Iterative linear-programming-based route optimization for cooperative wireless networks

Jonathan Yedidia
Mitsubishi Electric Research Labs (MERL)

Stark Draper
UW Madison, MERL

Lingjia Liu
UT Austin, MERL

Andy Molisch
MERL

Outline

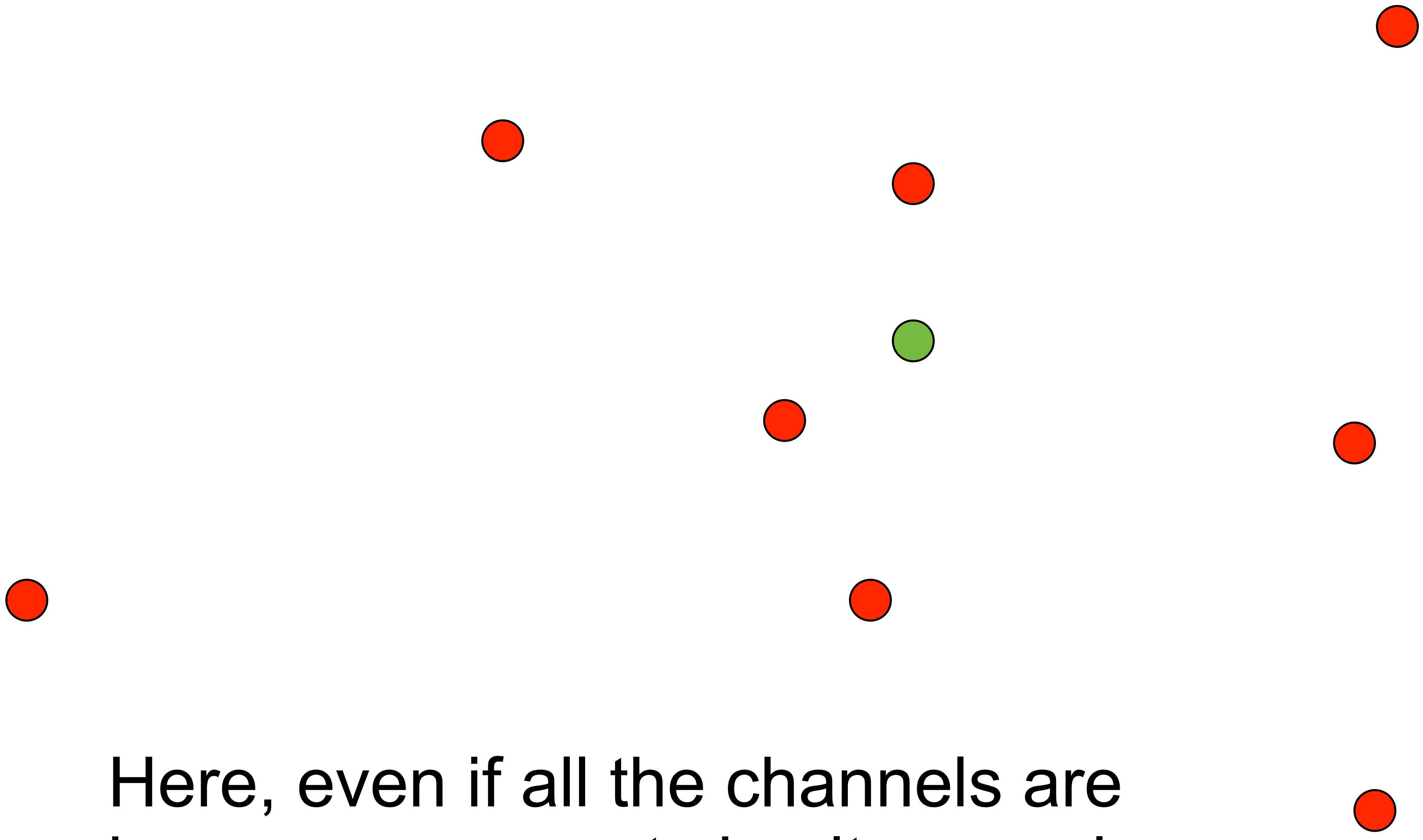
- Motivations and Fountain Code Background
- Routing Problem Statement
- Resource Allocation Linear Program
- Decoding Order Revision
- Sample Results

Point-to-point Communication

Source → Encoder → Channel → Decoder → Destination

We assume that the channel is known so that the correct strength code can be chosen. But what if the channel is unknown? Too strong a code is wasteful, too weak a code will fail.

