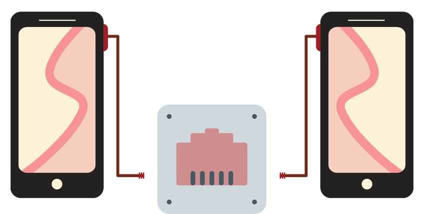
Low Complexity Algorithms for Transmission of Short Blocks over the BSC with Sparse Feedback

Sraavya Pradeep, Esha Thota
PI: Richard Wesel, DLS: Amaael Antonini

FAST TRACK TO SUCESS
CSL
UCLA Samueli School of Engineering
SUMMER UNDERGRADUATE RESEARCH PROGRAM

INTRODUCTION

Most communications channels are imperfect; noise will interfere and corrupt transmitted data. To combat this, communications systems relay information (regarding data sent to a receiver) back to the transmitter.



→ Encoder → Channel → Decoder →

The current method uses causal encoding, which simultaneously transmits and verifies bits.

OBJECTIVE

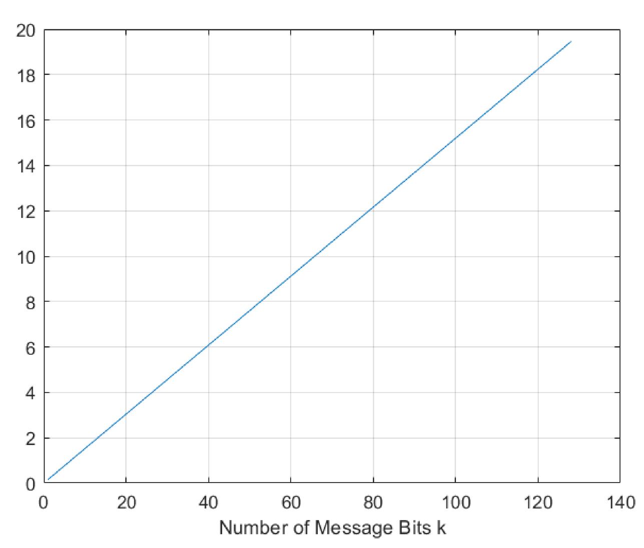
New research modifies the algorithm to utilize sparse feedback: sending feedback after a determined number of bits have gone to the receiver, instead of every bit, to increase efficiency without loss in performance.

MATERIALS

- Matlab
- CLion Software
- GMP Library
- CLion

PARTITIONING

After systematic transmission, we must determine how many partitions we should use for the number of bits being transmitted. We do this by: log2((1-p)^k), where p is the error probability of the channel, and k is the number of bits to be transmitted.



RESULTS

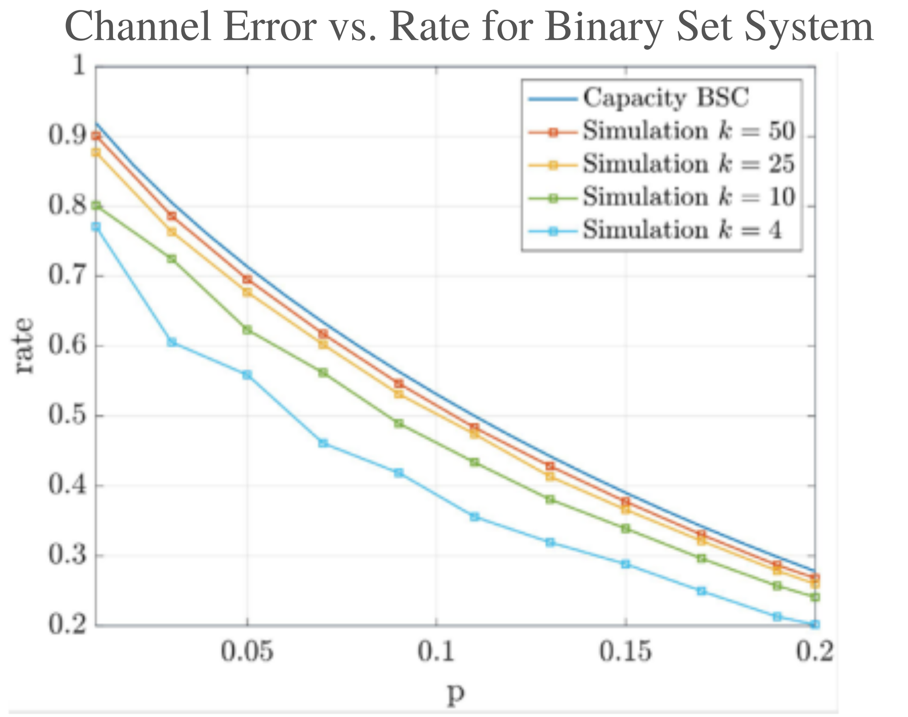


Figure 1. Regular Feedback System. ref. A. Antonini & R. Gimelshein

Regular Feedback System

The graph displays the error probability *p* of the channel against the performance rate (rate at which messages are transmitted & decoded). As the transmitted bits *k* increase, we approach performance closer to channel capacity.

