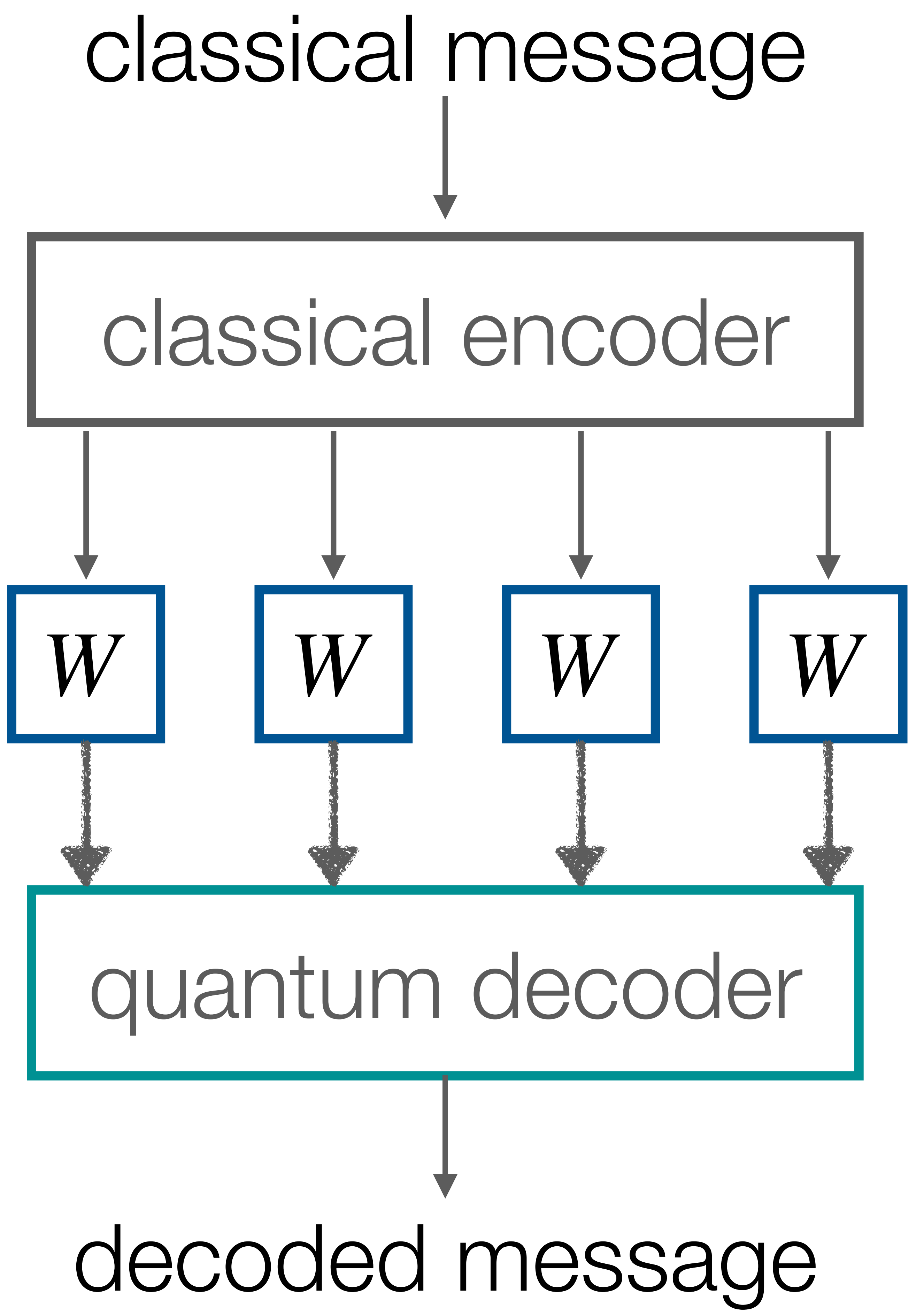
# Quantum message-passing algorithm for optimal and efficient decoding

Christophe Piveteau and Joseph M. Renes

Institute for Theoretical Physics, ETH Zürich

2022 IEEE
INTERNATIONAL SYMPOSIUM
ON INFORMATION THEORY
JUNE 26–JULY 1 AT AALTO UNIVERSITY
IN ESPOO, FINLAND

# Simple quantum decoding problem



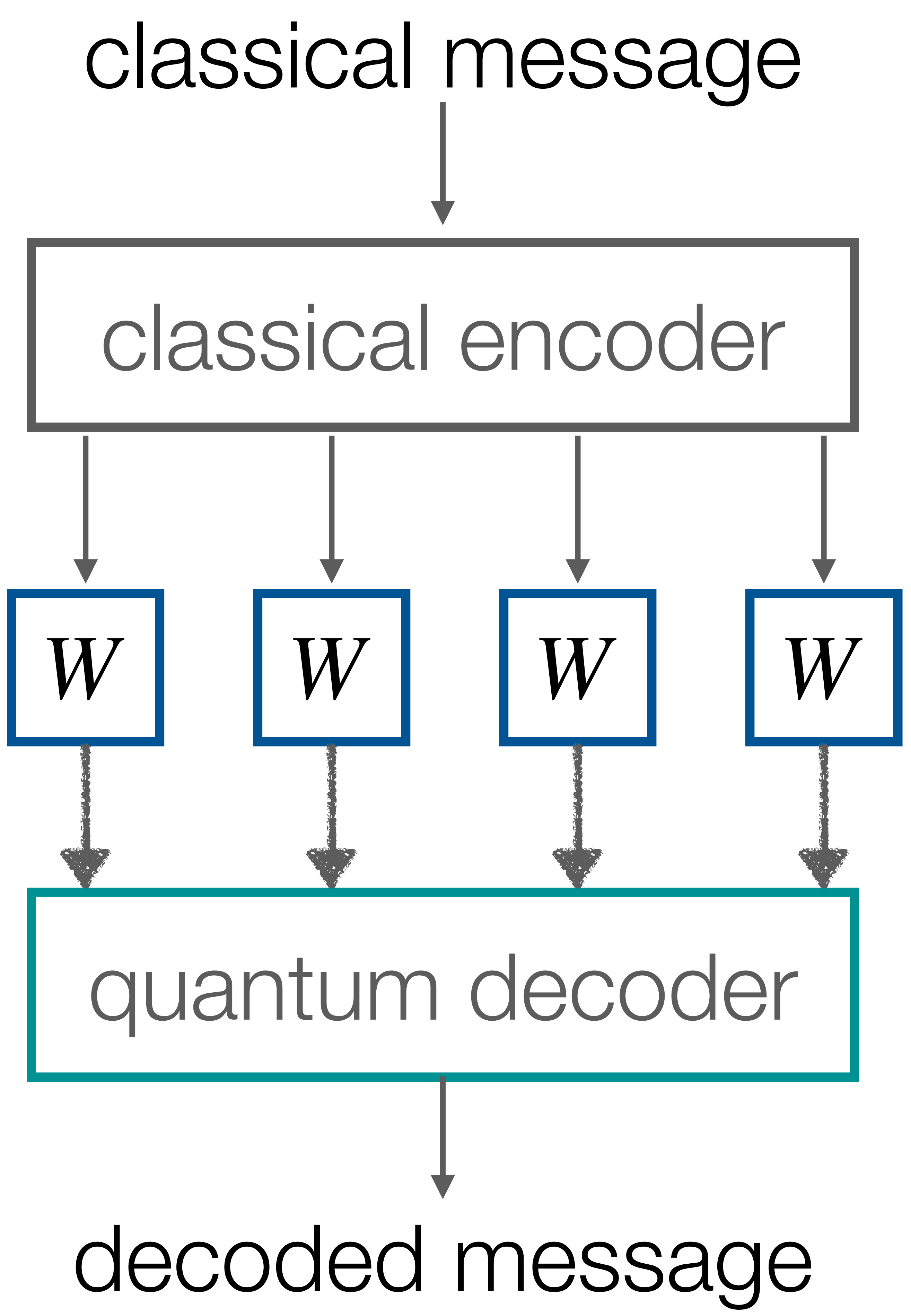

*uniformly random*

*linear code*

*CQ channel*

*????*

# Simple quantum decoding problem




*uniformly random*

*linear code*

*CQ channel*

*Follow BP and try to decode bitwise…*

# BPQM algorithm

- Introduced at ISIT 2017: "Belief propagation decoding of quantum channels by passing quantum messages"
- Studied by Rengaswamy et al. at ISIT 2020
  - Simplification in sequential decoding
  - Block optimality in a 5-bit example
- What's new this year?
  - Actual message passing version — original does not pass *all* info!
  - Efficient implementation — above flaw means original algorithm not efficient!
  - Application to non-tree codes via approximate cloning
  - Proof of block optimality for all tree codes

