Message-Passing Decoding of Lattices Using Gaussian Mixtures

Brian M. Kurkoski

University of Electro-Communications
Tokyo, Japan

Justin Dauwels

MIT and
Harvard Medical School
(formerly RIKEN Brain Science Institute)

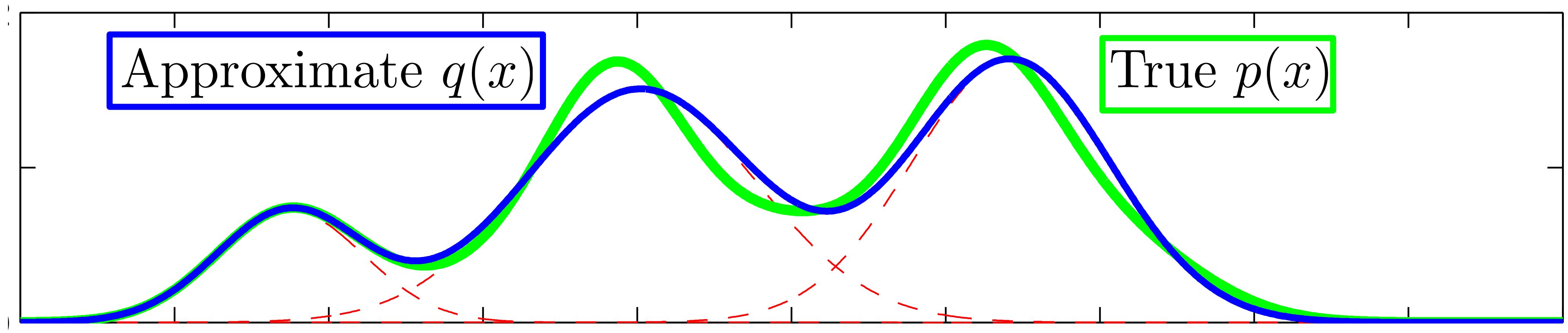
International Symposium on Information Theory
July 11, 2008
Toronto, Canada

Introduction

- Low-density lattice codes (LDLC) were proposed by Sommer, Feder and Shalvi.
- LDLC's are lattices, decoded using belief-propagation, like low-density parity check codes.
- Decoding complexity is linear in the dimension. Dimension n=10^5 possible
  - "Classical" lattices can be decoded in dimension n=$2\sim100$
- Messages (beliefs) are functions.
- Sommer et al. recognized the messages are mixtures of Gaussians, but in practice used quantized messages.

- This talk has two parts:
  - Propose **Gaussian Mixture Reduction algorithm**: Approximate a mixture of Gaussians by a smaller number of Gaussians.
  - Review LDLC lattices. Apply our algorithm to LDLC decoding. Messages that are true mixtures of Gaussians. **The proposed algorithm performs as well as quantization.**

How do we approximate a mixture of N Gaussians with M Gaussians, $M < N$?



Given:

A mixture of N Gaussians,

$$p(x) = \sum_{i=1}^{N} c_i \mathcal{N}(z; m_i, v_i)$$

with known mixing coefficients c_i, means m_i, variances v_i,

Find:

A mixture of M Gaussians, $q(x)$, which is a good approximation, $M < N$:

$$q(x) \approx p(x).$$

Other Applications: Kernel density estimation, classification, speech estimation, compressed sensing.

Global Minimization of KL Divergence

- Kullback-Leibler divergence is widely used as a measure of similarity of two distributions.
  - EM algorithm, variational methods, generalized belief propagation can be phrased as KL minimization.
- Ideally, minimize over all possible distributions of M Gaussians $\mathcal{Q}$:

$$q(x) \quad = \quad \arg\min_{q' \in \mathcal{Q}} \mathrm{KL}\big(\ p(x) \big|\big| q'(x)\ \big)$$

- Possible when $M = 1$: moment matching.
- Optimal approaches with $M > 1$ appear computationally demanding
- Not suitable for:
  - Large data sets (kernel density estimation)
  - **Decoding algorithm (our problem of LDLC decoding)**

Gaussian Mixture Reduction Algorithm

The proposed Gaussian Mixture Reduction algorithm uses greedy, pairwise replacement of Gaussians

Begin with a mixture of N Gaussians

1. Compute a **distance metric** between all pairs of Gaussians
2. For the pair with the lowest metric, **combine** into a single Gaussian
    - Now there is one fewer Gaussians

Repeat from Step 1 until a stopping condition is reached.

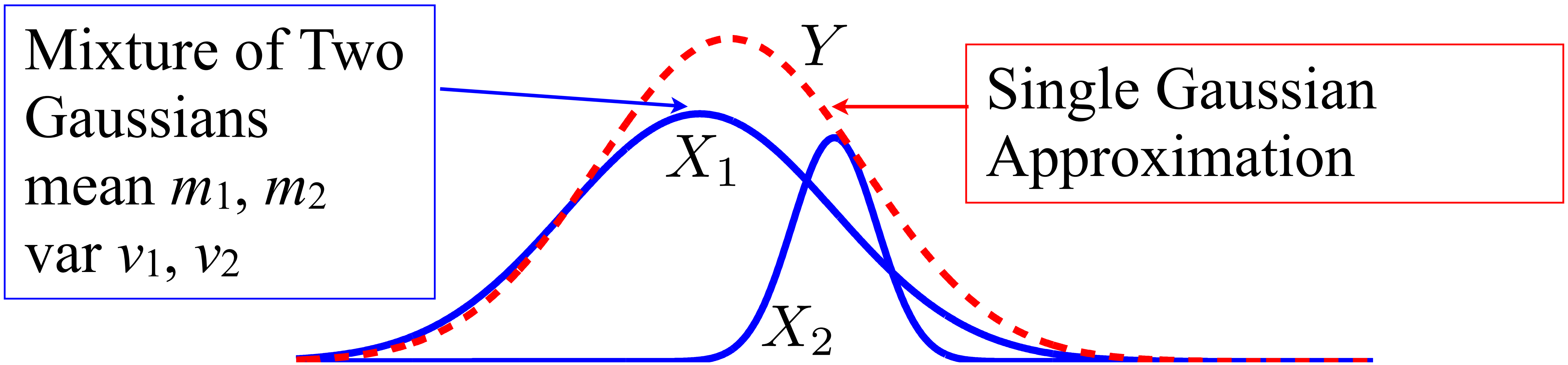
Combining — Easy

- Moment matching: the moments of the two distributions are equal
- Moment matching minimizes KL divergence

Distance Metric — Hard

- KL divergence is hard to compute efficiently
- Instead, use the squared distance.
    - Squared distance is a lower bound on KL divergence

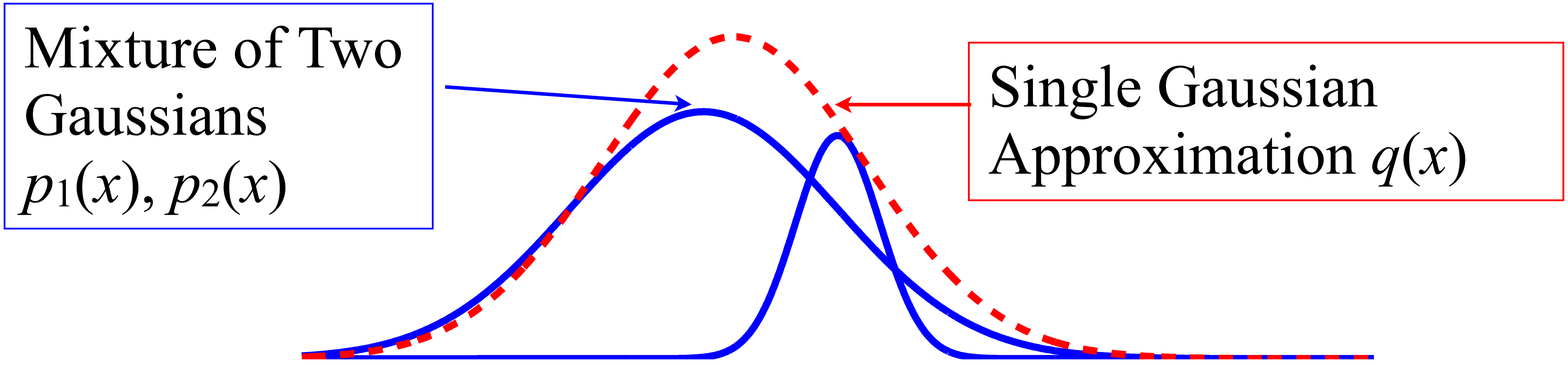
Moment Matching: Replace Two Gaussians with A Single Gaussian



