# Fast Channel Simulation via Error Correcting Codes

Sharang Sriramu, Rochelle Barsz, Elizabeth Polito, Aaron Wagner

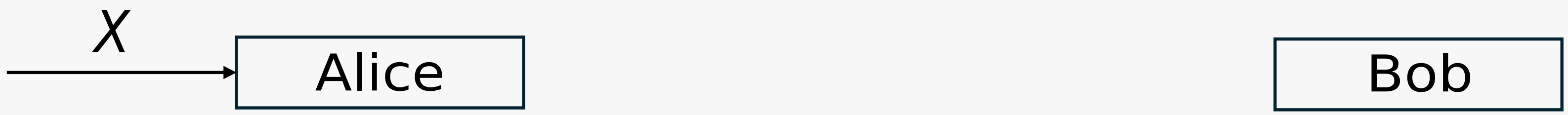
# Simulating $p_{Y|X}$

Alice

Bob

$p_{XY}$ known to Alice and Bob

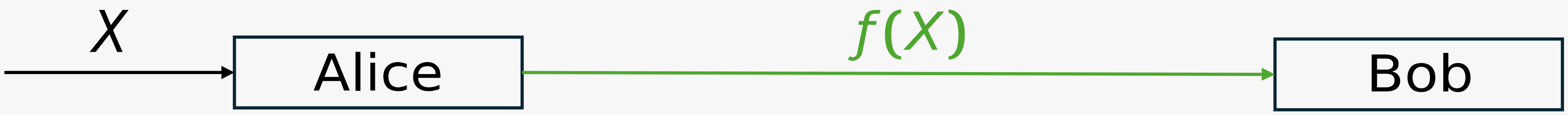
# Simulating $p_{Y|X}$




$p_{XY}$ known to Alice and Bob

$X \sim p_X$

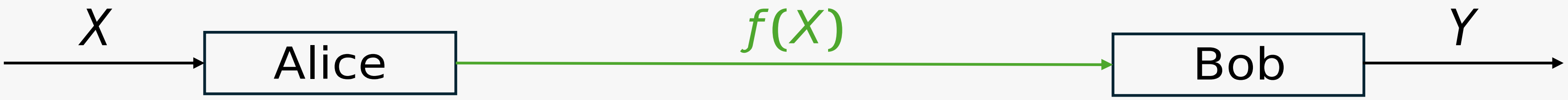
# Simulating $p_{Y|X}$




$p_{XY}$ known to Alice and Bob

$X \sim p_X$

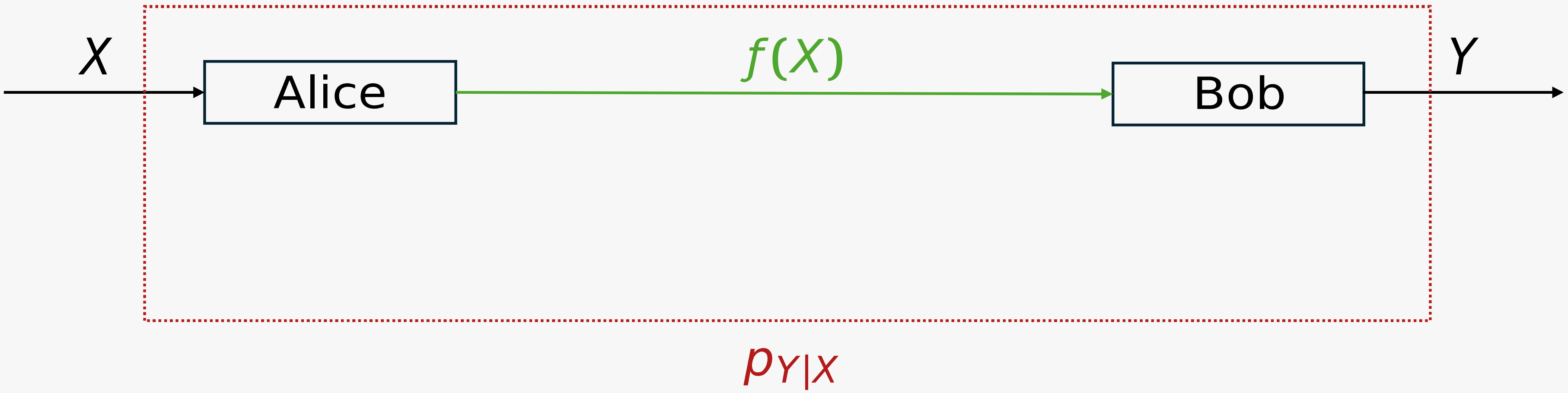
# Simulating $p_{Y|X}$




$p_{XY}$ known to Alice and Bob

$X \sim p_X$

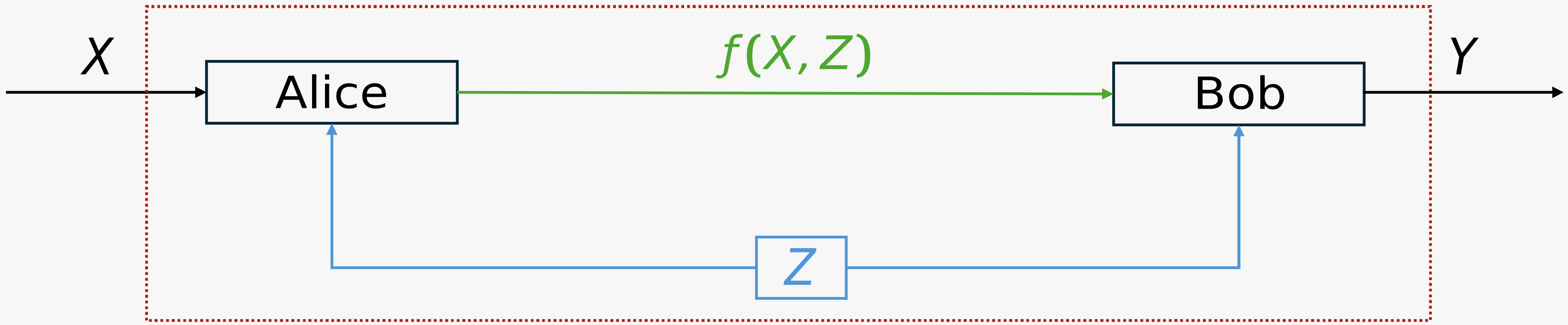