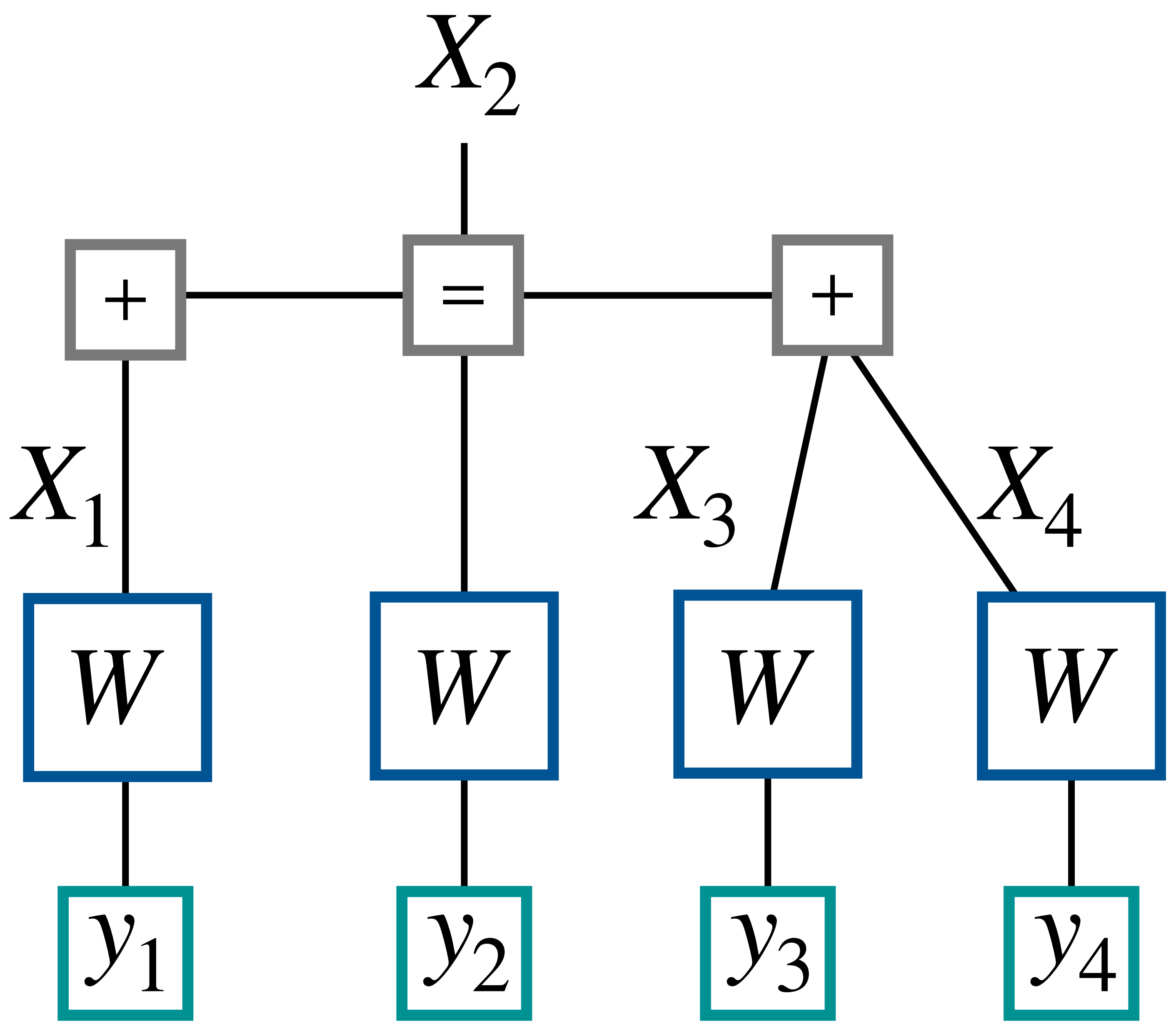
# Outline

- Variation of classical BP
- BPQM: Passing quantum messages for single bit estimation
- Successive BPQM for entire codewords
- Loopy BPQM
- Summary and open questions

# Outline

- Variation of classical BP
- BPQM: Passing quantum messages for single bit estimation
- Successive BPQM for entire codewords
- Loopy BPQM
- Summary and open questions

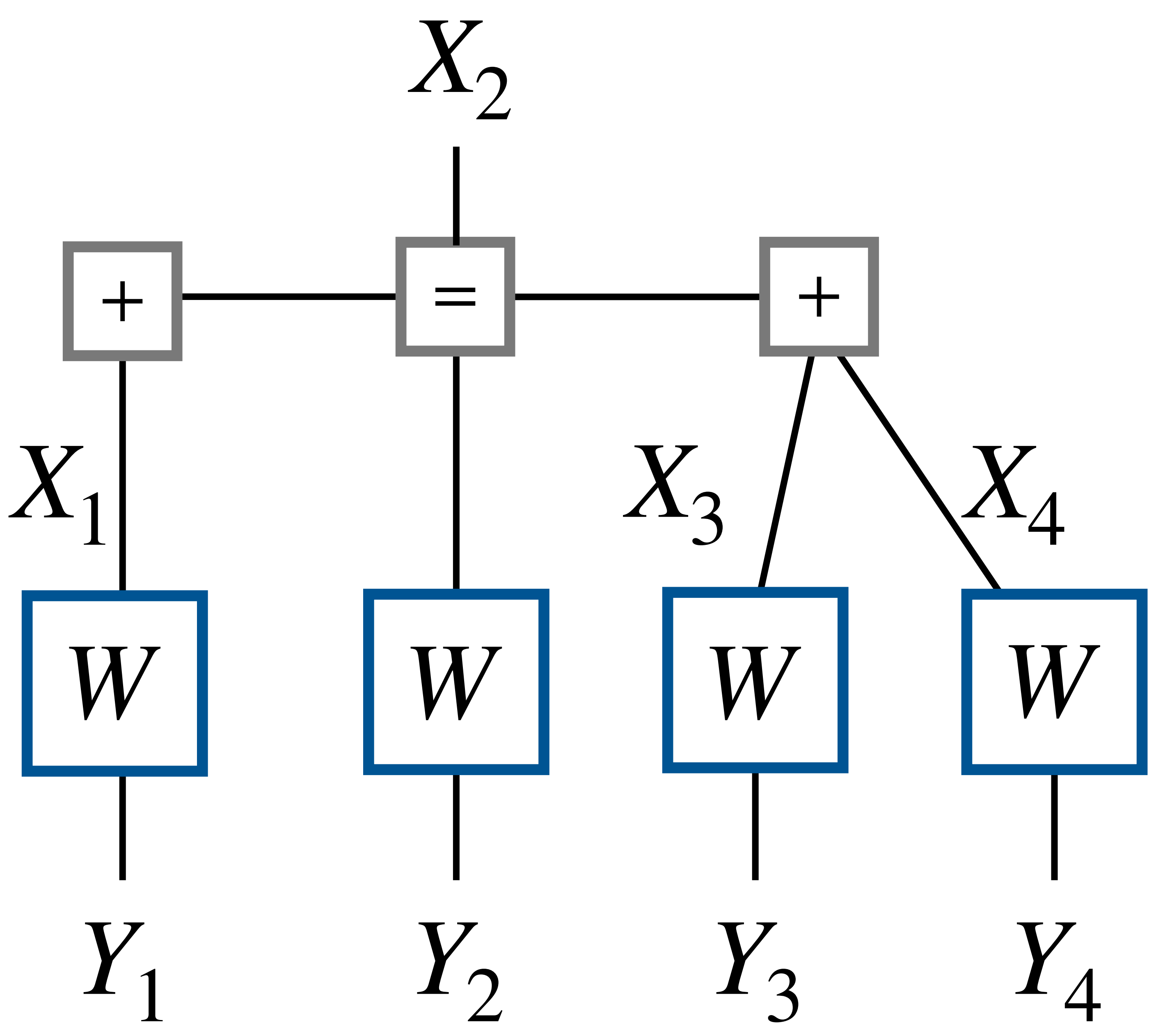
# Belief propagation decoding as tensor network contraction




Contract to find estimate of $X_2$ given observed $y_1y_2y_3y_4$.