- Find a single Gaussian which is a good approximation of two Gaussians
  - Use "moment matching": the mean and variance of the mixture is equal to the mean and variance of the single Gaussian
  - Minimizing the Kullback-Leiber divergence leads to moment matching

$$
\begin{aligned}
E[Y] &= c_1 m_1 + c_2 m_2 \\
E[Y^2] &= c_1 \cdot (v_1 + m_1^2) + c_2 \cdot (v_2 + m_2^2)
\end{aligned}
$$

Distance Metric: Squared Distance has Closed Form



The local divergence between p_1+p_2 and q:

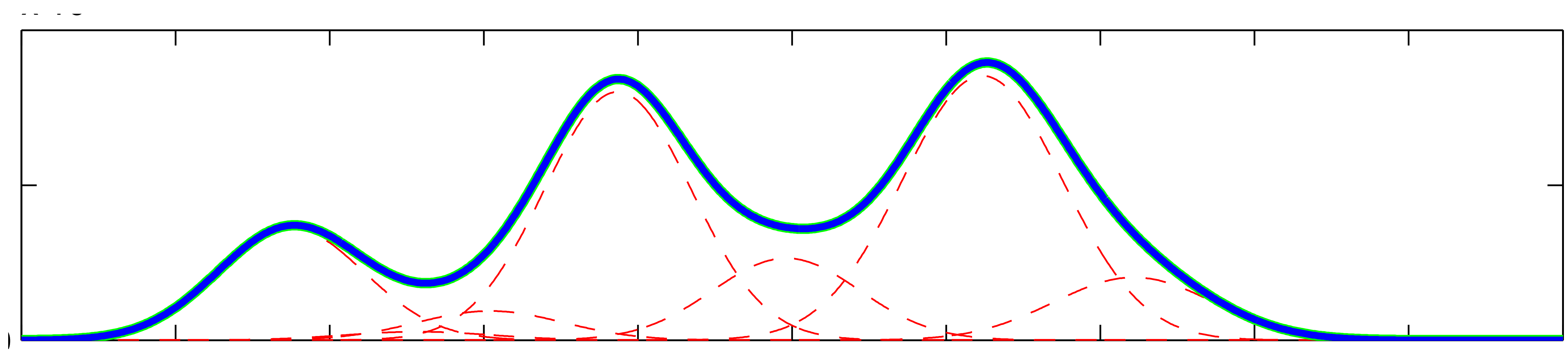
$$\mathrm{KL}\big(p_1(x) + p_2(x)\big|\big|q(x)\big)$$

also has no closed-form solution. However, the squared distance:

$$\mathrm{SD}\big(p(x)\big|\big|q(x)\big) \quad = \quad \int \big(p(x) - q(x)\big)^2 dx$$

does have a closed-form solution. Showed that the KL divergence is lower bounded by the squared distance.

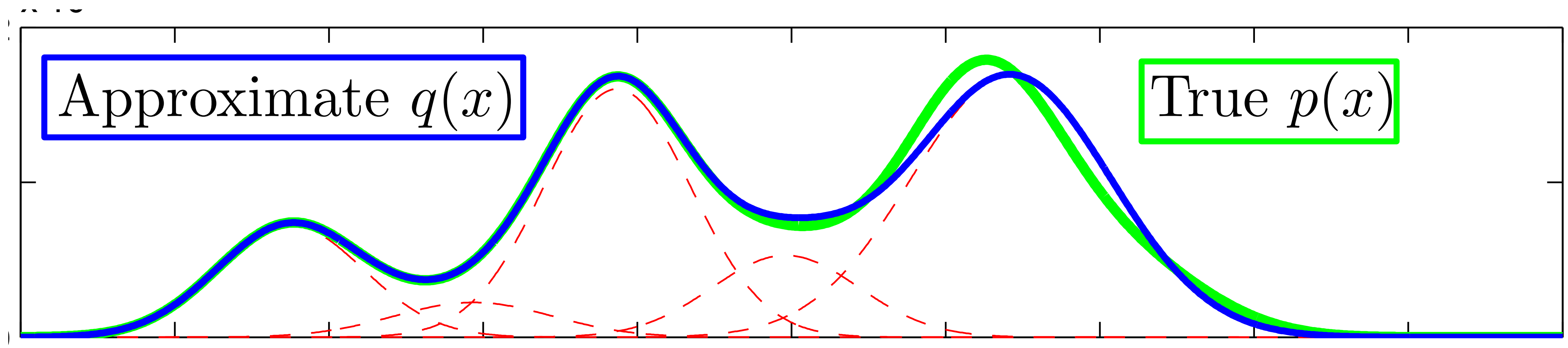
Example:
Gaussian Mixture Reduction Algorithm



Begin: 12 Gaussian Mixture

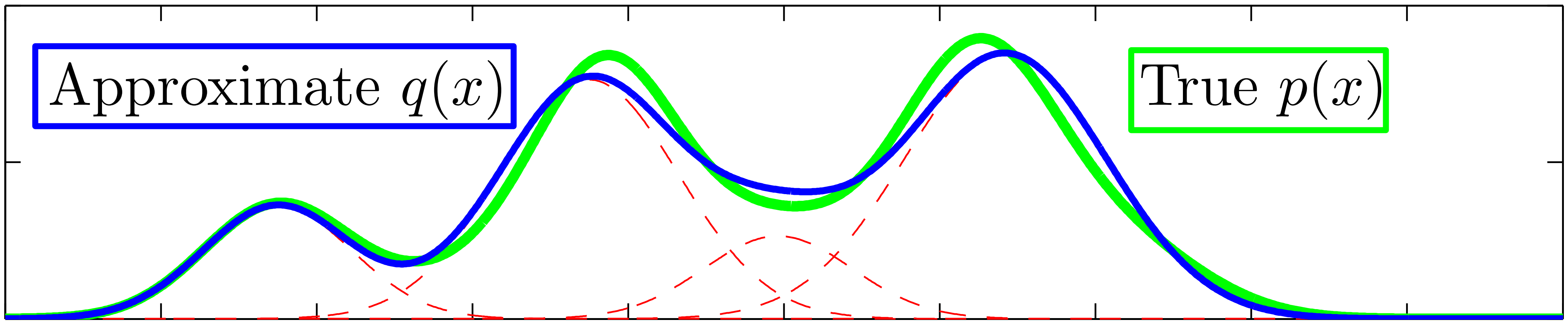
- Looks like it would be well approximated by a three-Gaussian function