# Simulating $p_{Y|X}$




$p_{XY}$ known to Alice and Bob

$X \sim p_X$

# Simulating $p_{Y|X}$



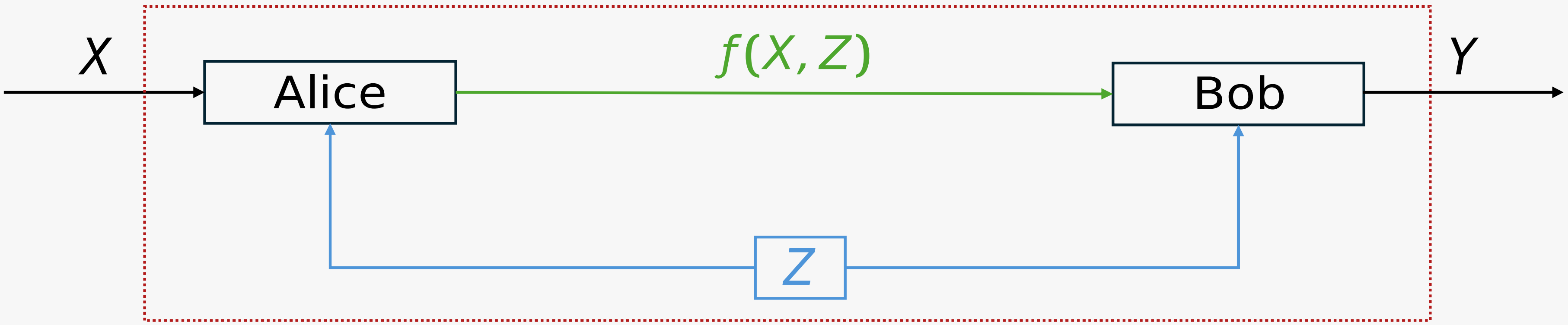

$p_{Y|X}$

$p_{XY}$ known to Alice and Bob

$X \sim p_X$

$Z \perp\!\!\!\perp X$: Unlimited Common Randomness

# Simulating $p_{Y|X}$




$p_{Y|X}$

$p_{XY}$ known to Alice and Bob

$X \sim p_X$

$Z \perp\!\!\!\perp X$: Unlimited Common Randomness

Objective: Minimize $R_1 = E[\text{len}(f(X, Z))]$

# Applications

1. Neural network-based compression
2. Model Compression [Havasi *et al*., 2019]
3. Differential privacy [Shah *et al*., 2022], [Liu *et al*., 2024]