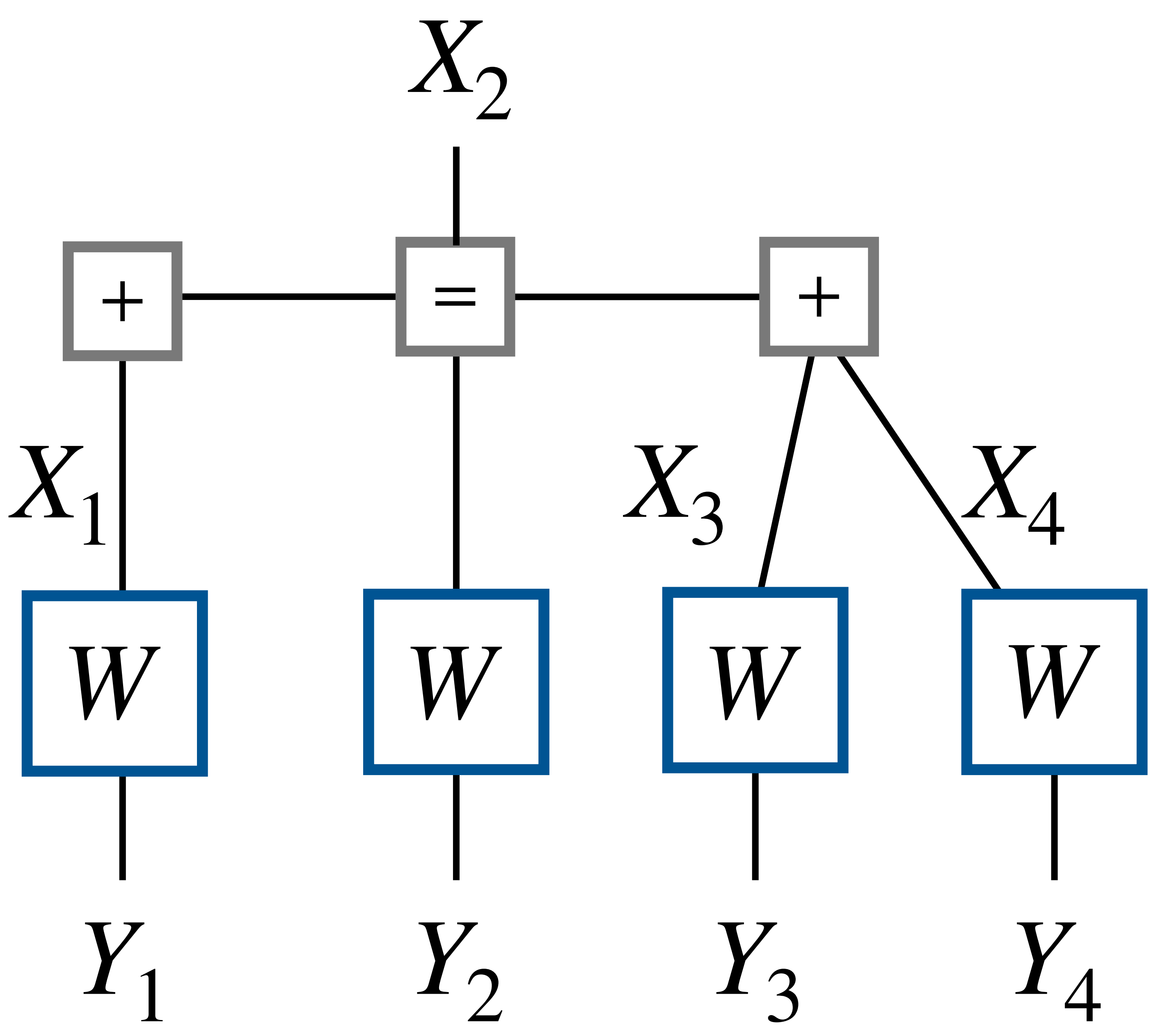
Run in parallel to estimate all other codeword bits.

# Belief propagation decoding acting on output bits: BSC




- Associate a bit $b$ and likelihood $\ell = \frac{\delta}{1-\delta}$ to each node
- Traverse tree from leaves to root, generating node $(b,\ell)$ data from children node data.

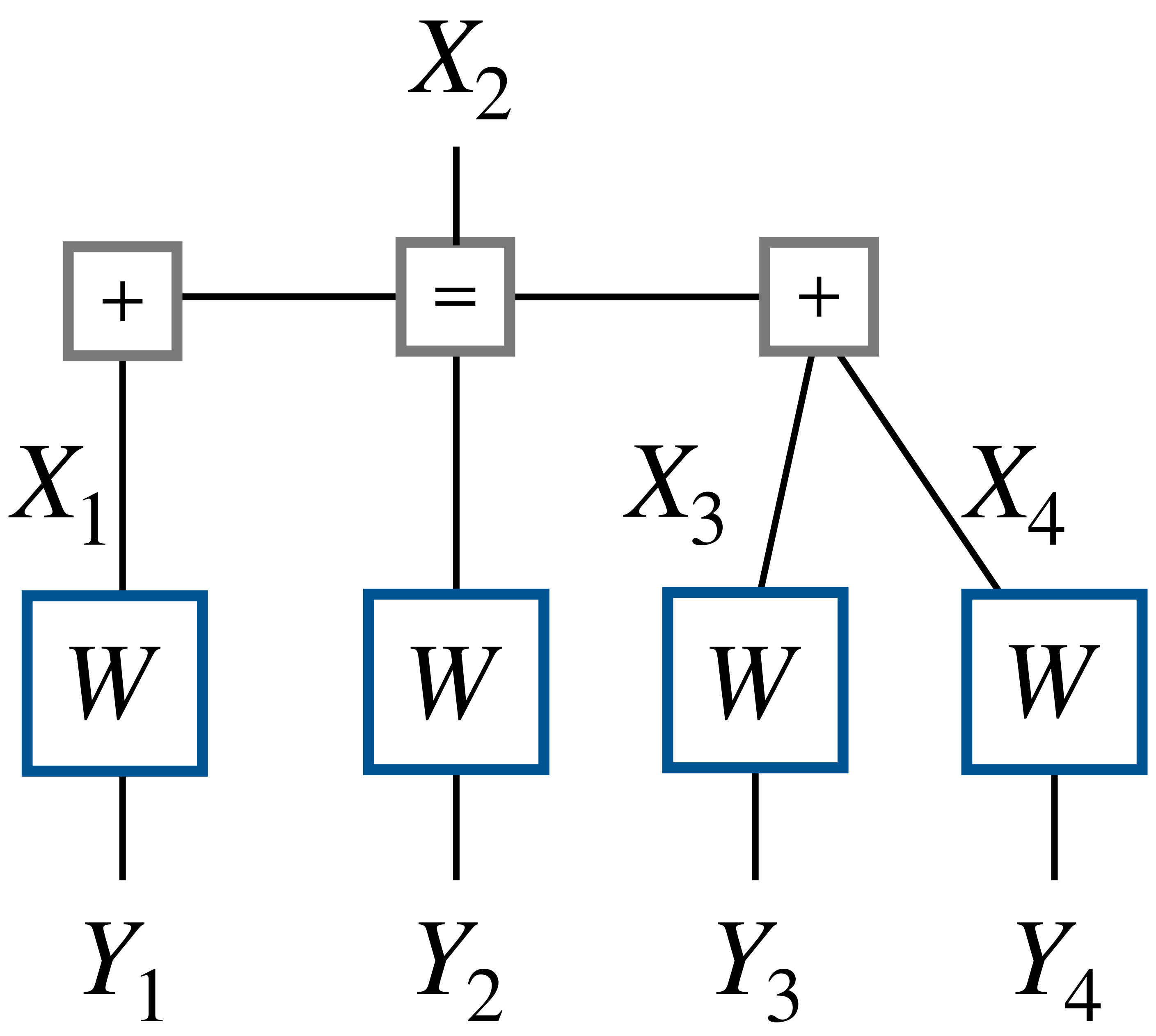
# Belief propagation decoding acting on output bits: BSC




- Associate a bit $b$ and likelihood $\ell = \frac{\delta}{1-\delta}$ to each node
- Traverse tree from leaves to root, generating node $(b,\ell)$ data from children node data.

- Leaf nodes: $b$ is channel output, $\delta$ from $W$
- At + nodes: $b = b_1 \oplus b_2$ and $\ell = \frac{\ell_1 + \ell_2}{1 + \ell_1 \ell_2}$.
- At = nodes: $b = b_1$. Determine parity $k = b_1 \oplus b_2$, set $\ell_2 \leftarrow \ell_2^{(-1)^k}$ and then $\ell = \ell_1 \ell_2$
- At root, generate estimate given the root bit $b$ and $\ell$ .