Example
Gaussian Mixture Reduction Algorithm



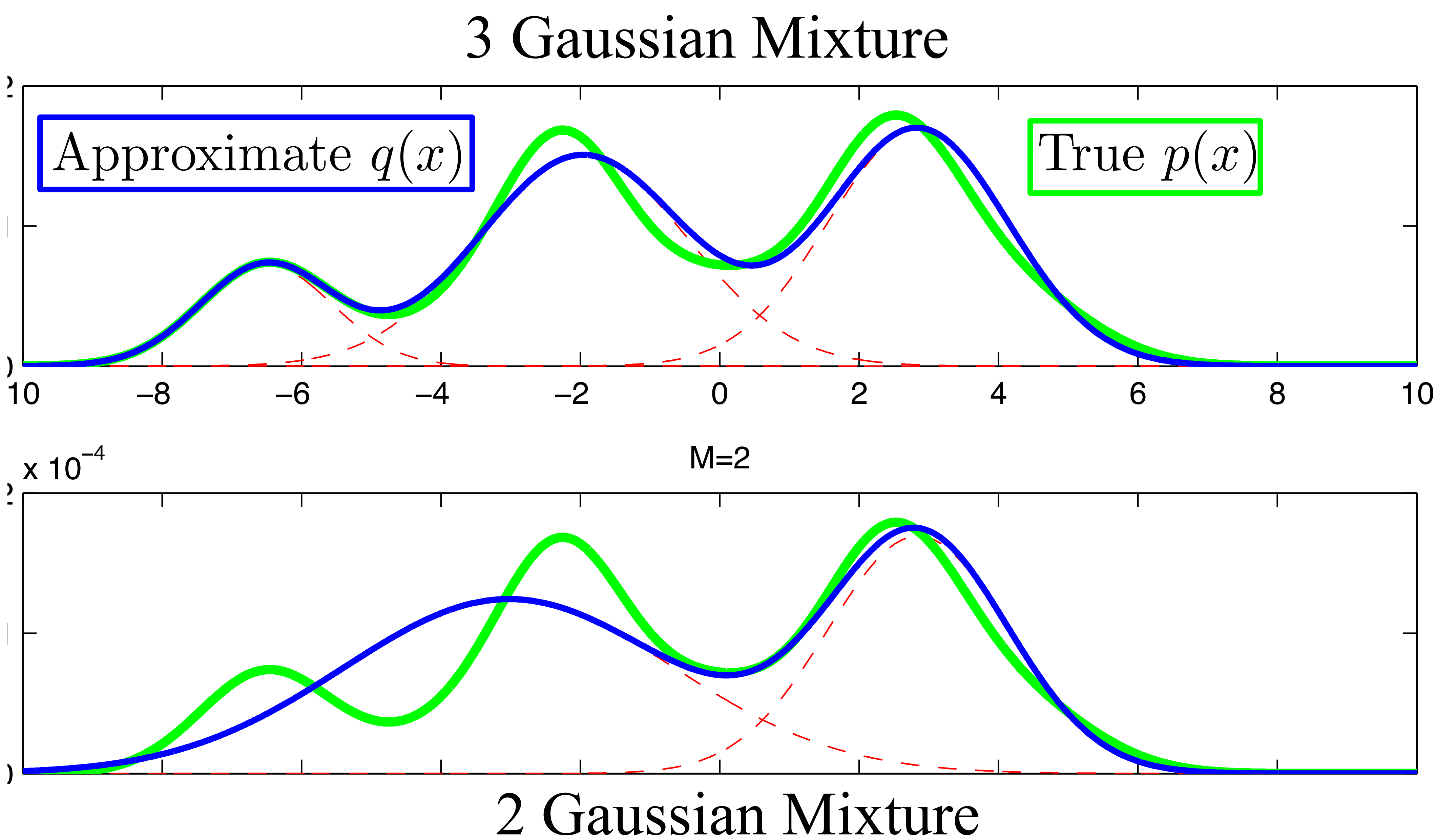
After 7 iterations: 5 Gaussian Mixture

Example
Gaussian Mixture Reduction Algorithm



4 Gaussian Mixture

Example
Gaussian Mixture Reduction Algorithm



- When error (squared distance) becomes too large, stop.

Lattices

A lattice is an infinite and regular set of points $\mathbf{x}$ in n-dimensional Euclidean space:

$$\mathbf{x} = G \cdot \mathbf{b}$$

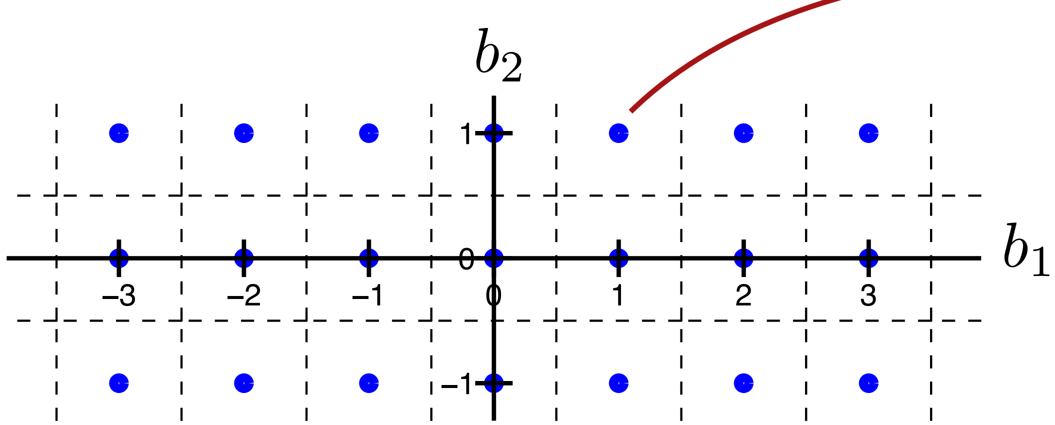
$\mathbf{b} = (b_1, b_2, \ldots, b_n)$: n-by-1 vector of integers

G: n-by-n generator matrix

$\mathbf{x} = (x_1, x_2, \ldots, x_n)$: n-by-1 vector, lattice point

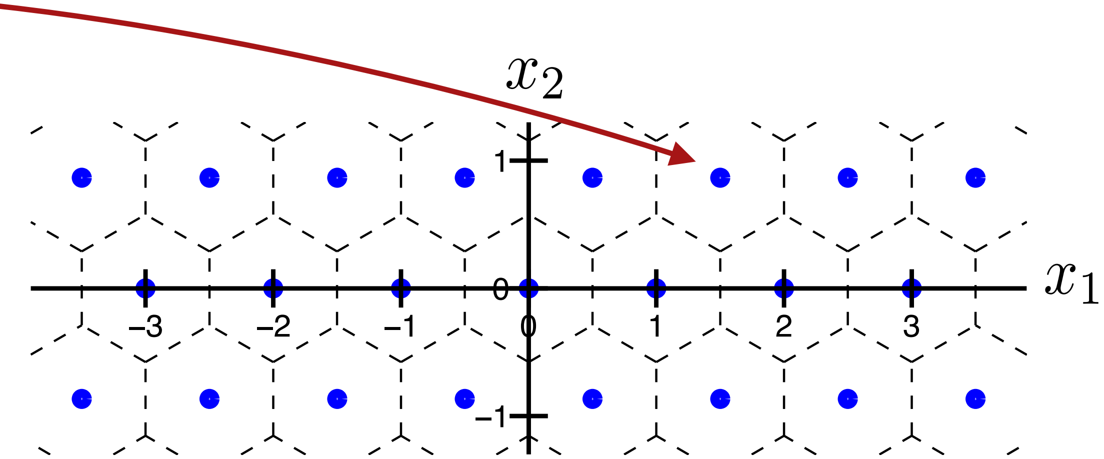
e.g. n =2, $G = \begin{bmatrix} 1 & \frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} \end{bmatrix}$

Pictures n=2
Lattices n=100~100000



$\mathbf{x} = G \cdot \mathbf{b}$

$\Longrightarrow$



Low-Density Lattice Codes (LDLCs)

Sommer, Feder and Shalvi gave a lattice construction and decoding algorithm based upon low-density parity-check codes. Extensive convergence analysis in IT Trans, April 2008.