Wireless Broadcast Scenario



Here, even if all the channels are known, we cannot simultaneously efficiently and reliably broadcast to everyone using standard codes.

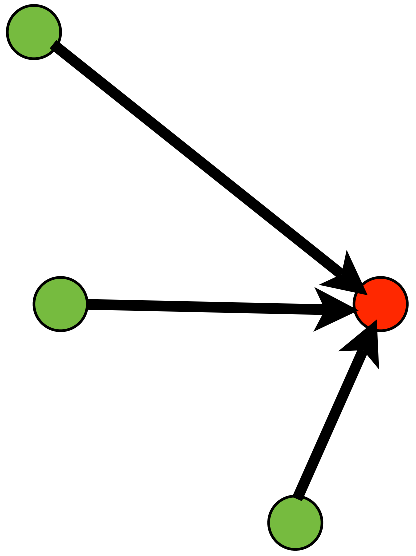
Fountain Codes

- Rate-less: from a set of information bits, produce an infinite stream of transmitted bits. A sufficiently large subset of received bits (how large depends on the channel) allows recovery of the input.

Fountain Codes

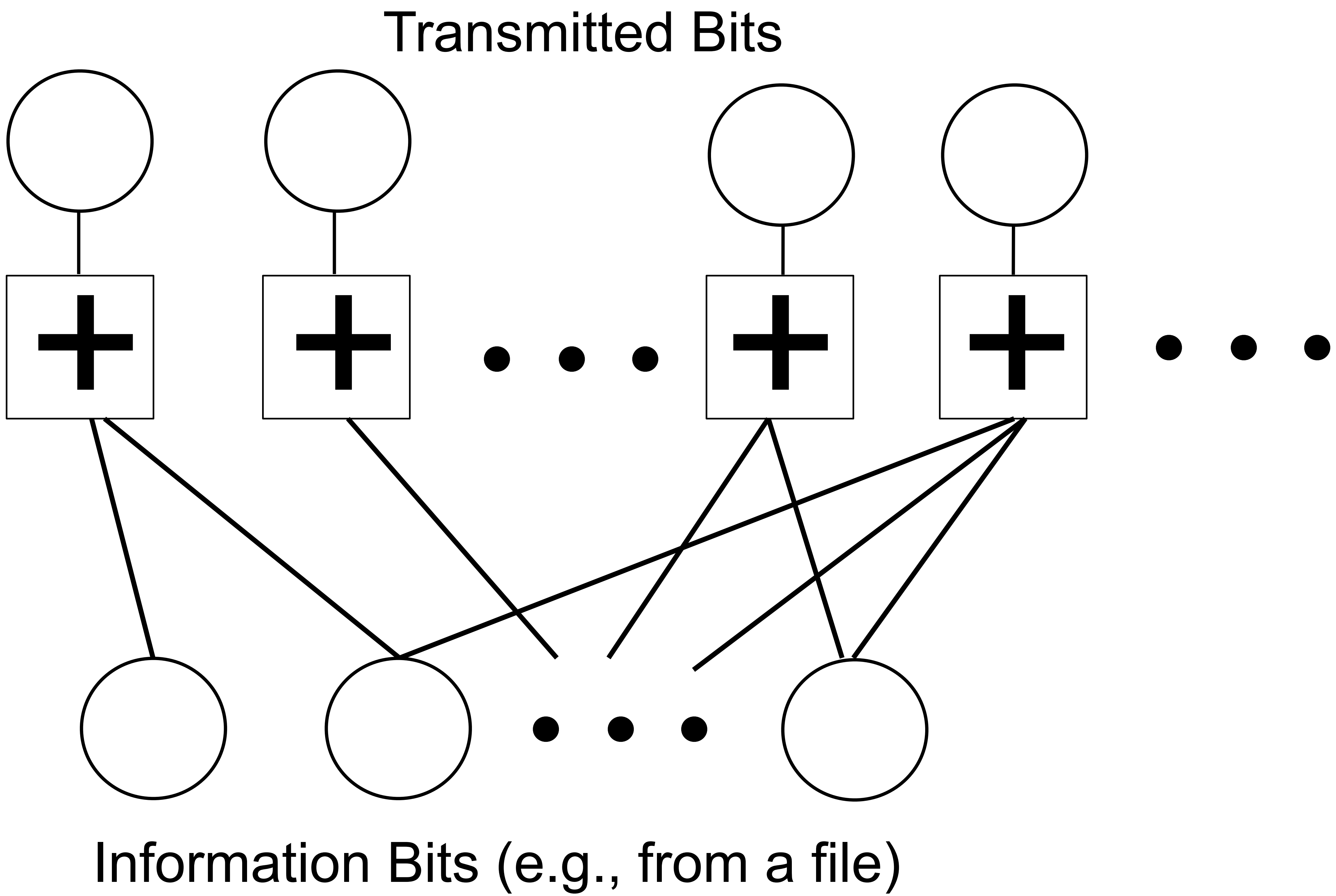
- Rate-less: from a set of information bits, produce an infinite stream of transmitted bits. A sufficiently large subset of received bits (how large depends on the channel) allows recovery of the input.