# Belief propagation decoding acting on output bits: BSC

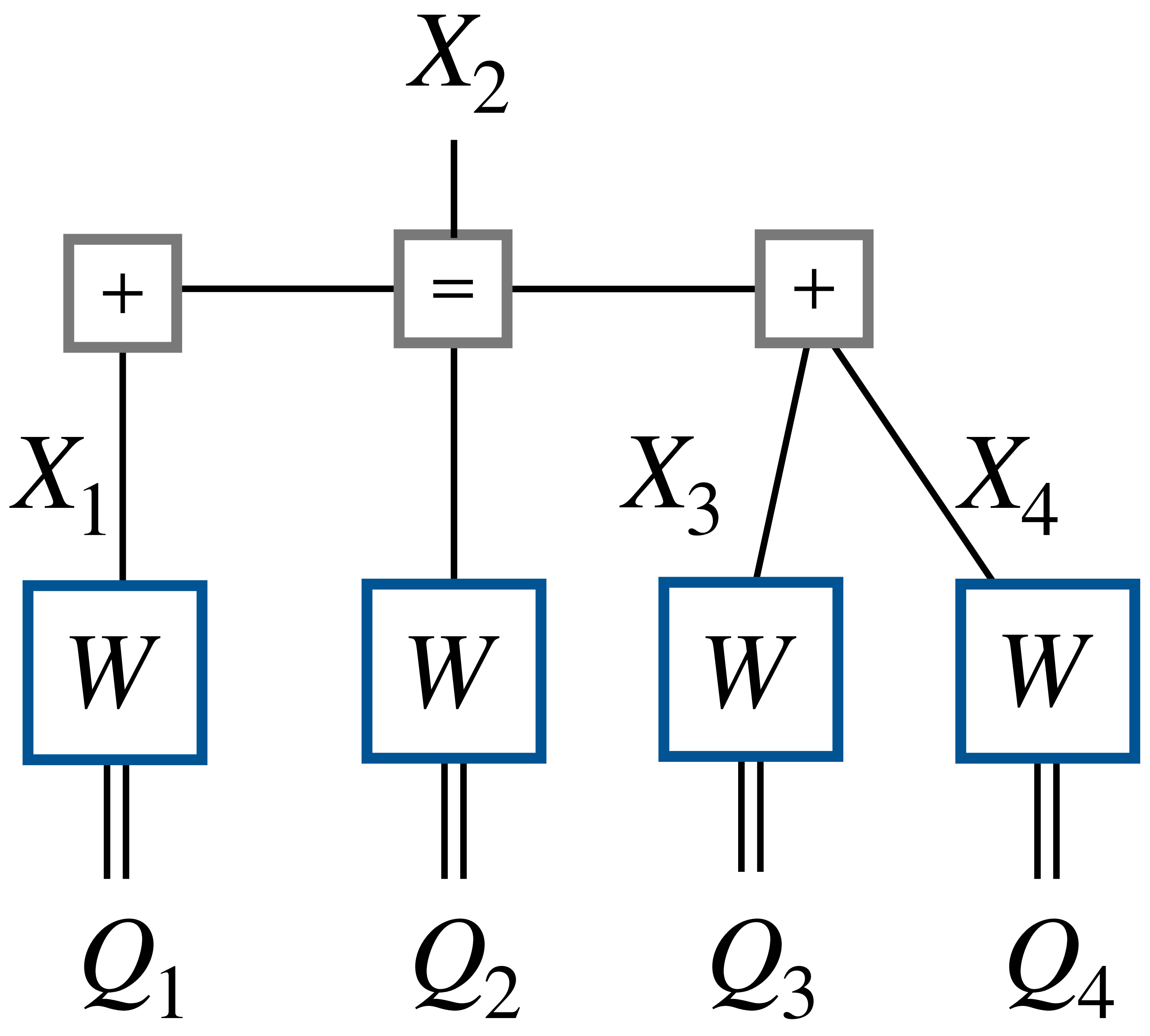



- Message passing: $b$ and $\ell$
- The operations add to the factor graph, but then it simplifies by channel combining rules.
- Results in a single input to a BSC whose output is the root bit $b$, with channel param. $\ell$
- Completely unnecessary, of course: LLR processing in BP includes both $b$ and $\ell$

# Outline

- Variation of classical BP
- BPQM: Passing quantum messages for single bit estimation
- Successive BPQM for entire codewords
- Loopy BPQM
- Summary and open questions

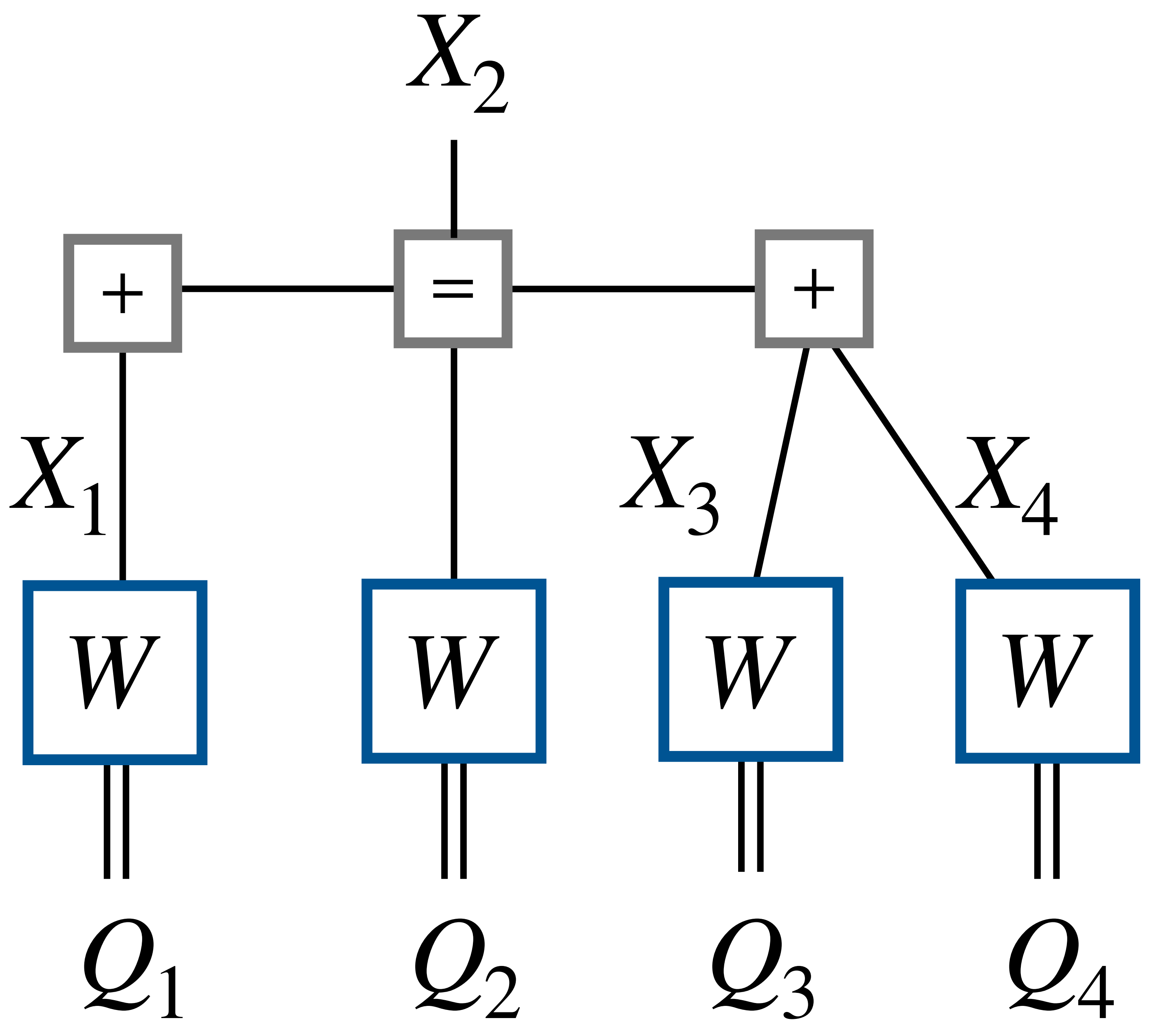
# BP for decoding CQ channel outputs




Pick the simplest possible quantum extension:

Channel with symmetric pure state outputs $|\varphi_x\rangle$

# BP for decoding CQ channel outputs



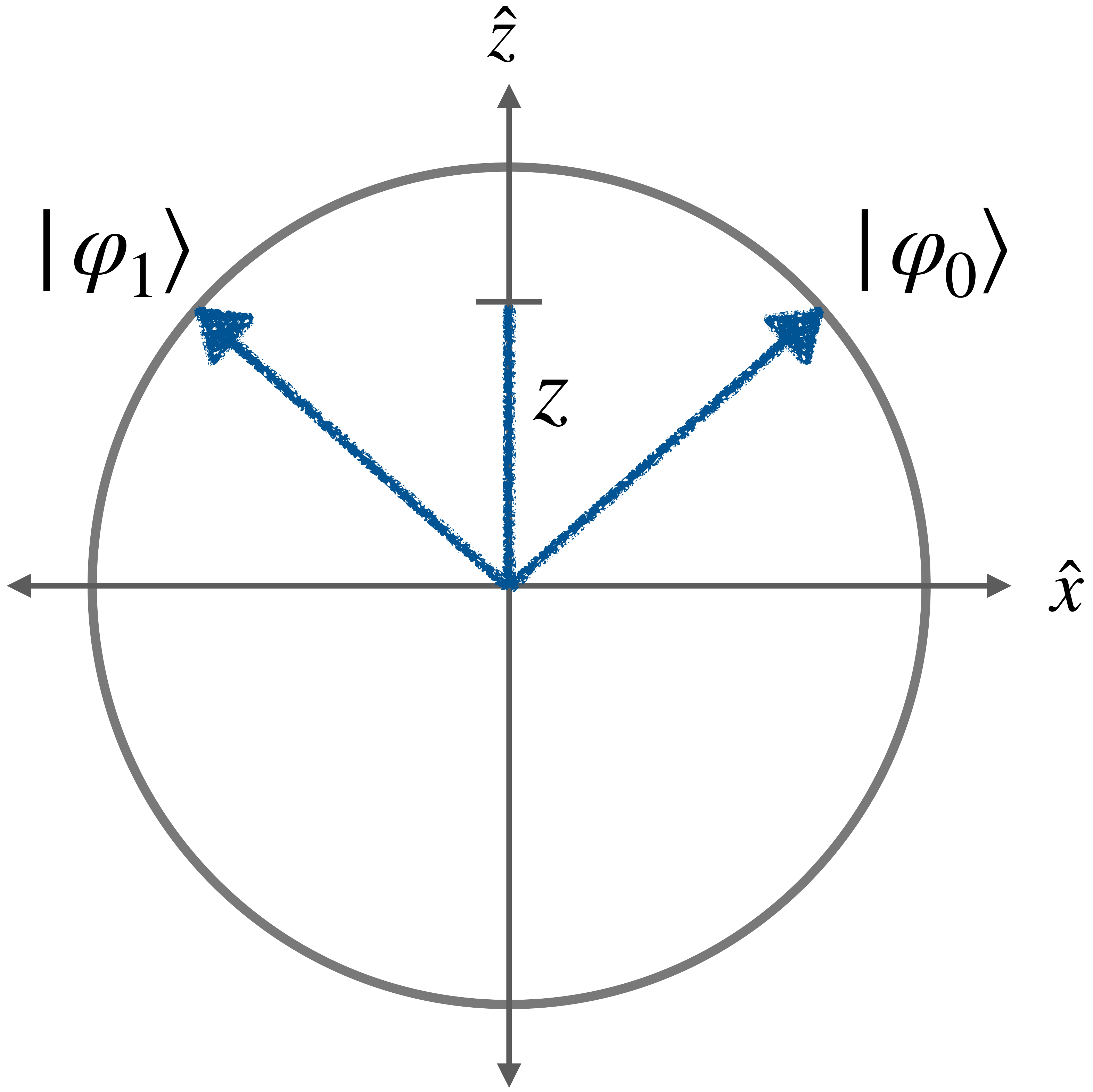

Pick the simplest possible quantum extension:

Channel with symmetric pure state outputs $|\varphi_x\rangle$

Need to construct a measurement to estimate $X_2$ from $Q_1Q_2Q_3Q_4$

Tensor network contraction method not possible!