Low-Density Parity-Check Codes Codes

- Code over a finite field (binary)
- Sparse parity check matrix

LDLC

- Lattice: Code over the real numbers
- Inverse generator $H=G^{-1}$ is sparse

Low-Density Parity-Check Codes Codes

Variables are from the field

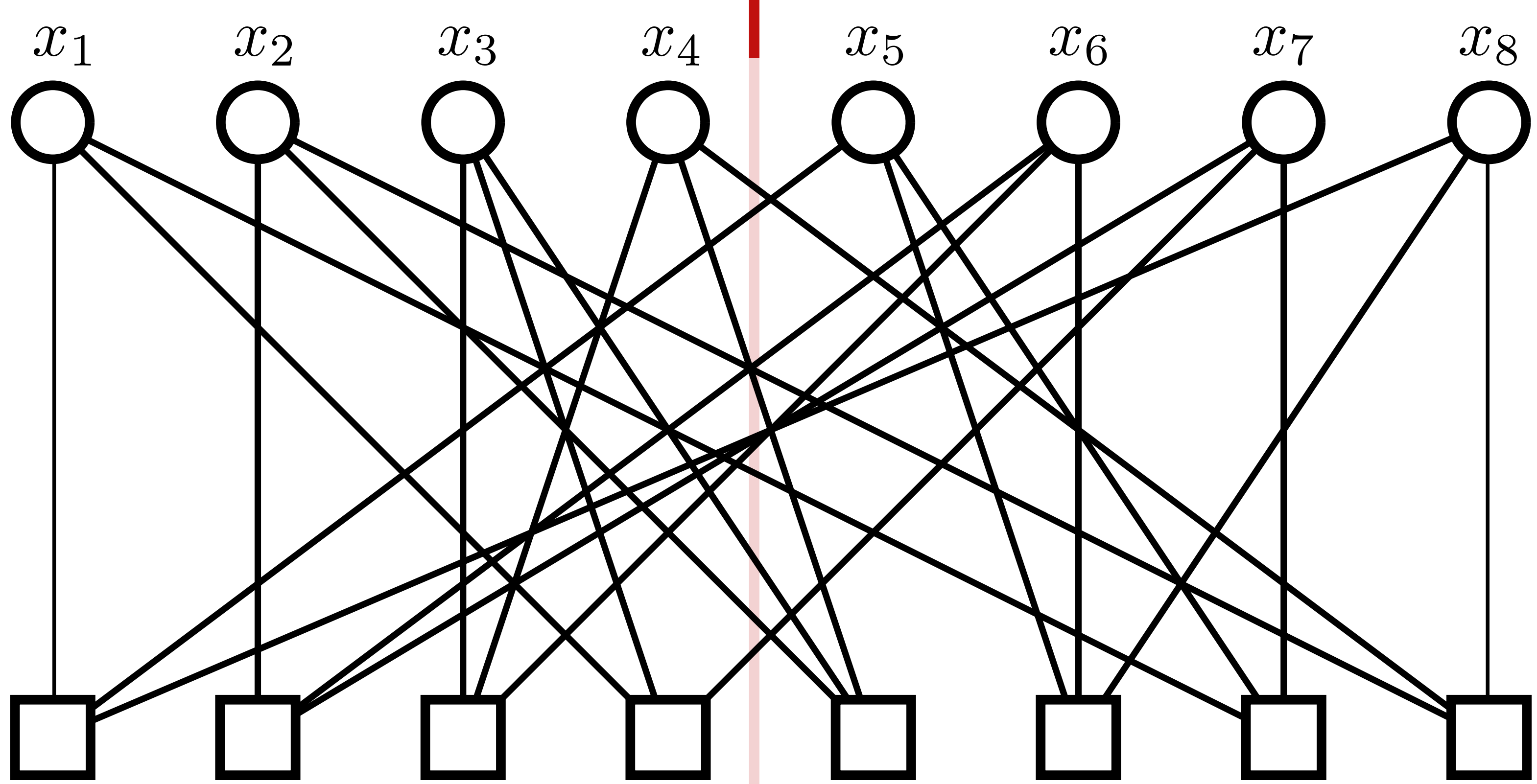
$x_i \in \{0, 1\}$

So, messages are scalars (LLRs)

check nodes < # variable nodes

$x_1 + x_2 + x_3 = 0$ (over field)

Approaches BIAWGN channel capacity

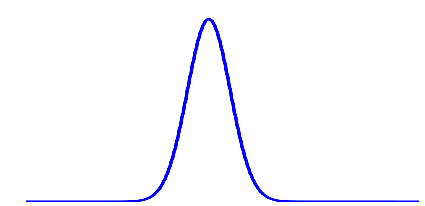


Low-Density Lattice Codes (LDLC)

Variables are real numbers

$x_i \in \mathbb{R}$

So, messages are functions (!)



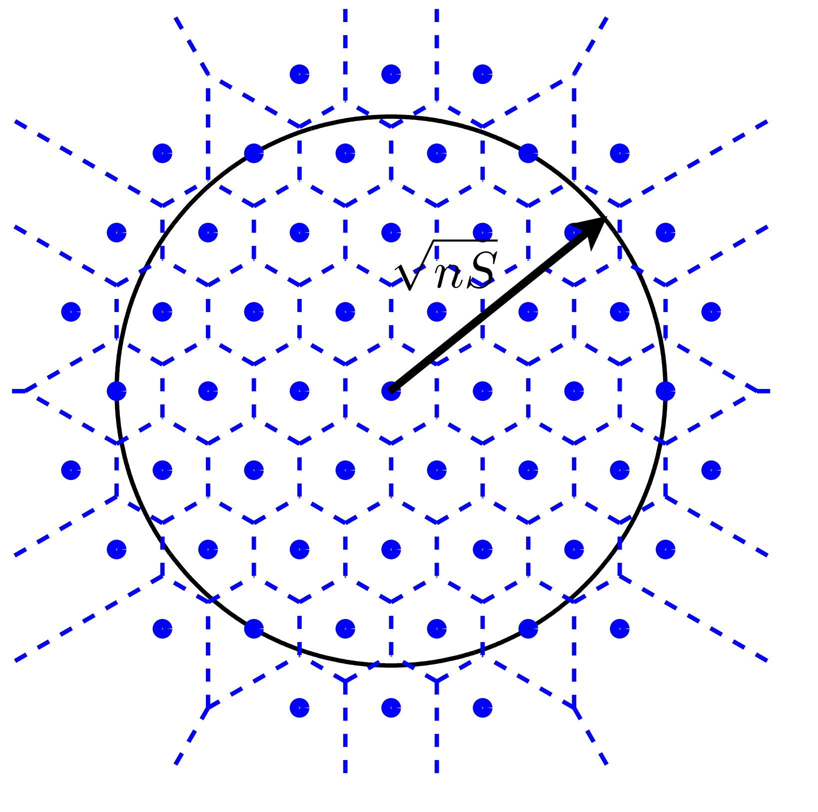
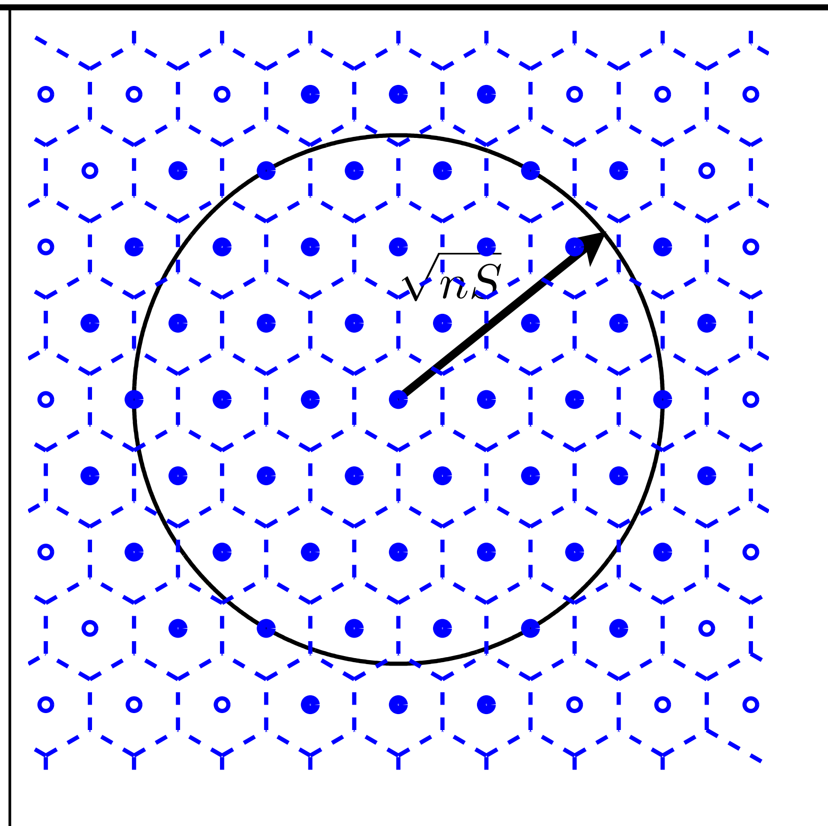
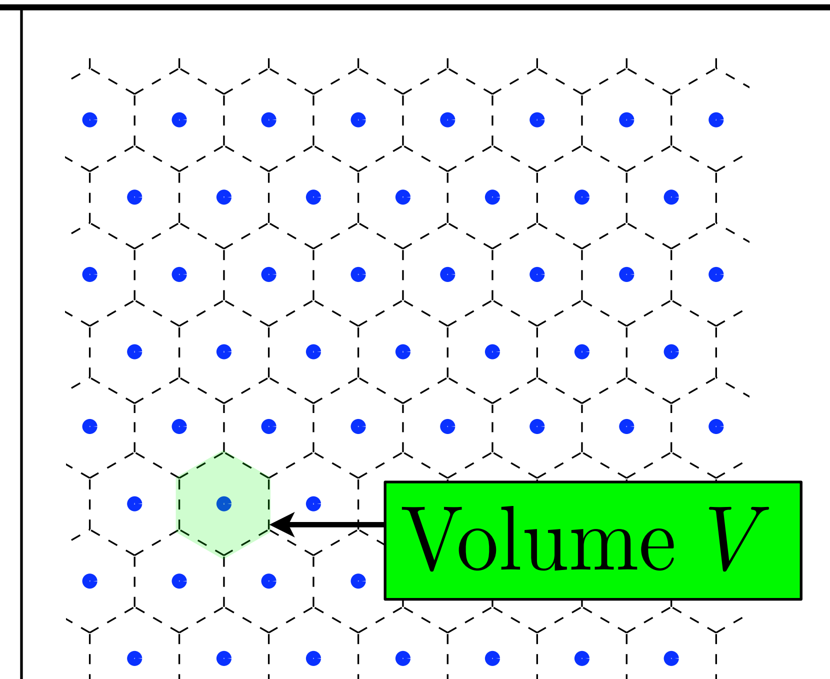
check nodes = # variable nodes