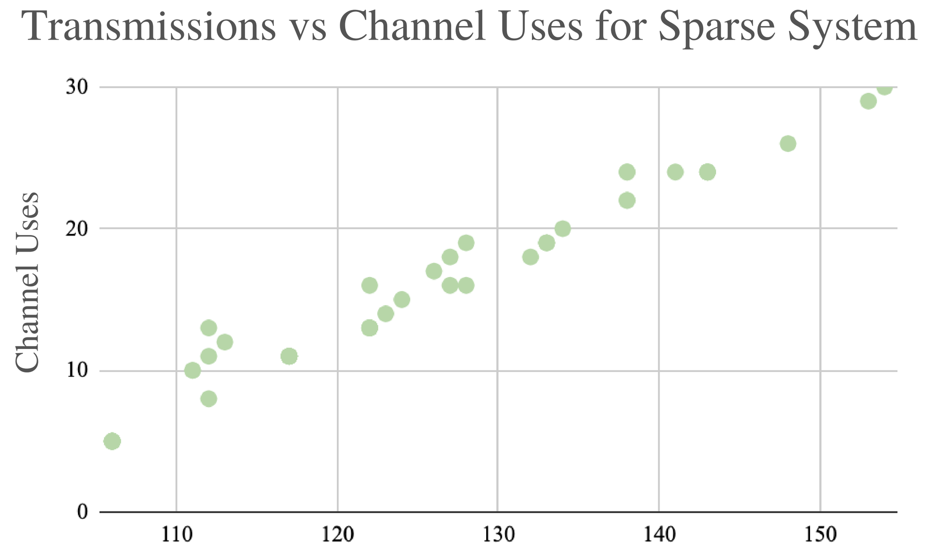
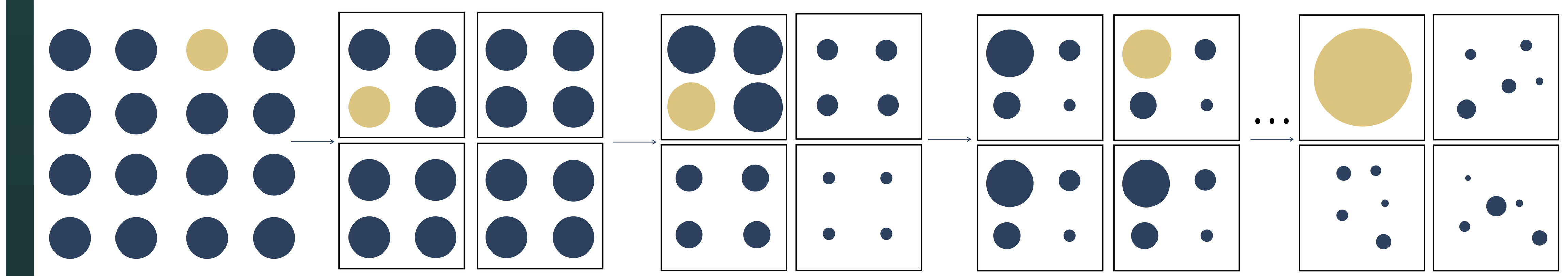


Figure 2. Sparse Feedback System

Sparse Feedback System

The graph displays the transmissions vs channel uses of the Sparse Feedback system. We see that as we increase the transmissions, the channel usage will increase. (Plot shown for block size 2, partition count 4)

SPARSE SYSTEM



16-bit system; all messages have equal probability of being true. Receiver's belief state of each is 1/16. The true message in yellow.

Partition all messages into four sets with equal probability (S0, S1, S2, S3). Transmitter sends data about theta being in a set.

After some communication between the transmitter & receiver, the probability of S0 has gone up, and other sets have gone down.

Rearrange messages within sets so probabilities of each set is close to equal. We then continue the process from steps 2 & 3.

After much repetition, the true message reaches a probability such that it is the only message contained in a set. We enter confirmation phase.

METHODS

Updating Probabilities.

Using **GMP library** for arbitrary precision + **Matlab** functionality. Modify the current algorithm to update & merge probabilities of multiple sets. First, recalculate multiple-way probabilities using **Bayes' Rule of Conditional Probability**, which is:

$$p(B\,|\,A) = \frac{p(A\,|\,B)\,p(B)}{p(A)}$$

Must use the error probability *p* and its complement *q*, aka (1-*p*), to calculate probability of the transmitted message.

Merging Probabilities.

We must store all messages with their probabilities in an ordered fashion, so we can later sort through them & partition appropriately. To do this, we place the message structs into an ordered linked list, ordering them by **decreasing probability**.

Regrouping The Sets.

Reorganize messages into determined number of sets with equal (or close to equal)probability. Use partition (see: sparse system) to organize groups by filling up each probability "bucket" (partition) to a value within tolerance of the **target probability**. Once message reaches a probability that exceeds a bucket's tolerance, we halve number of partitions.

CONCLUSION

Regular Feedback System

Binary Set system- splits all messages into one of two sets. Uses a version of ACK/NACK (Acknowledgement/Negative Acknowledgement) feedback to determine probability of theta.

- Pros: Simpler logic means less processing time/overhead
- Cons: Channel use is less efficient, bitwise transmission and decoding

Sparse Feedback System

Multiple Set system- finds the ideal number of partitions based on # of k bits. Splits probabilities into the "buckets". Uses more complicated logic to determine probability of theta from a message of multiple bits.

- Pros: Channel use decrease within certain range of tolerance, improved performance
- Cons: Limited range of improvement, overhead affected by extra processing

ACKNOWLEDGEMENTS

We would like to thank the National Science Foundation, UCLA Summer Undergraduate Research Program, and UCLA's Fast Track to Success for providing the resources for this publication. We would also like to thank Professor Richard Wesel, and Amaael Antonini for their guidance and support.

REFERENCES

Antonini, Rita G. and Richard Wesel. Causal (Progressive) Encoding over Binary Symmetric Channels with Noiseless Feedback, D1-S3-T1.2, ISIT 2021.