# CQ channel output description




Bloch sphere

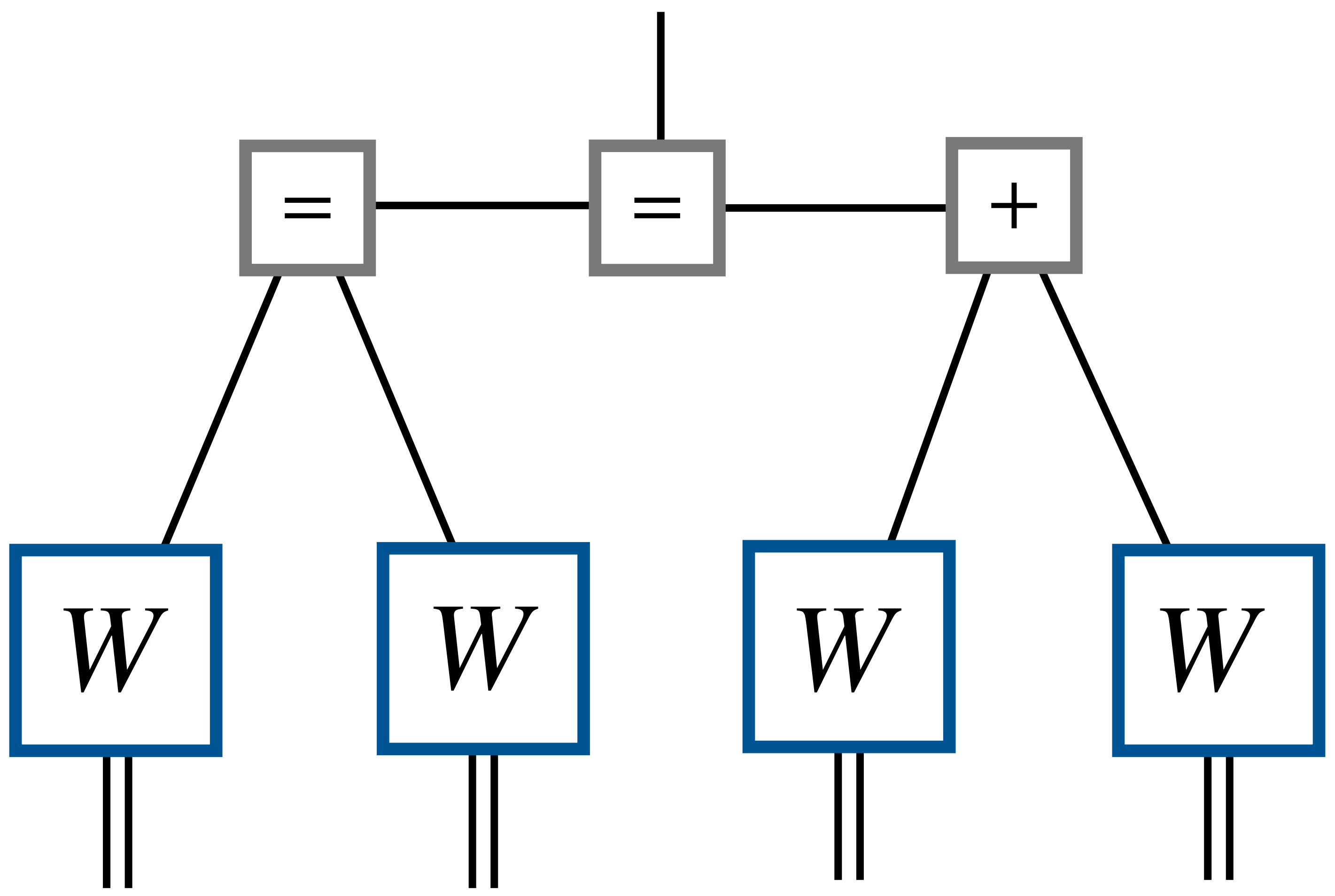
Bloch vector:

$$\hat{n} = z\,\hat{z} + (-1)^x \sqrt{1 - z^2}\ \hat{x}$$

Like $\ell$ from BSC:

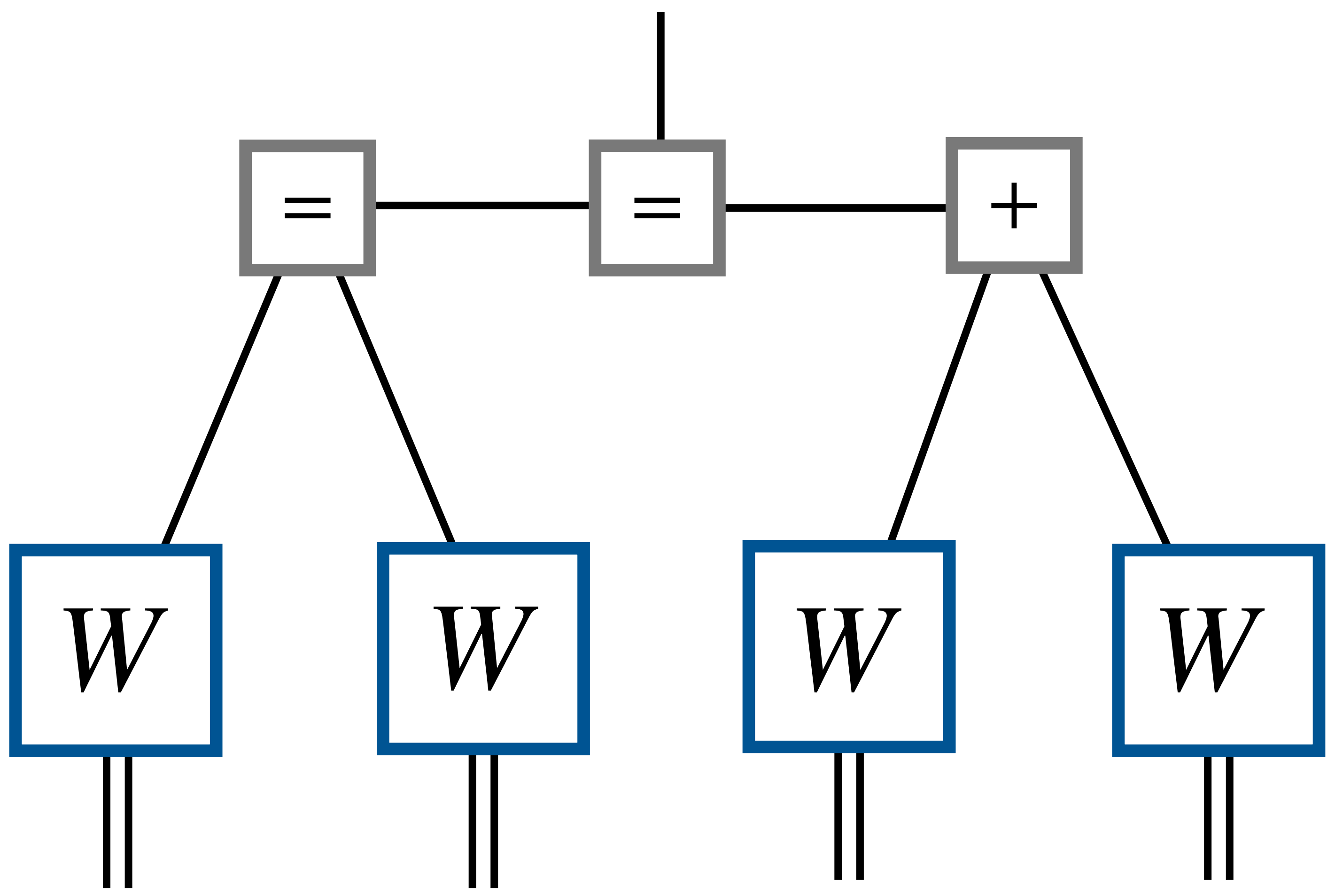
Small value indicates a reliable channel