$x_1 + x_2 + x_3 = \mathrm{b}$ (over real numbers)

b is an integer

Comes within 0.6 dB of a specialized communications problem

Lattices Can Achieve Channel Capacity

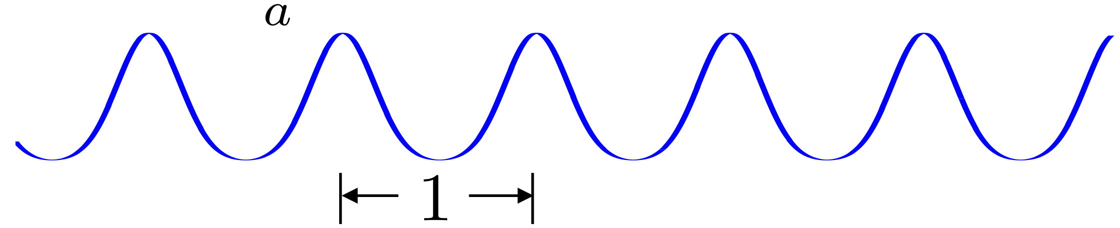
<table>
<tr>
<td></td>
<td></td>
<td rowspan="2">
Code: Lattice only
Power constraint through volume</td>
</tr>
<tr>
<td colspan="2">Code: Lattice and shaping region, power constraint S</td>
</tr>
<tr>
<td>Dec.:Maximum likelihood</td>
<td colspan="2">Decoder: Lattice decoding</td>
</tr>
<tr>
<td>Achieves Shannon limit:
$$R \leq C = \frac{1}{2}\log(1+\frac{S}{N})$$</td>
<td>Achieves only:
$$R \leq \frac{1}{2}\log(\frac{S}{N})$$</td>
<td>No power constraint:
$$N \leq \frac{\sqrt[n]{V^2}}{2\pi e}$$</td>
</tr>
<tr>
<td>Urbanke & Rimoldi 1998
Erez & Zamir 2004</td>
<td>de Buda, 1975
Loeliger, 1997</td>
<td>Poltyrev, 1994</td>
</tr>
</table>

LDLC Decoder: Check Node

Convolution:

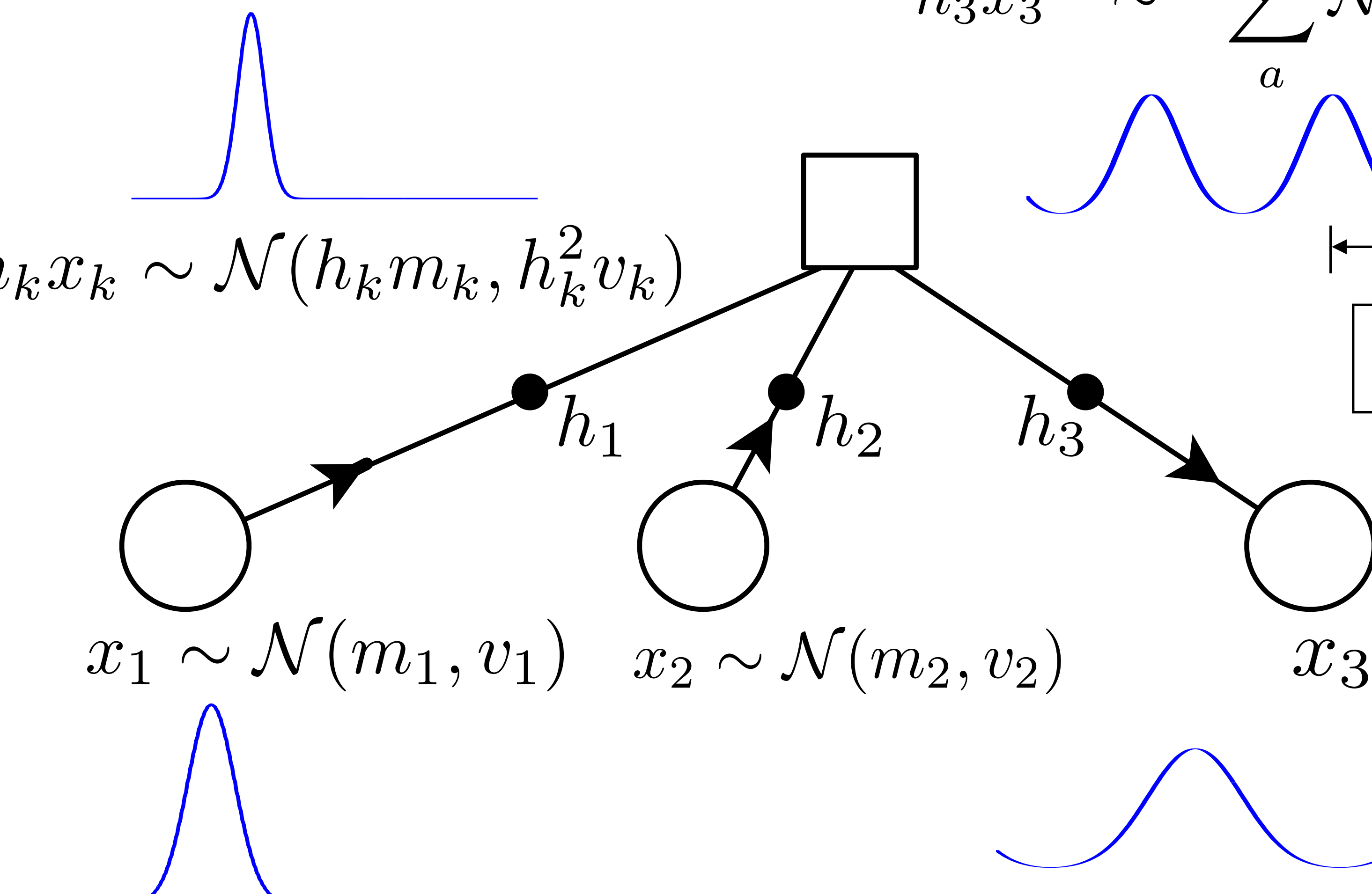
$$h_3 x_3 \sim \mathcal{N}(h_1 m_1 + h_2 m_2, h_1^2 v_1 + h_2^2 v_2)$$