Goal: Simulate $n$ i.i.d. uses of the target channel simultaneously

Existing simulation algorithms: $\exp(n)$ computational complexity

# Sampling-based Methods

Existing SOTA algorithms fall under this category: [Flamich, 2024], [Flamich *et al.*, 2024] etc.

Common Randomness: $Y_1^n, Y_2^n, Y_3^n, \ldots, Y_I^n, \ldots$ i.i.d. codebook $\sim p_{Y^n}$

$\uparrow$

Selection rule at encoder, depends on $X^n$ and $Y_1^n$

# Sampling-based Methods

Existing SOTA algorithms fall under this category: [Flamich, 2023], [Flamich *et al.*, 2024] etc.

Common Randomness: $Y_1^n, Y_2^n, Y_3^n, \ldots, Y_I^n, \ldots$ i.i.d. codebook $\sim p_{Y^n}$

$\uparrow$

Selection rule at encoder, depends on $X^n$ and $Y_2^n$

# Sampling-based Methods

Existing SOTA algorithms fall under this category: [Flamich, 2024], [Flamich *et al.*, 2024] etc.

Common Randomness: $Y_1^n, Y_2^n, Y_3^n, \ldots, Y_I^n, \ldots$ i.i.d. codebook $\sim p_{Y^n}$

$\uparrow$

Selection rule at encoder, depends on $X^n$ and $Y_3^n$

# Sampling-based Methods

Existing SOTA algorithms fall under this category: [Flamich, 2024], [Flamich *et al.*, 2024] etc.

Common Randomness: $Y_1^n, Y_2^n, Y_3^n, \ldots, Y_I^n, \ldots$ i.i.d. codebook $\sim p_{Y^n}$

$\uparrow$

Selection rule at encoder, depends on $X^n$ and $Y_I^n$

Transmit selected index $I$ to the decoder, using $\approx \log I$ bits

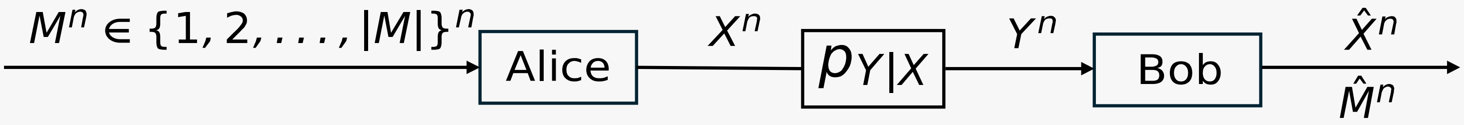
The rate, $\log I$, scales linearly in $n$

Computational complexity ($\propto I$) scales exponentially in $n$

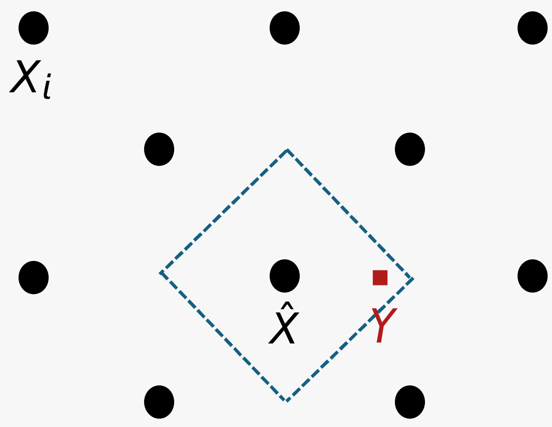
# Error-Correcting Codes for Simulation

# Error-Correcting Codes for Simulation

Channel Coding Setup




$M^n = \hat{M}^n$ with high probability




Good decoders are highly efficient vector quantizers

Channel simulation subsumes quantization

# Error-Correcting Codes for Simulation

Polar codes [Arikan, 2008]:

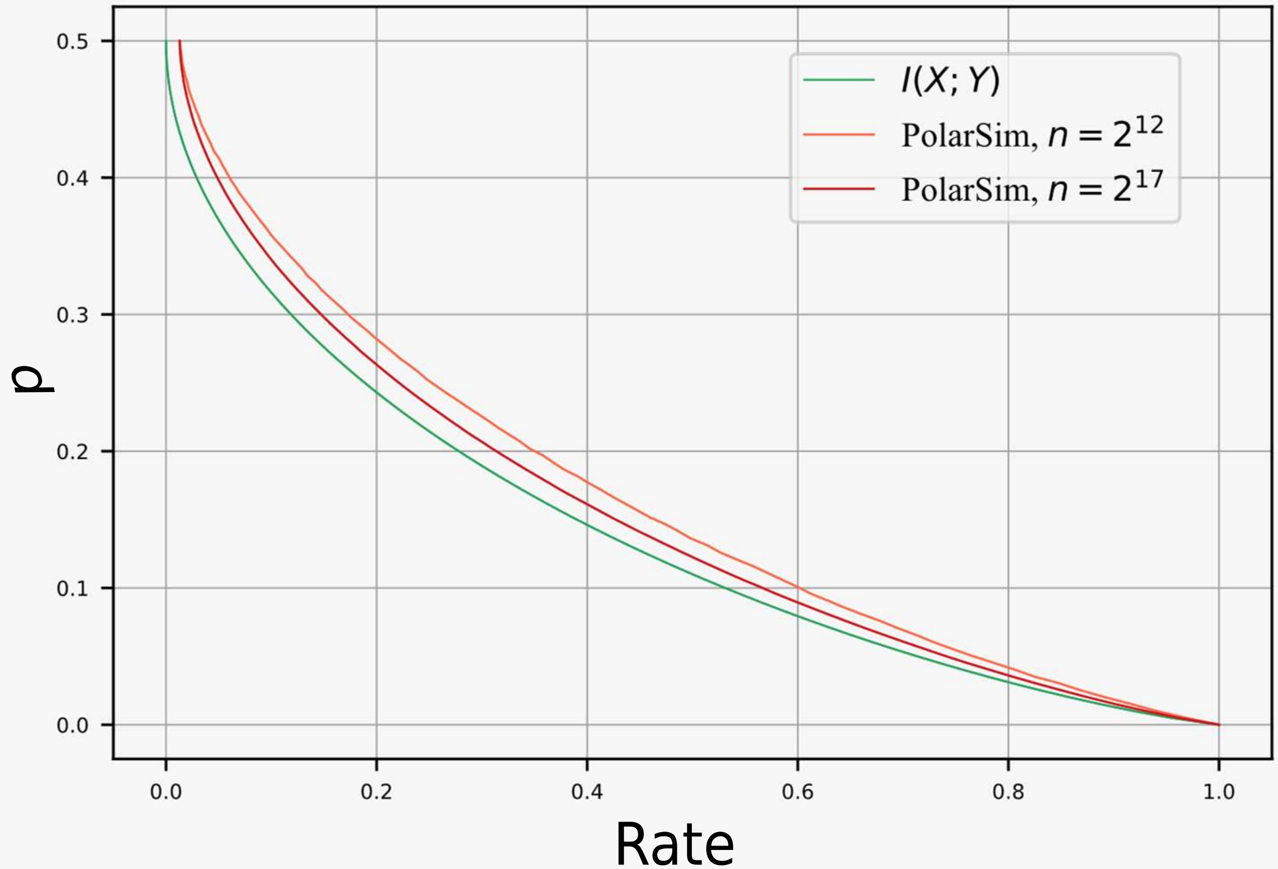
- Capacity achieving codes for symmetric binary input channels
- $O(n \log n)$ encoding and decoding complexity

PolarSim:

- Rate-efficient simulation algorithm for symmetric binary output channels
- $O(n \log n)$ encoding and decoding complexity