# Quantum message passing algorithm: BPQM



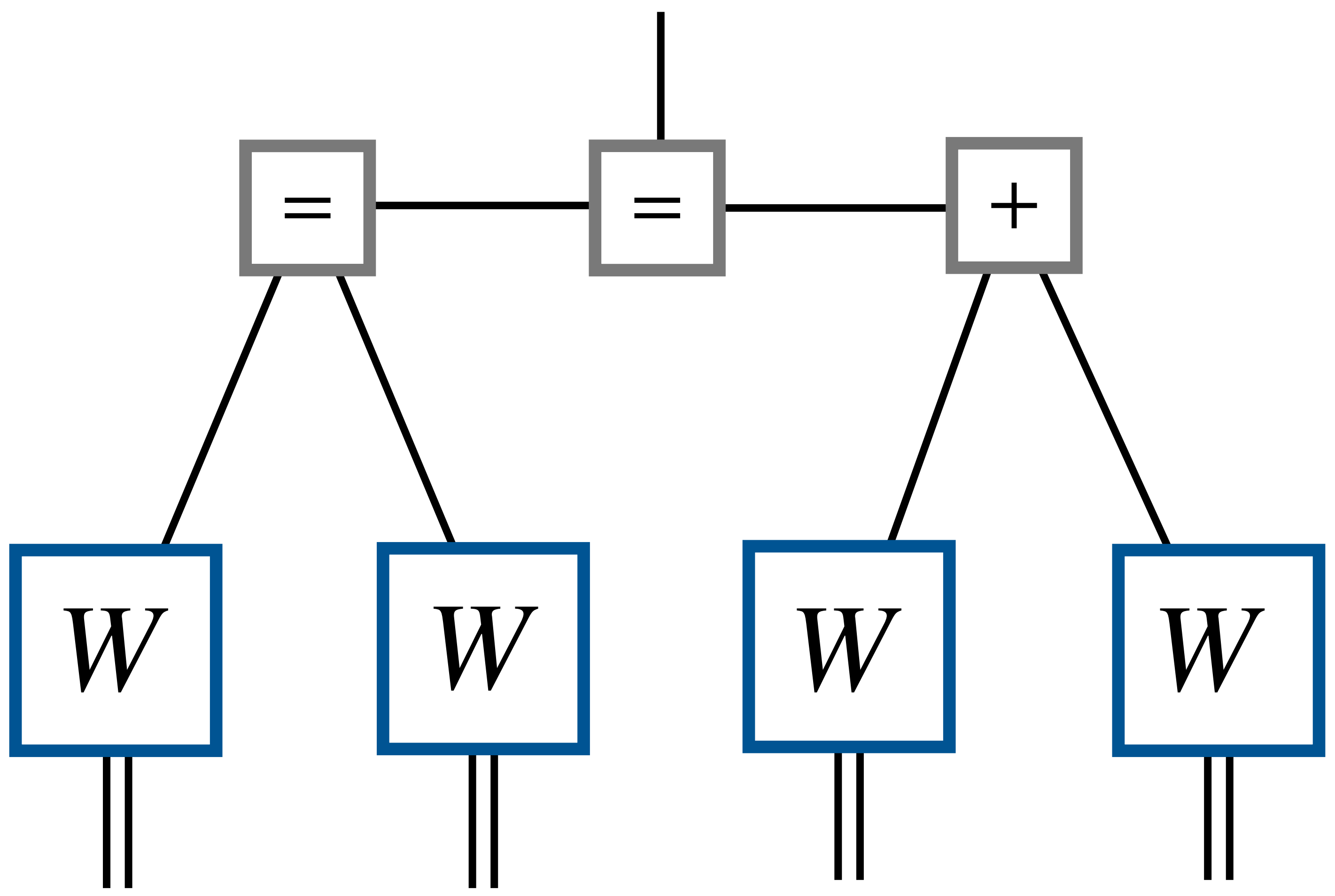

- Associate a qubit and $z$ parameter to each node
- Traverse the tree from $W$ leaves to root

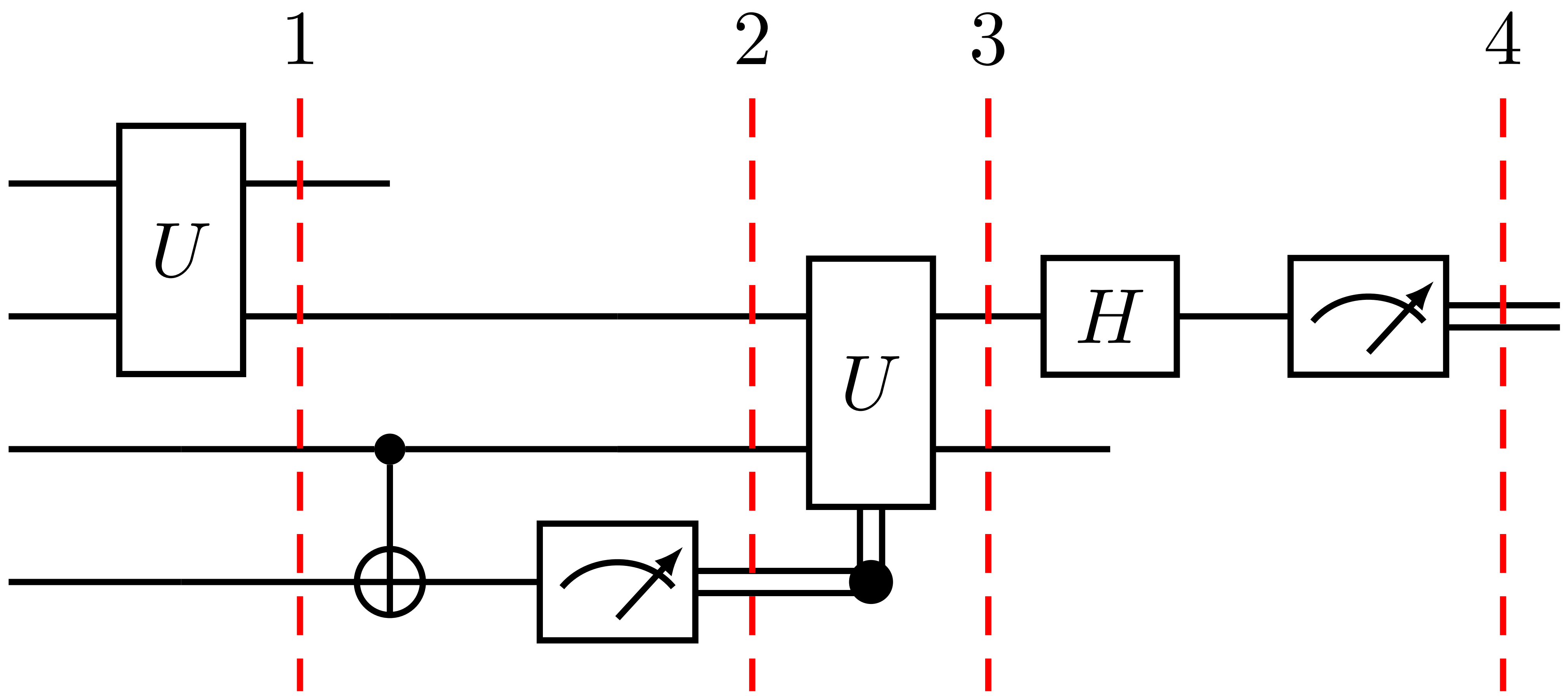
# Quantum message passing algorithm: BPQM




- Associate a qubit and $z$ parameter to each node
- Traverse the tree from $W$ leaves to root
- At = nodes: Apply unitary $U(z_1, z_2)$ and keep just 1st qubit. Set $z = z_1 z_2$.
- At + nodes: Apply CNOT, measure 2nd qubit $\to k$. Reset $z_2 \leftarrow (-1)^k z_2$ and set param to $\frac{z_1 + z_2}{1 + z_1 z_2}$.
- Measure root qubit in $\hat{x}$ basis.

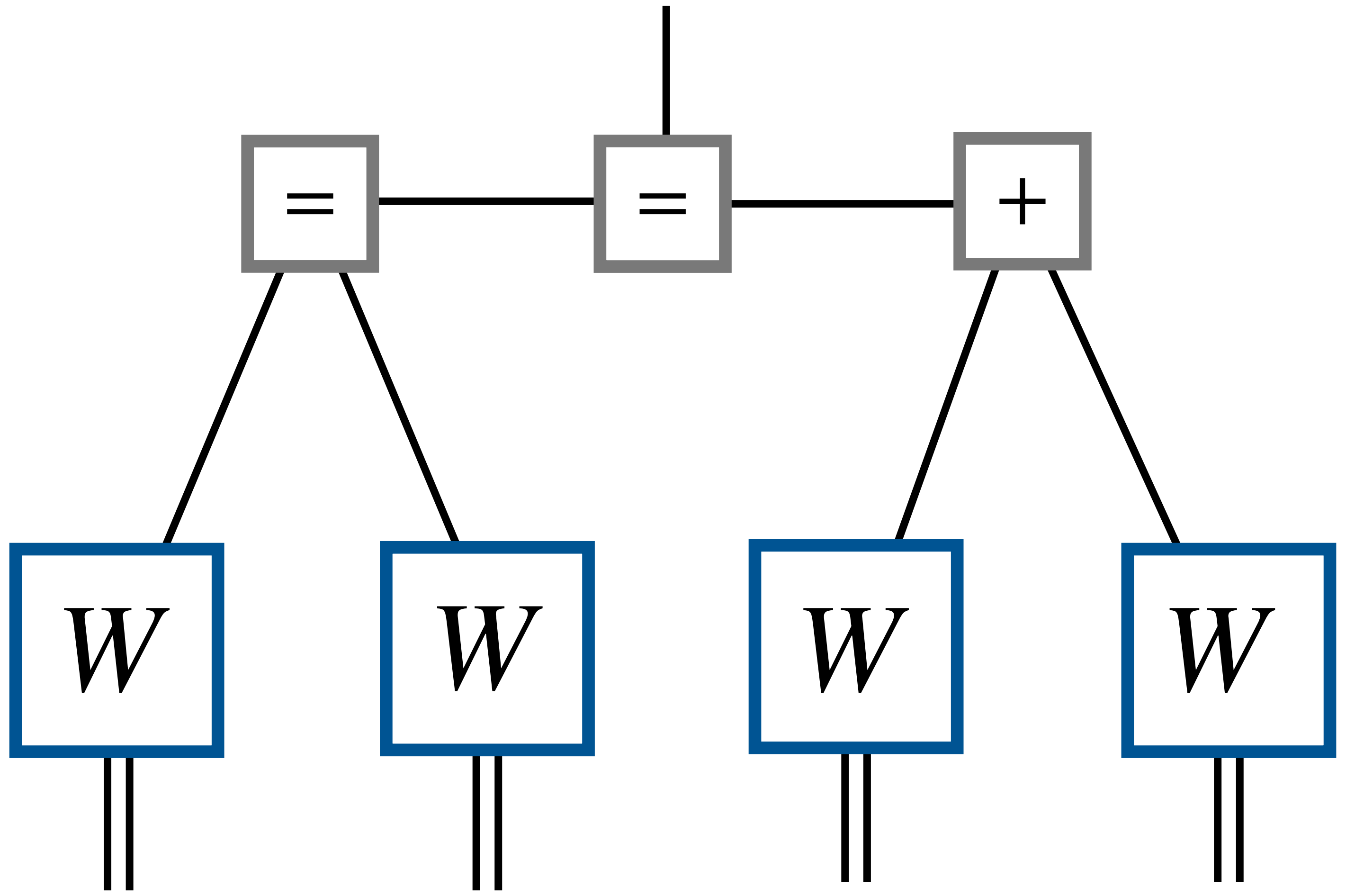
# Quantum message passing algorithm: BPQM