$$h_3 x_3 \sim \sum_a \mathcal{N}(h_1 m_1 + h_2 m_2 - a, h_1^2 v_1 + h_2^2 v_2)$$



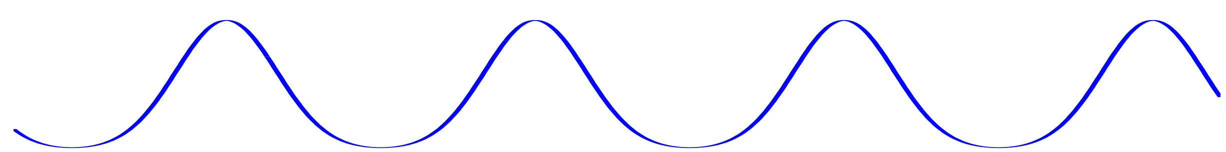
Shift-and-repeat: unknown integer

$$h_k x_k \sim \mathcal{N}(h_k m_k, h_k^2 v_k)$$



$x_1 \sim \mathcal{N}(m_1, v_1)$ $x_2 \sim \mathcal{N}(m_2, v_2)$ x_3

Output: Always a Mixture

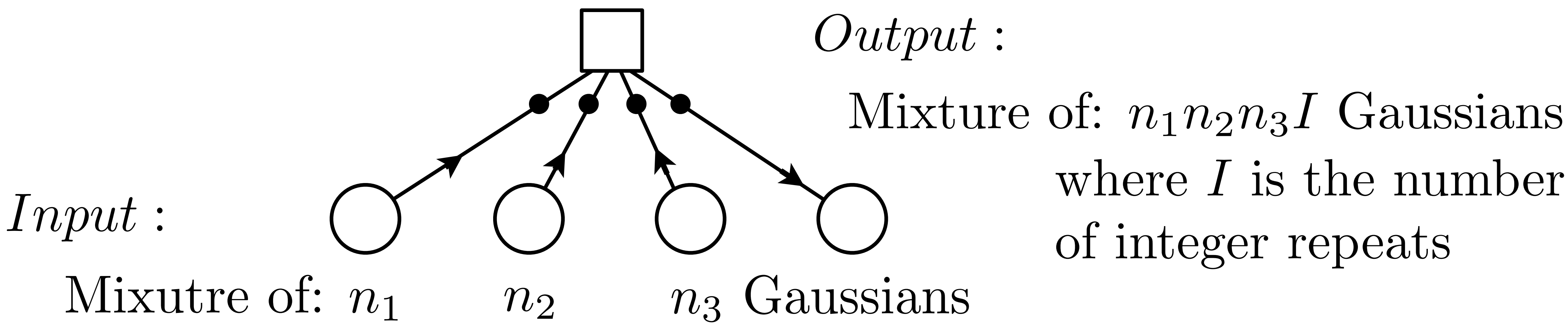


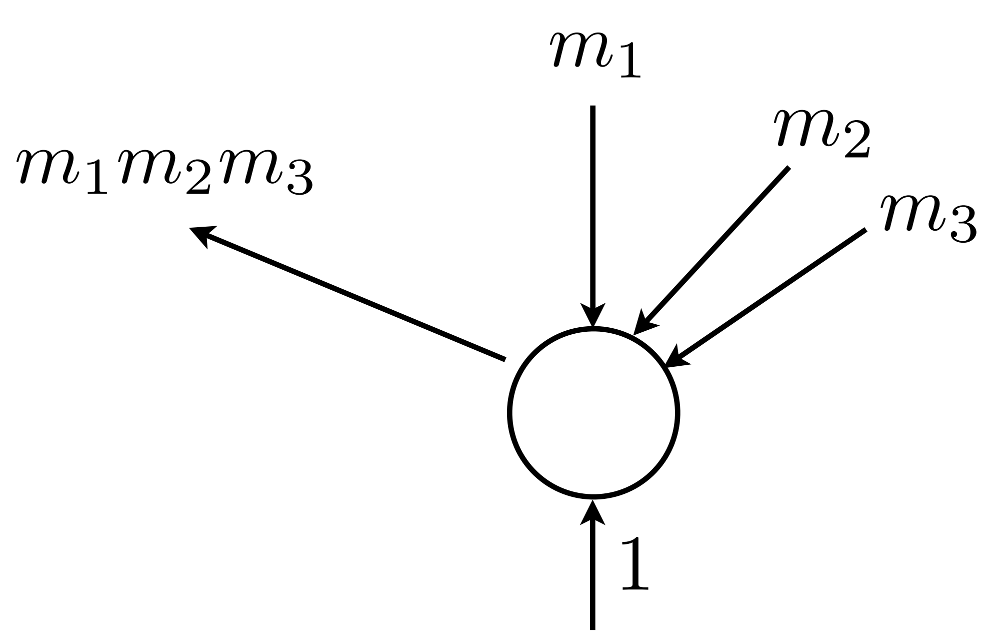
$$x_3 \sim \sum_a \mathcal{N}\left(-\frac{h_1 m_1 + h_2 m_2 - a}{h_3}, \frac{h_1^2 v_1 + h_2^2 v_2}{h_3^2}\right)$$

Input: Single Gaussian or Mixture

- All operations preserve Gaussianity: input is a Gaussian, output is a Gaussian
- Similarly at variable node: Gaussians input, Gaussian output