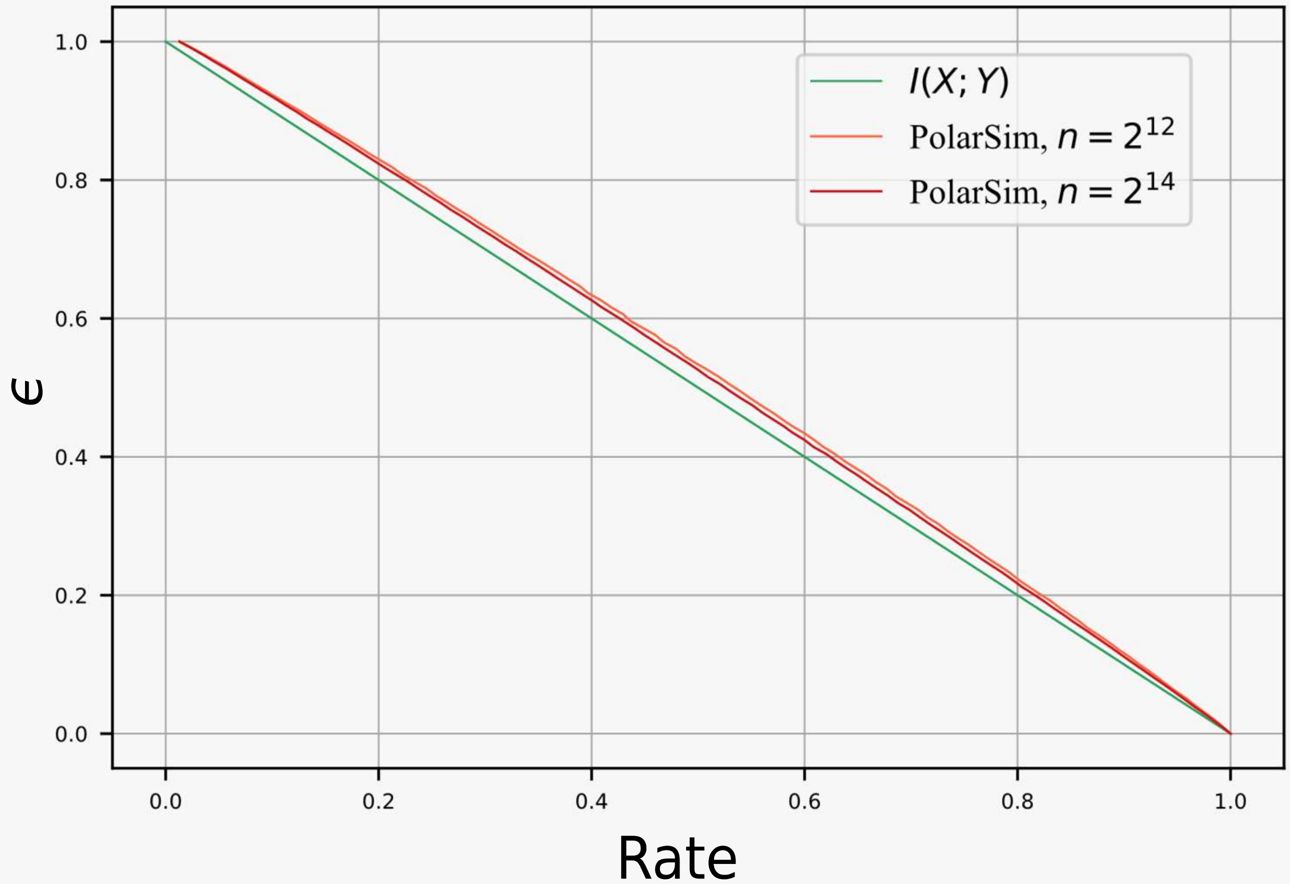
# Experimental Results: BSC

$$Y = X \oplus Z, \;\; Z \sim \text{Bern(p)}, \;\; X \sim \text{Bern}\left(\frac{1}{2}\right)$$