- =: Apply unitary $U(z_1, z_2)$, discard 2nd qubit. Set param to $z_1 z_2$.
- +: Apply CNOT, measure 2nd qubit $\rightarrow k$. Discard 2nd qubit. Reset $z_4 \leftarrow (-1)^k z_4$ and set param to $\dfrac{z_3 + z_4}{1 + z_3 z_4}$.
- Measure last qubit in $\hat{x}$ basis.

# Quantum message passing algorithm: BPQM

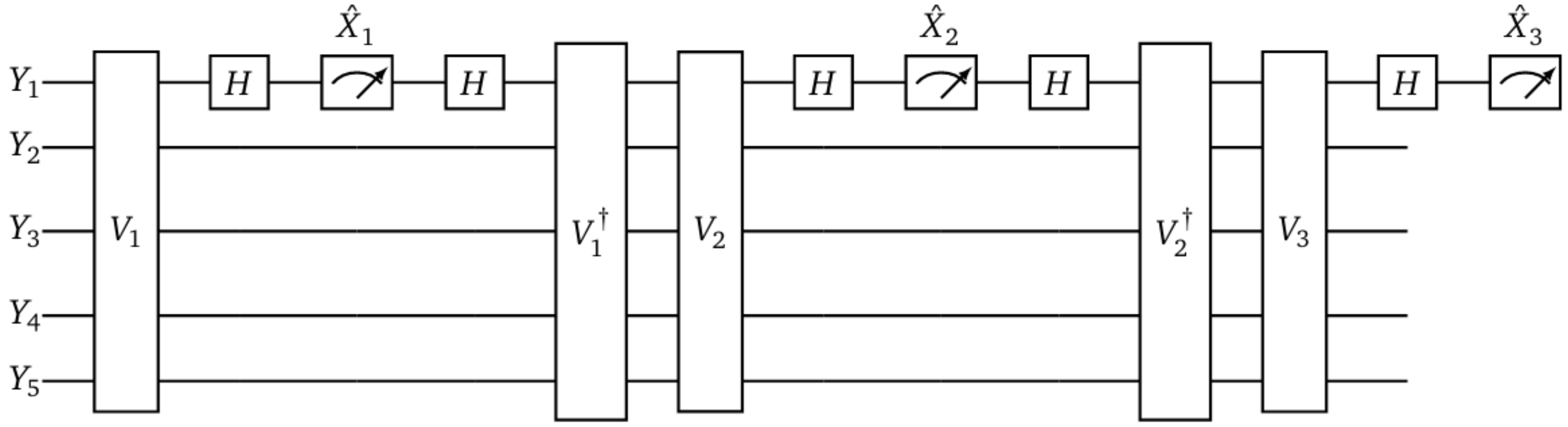



- Implements optimal bitwise measurement: operations are actually reversible
- Factor graph simplifies as before, to a single classical input and pure state output.
- Messages passed are one part classical ($z$), one part quantum (qubit)

# Outline

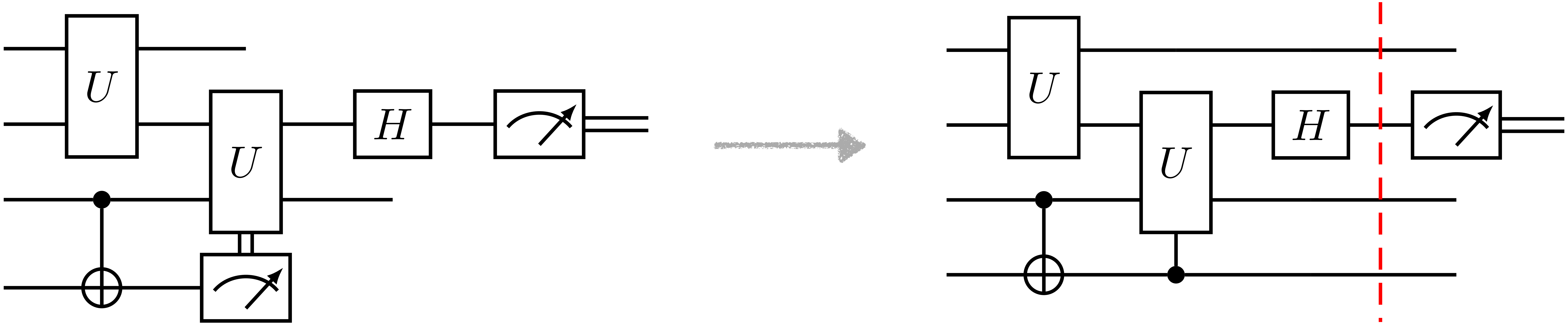
- Variation of classical BP
- BPQM: Passing quantum messages for single bit estimation
- Successive BPQM for entire codewords
- Loopy BPQM
- Summary and open questions

# Successive BPQM for decoding entire codeword



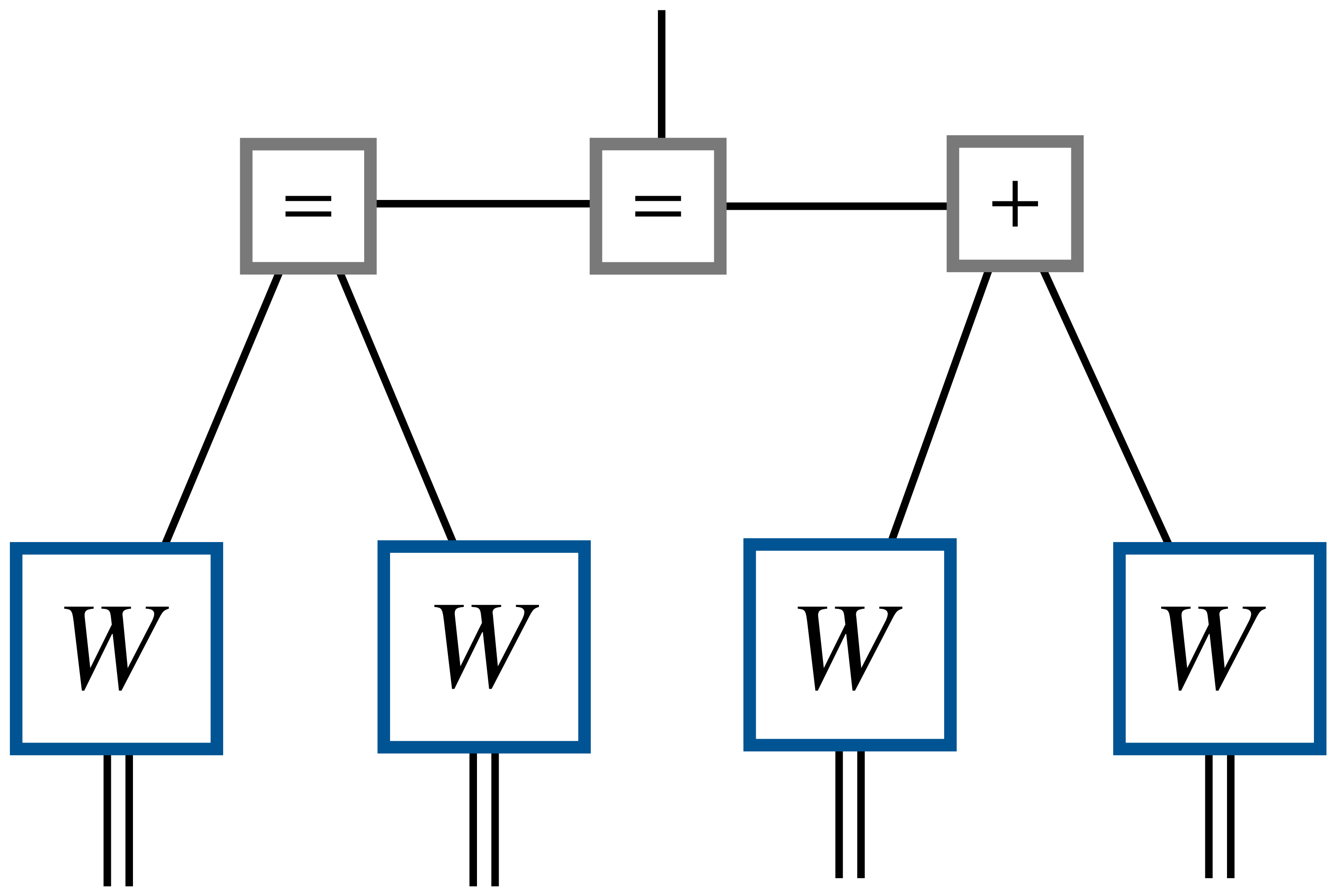
# Successive BPQM for decoding entire codeword

- Problem: Intermediate measurements.
  Solution: Perform BPQM coherently ("deferred measurement").
  Rewind the circuit after measuring the output qubit.



