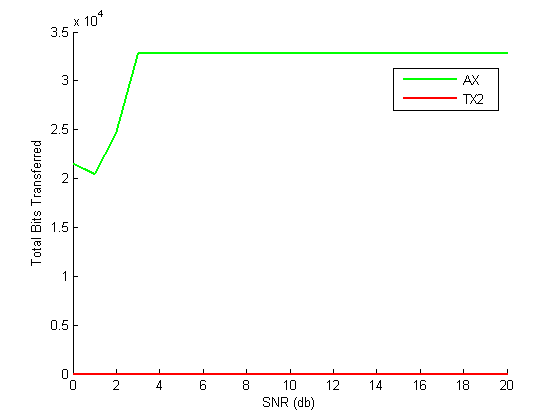
**Figure 1**: No Interference



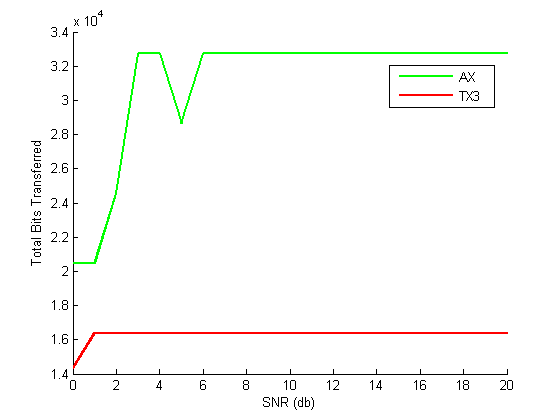
**Figure 2**: TX1



**Figure 3**: TX2



**Figure 4**: 4-QAM on One Channel



**Figure 5**: 4-PSK on One Channel