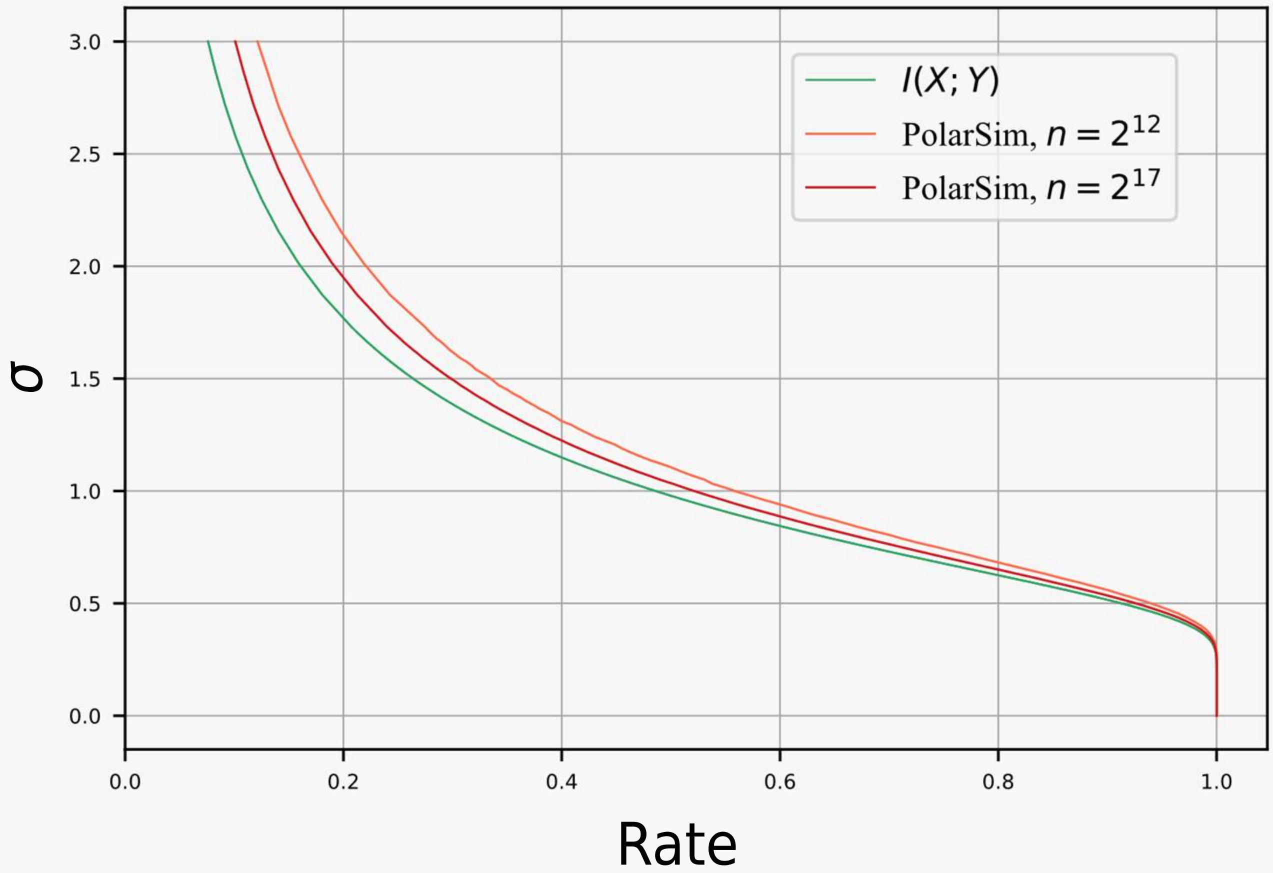
# Experimental Results: Reverse BEC

$$X = Y \cdot Z, \;\; Z \sim \text{Bern}(1-\epsilon), \;\; Y \sim \text{Unif}\{-1, 1\}$$