# Successive BPQM for decoding entire codeword

- Problem: Intermediate measurements.
  Solution: Perform BPQM coherently.
  Rewind the circuit after decoding each bit.
- Problem: Exponential overhead from + controls.
  Solution: Quantize $z$ register. Uncompute after use.
- Problem: Need infinite dimensions for $z$ register.
  Solution: Discretize to finite precision.
  For target error $\varepsilon$, register size only $O(\log 1/\varepsilon)$.
- All messages passed are now quantum!



# Outline

- Variation of classical BP
- BPQM: Passing quantum messages for single bit estimation
- Successive BPQM for entire codewords
- Loopy BPQM
- Summary and open questions

# Loopy BPQM: Setup

Unroll Tanner graph to computational graph

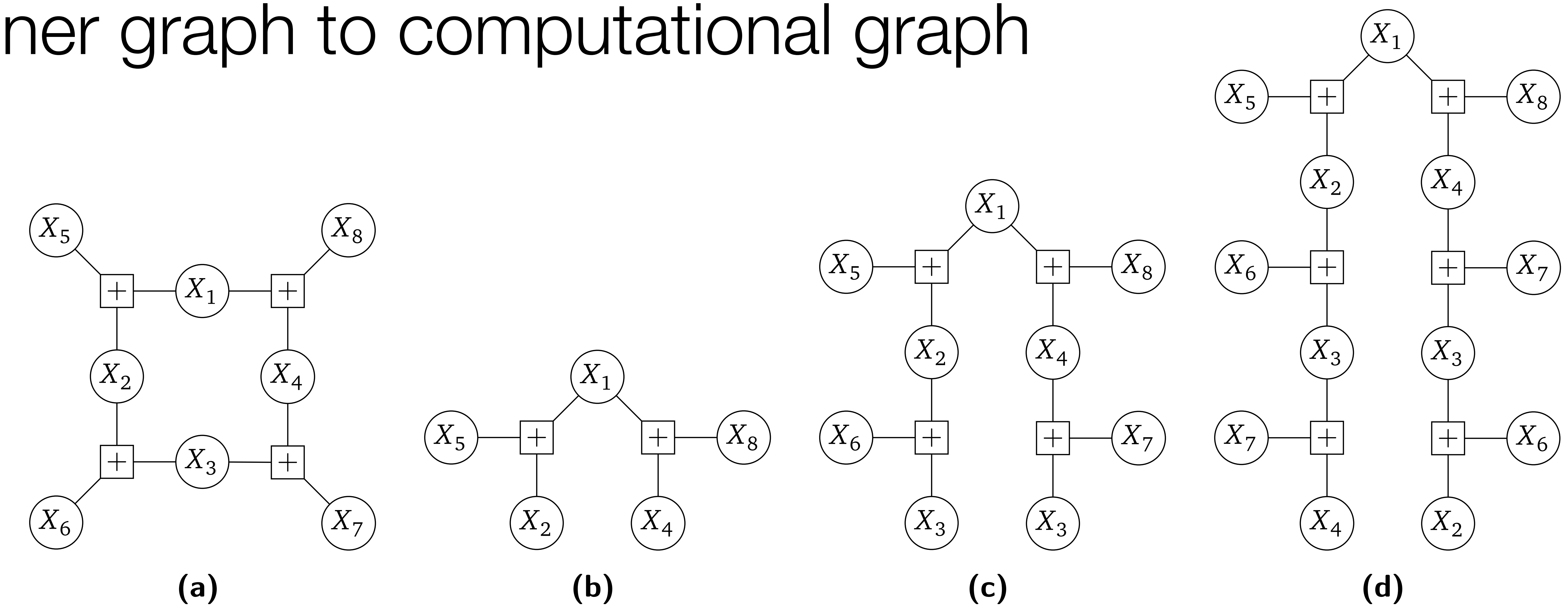



**Figure 16:** Tanner graph of the (8,4) code $\mathcal{C}$ and associated $X_1$ computation trees for $h = 1, 2, 3$.

Run BPQM:

Initialize leaves with approximately cloned qubits and appropriate $z$

# Loopy BPQM: Performance

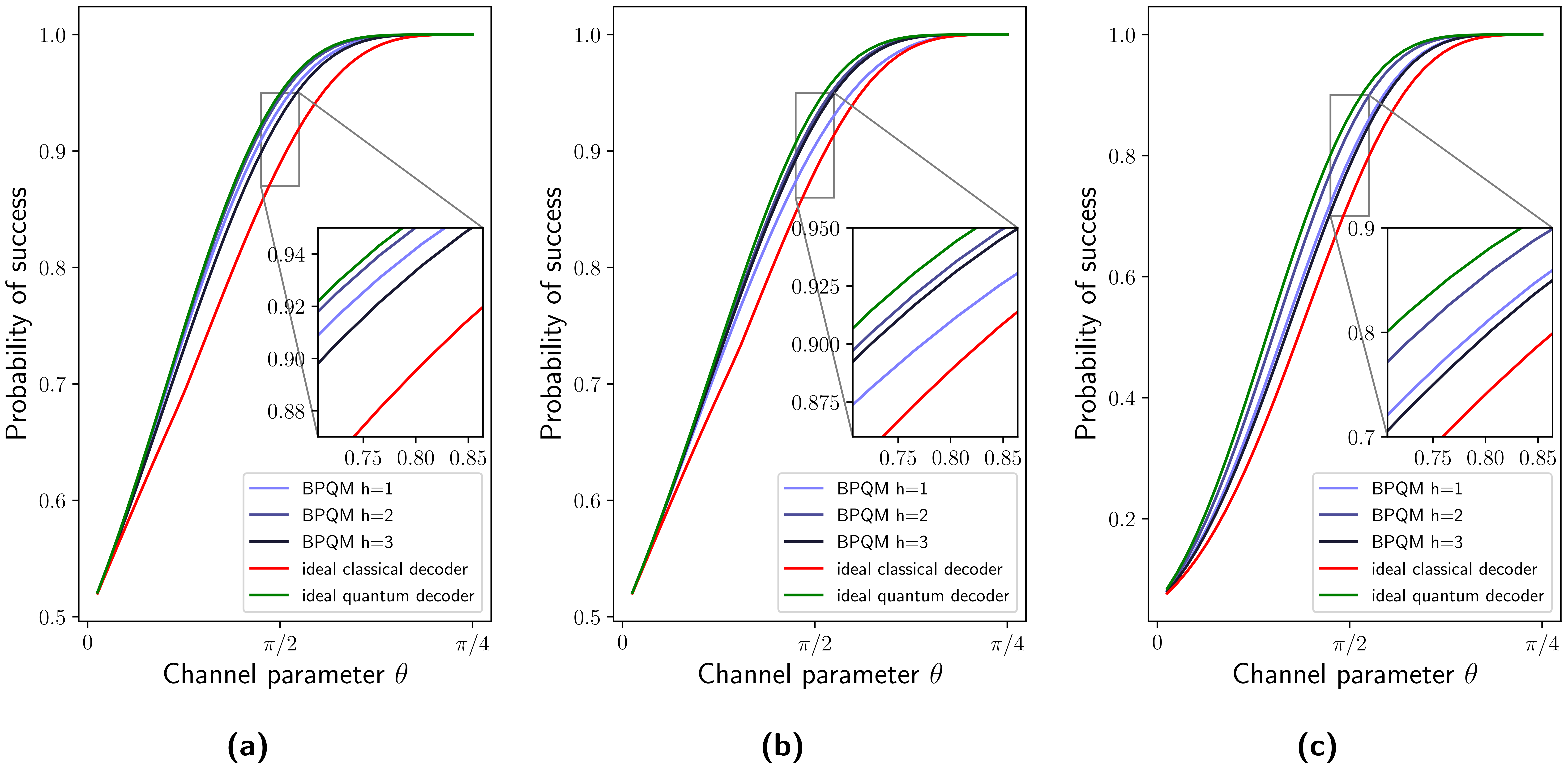



**Figure 17:** Numerical results from decoding $X_1$, $X_5$ and the complete codeword in the 8-bit code.

# Outline

- Variation of classical BP
- BPQM: Passing quantum messages for single bit estimation
- Successive BPQM for entire codewords
- Loopy BPQM
- Summary and open questions

# Summary & Open questions

- BPQM: efficient bitwise-optimal quantum message passing decoder
- Also blockwise optimal!
- Applications to capacity-achieving polar codes:
  - BPSK on pure loss Bosonic channel for transmitting classical information
  - CSS codes for amplitude damping channel for transmitting quantum information

- LDPC codes?
- Codes with loops?
- BPQM for mixed state output channels?