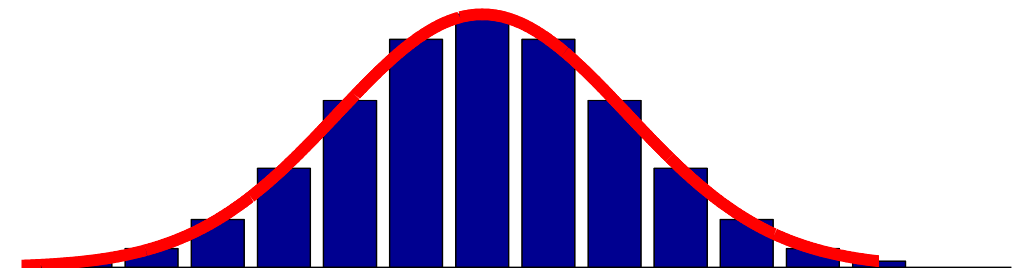
Explosion of Gaussians

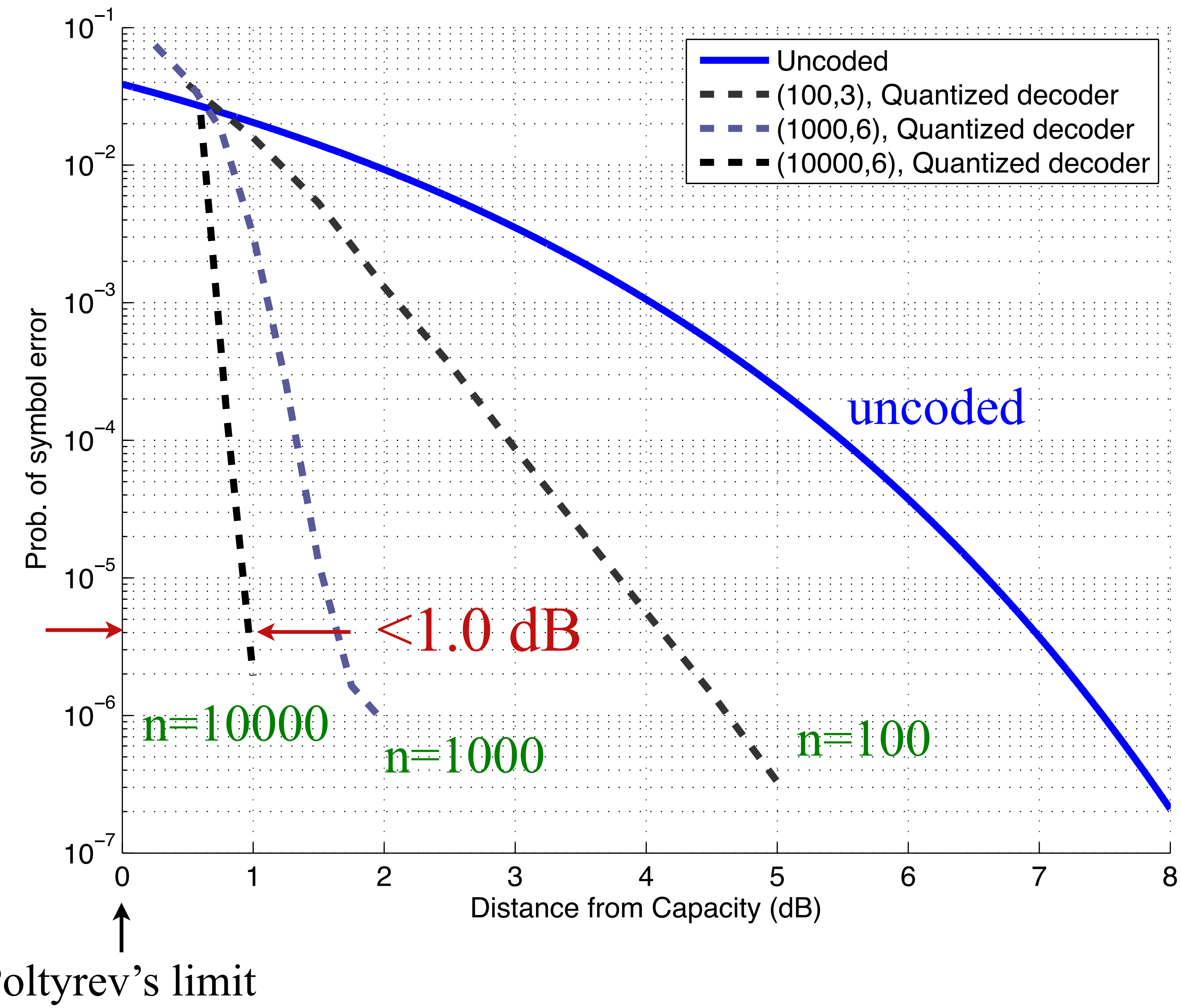




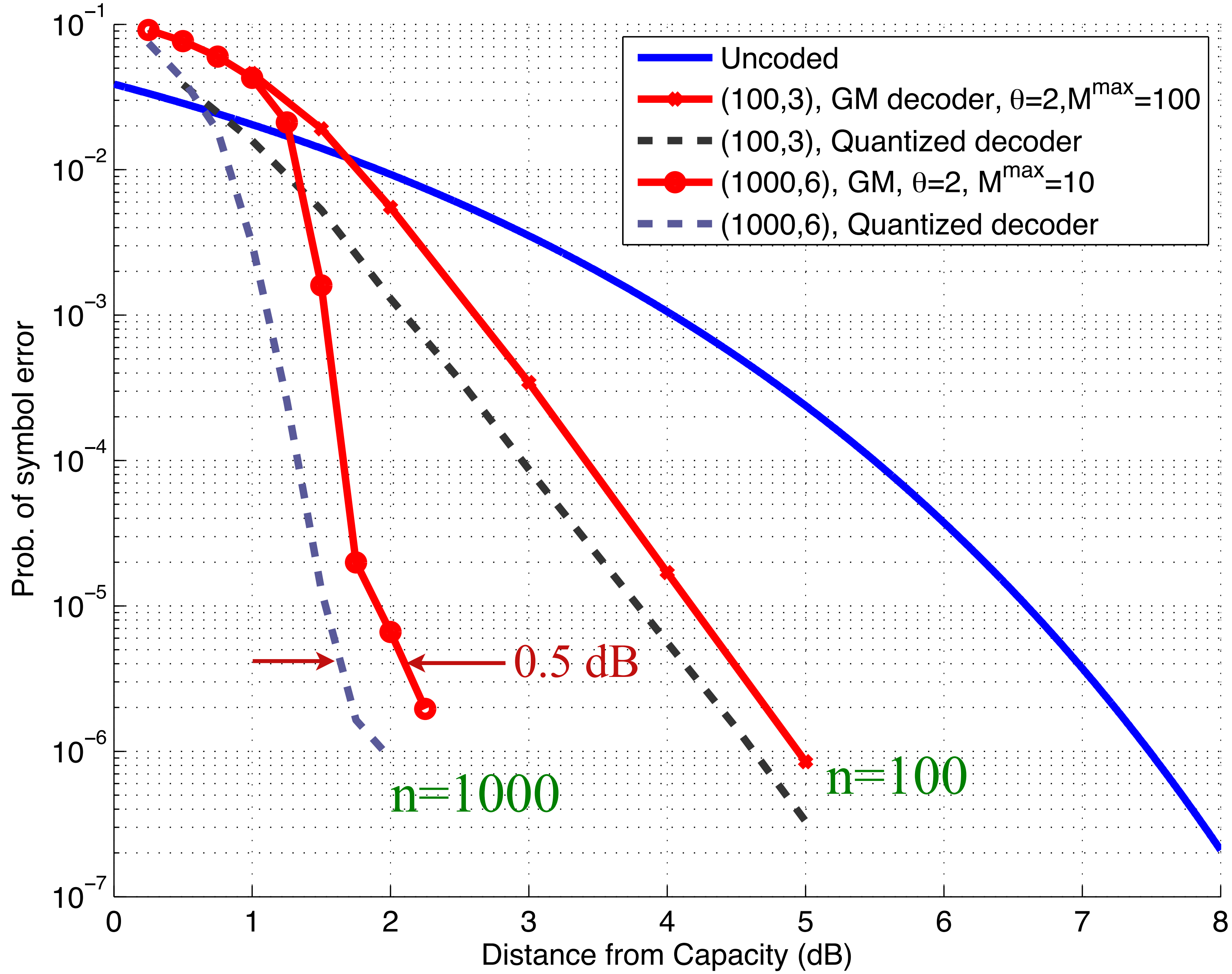
- Number of Gaussians grows quickly
- **Prior work**: quantize the Gaussians
- **Our work**: apply the Gaussian-mixture reduction algorithm