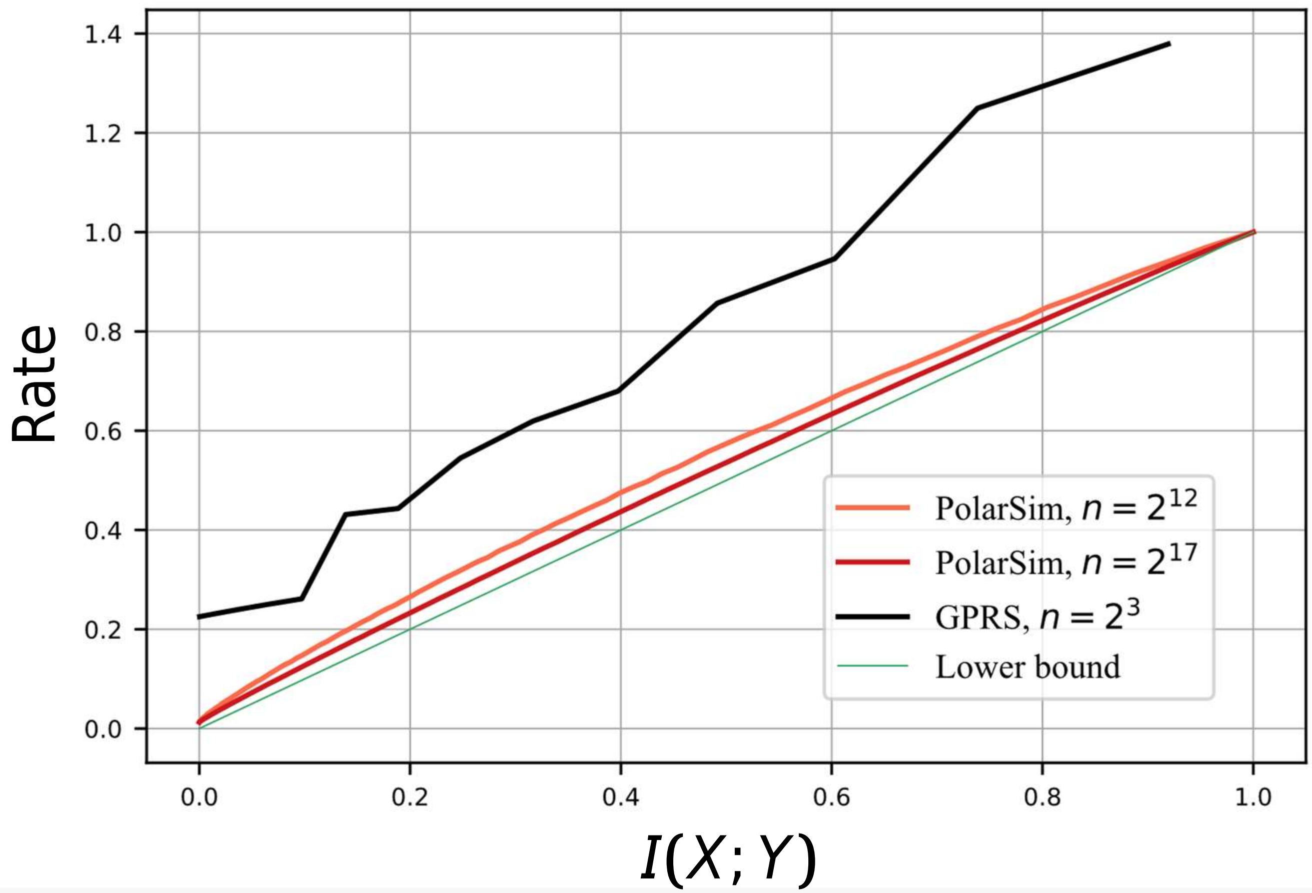
# Experimental Results: Reverse AWGN

$$X = Y + Z, \ \ Z \sim \mathcal{N}(0, \sigma^2), \ \ Y \sim \text{Unif}\{-1, 1\}$$



# Comparison with SOTA: BSC

GPRS: [Flamich, NeurIPS, 2023]



# Theorem

[Sriramu, Barsz, Polito, Wagner, 2024]

Consider a symmetric distribution $P_{XY}$ in which $Y$ is binary.

1. (*Correctness:*) Our scheme simulates the channel $p_{Y|X}^{\times n}$ exactly.
2. (*Optimality:*)

$$\lim_{n\to\infty} \frac{1}{n} E[\,\mathrm{len}(b)] \to I(X;Y),$$

   where $b$ is the output of the encoder.

3. (*Efficiency:*) The encoder and decoder have $n \log n$ complexity.