LDLC Performance: Quantized Decoder

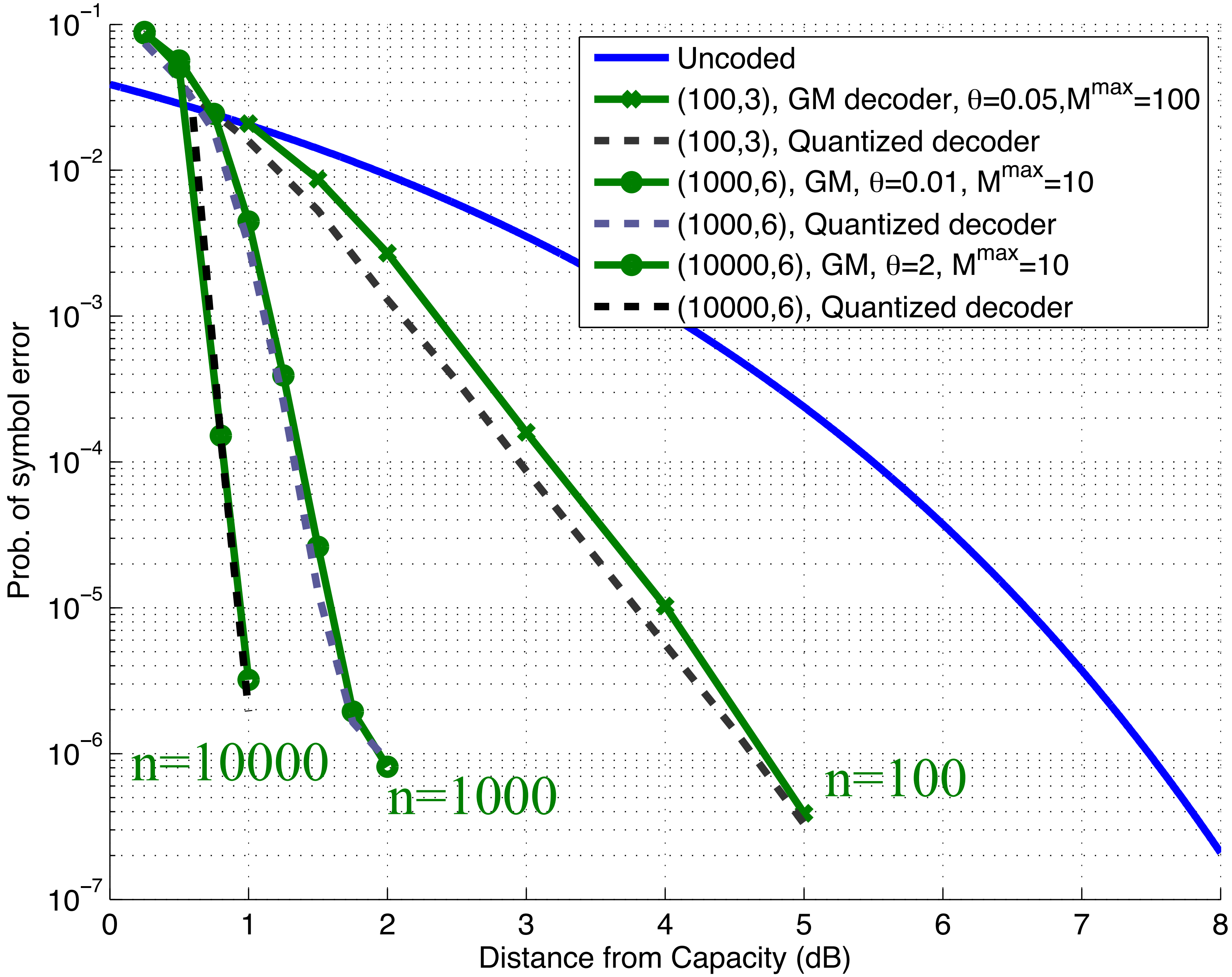




Gaussian Mixture Decoding
Max allowed penalty θ=2

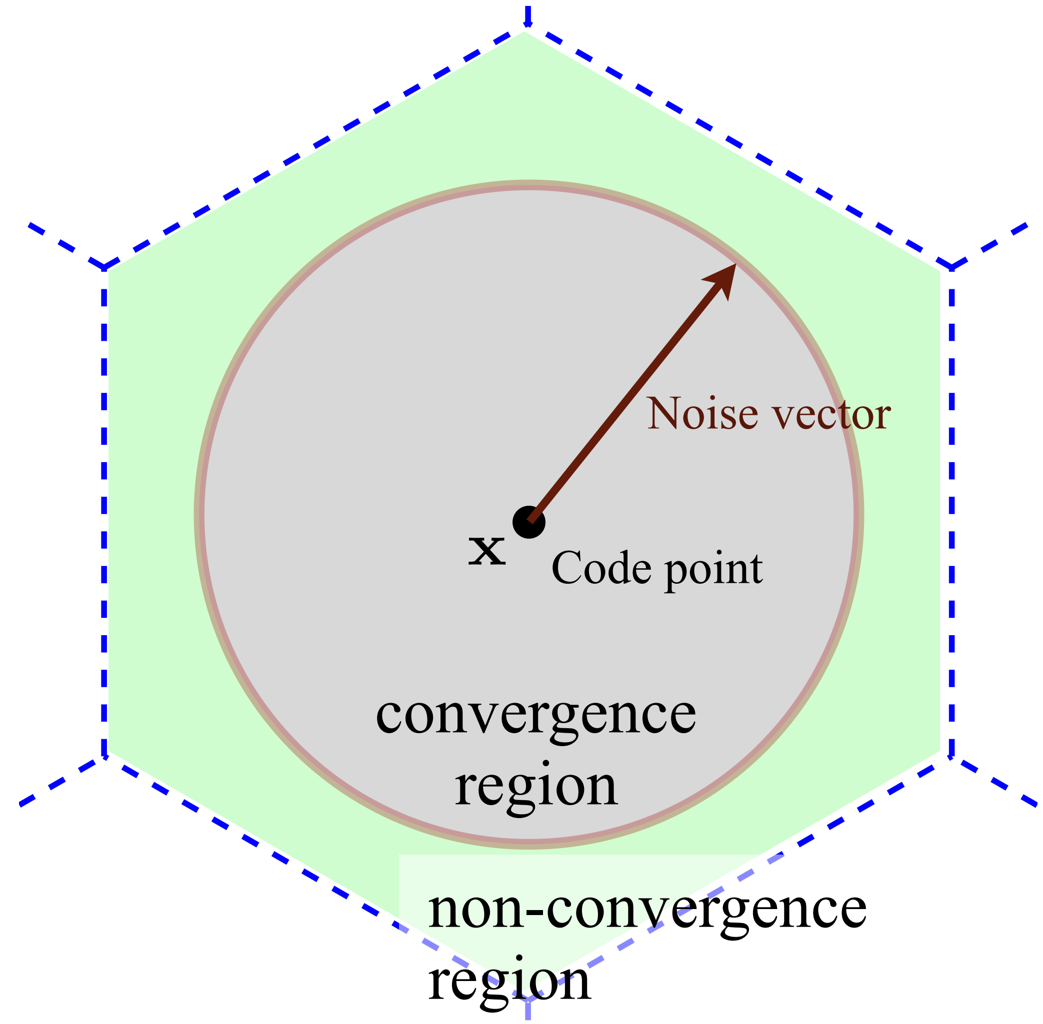


Gaussian Mixture Decoding
Max allowed penalty θ=0.01



Open Problem

- Convergence is obtained for lattice point + noise.
- BP fails for general quantization, greater noise.
- A high dimension **lattice quantizer** is of great interest for achieving channel capacity, dirty paper coding, lossy source coding.



Summary

- **Low density lattice codes**, proposed by Sommer, Feder and Shalvi are interesting.
- Proposed a **Gaussian mixture reduction algorithm**, which approximates a mixture of Gaussians with a smaller number of Gaussians
- This algorithm can be applied to LDLC decoding, and shows no performance loss
    - "Nicer" to represent the messages as Gaussians, amenable for further analysis.

BACKUP

Complexity: Memory

- Quantized: Used 1024 points to quantize each point
- Gaussian mixtures: Each message requires a maximum of 10x3= 30 numbers

Complexity: Computation

- Quantized: computation is dominated by a fast Fourier transform, we used length 128
- Gaussian mixtures: Computation is about $O(M^4)$ where
  - Distribution of M depends upon other parameters.