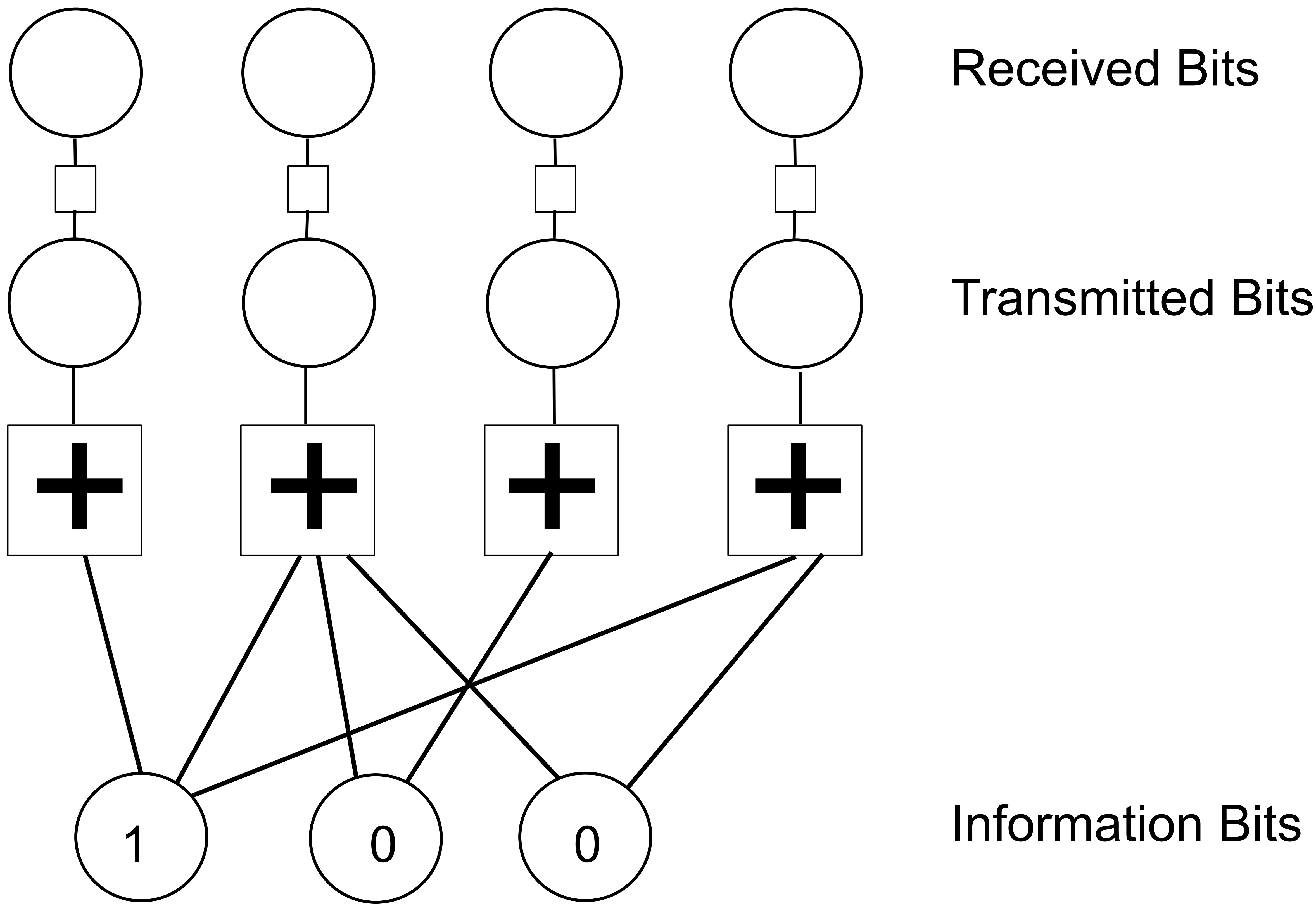
- Mutual-information combining: A receiver can collect bits from two or more independent transmitters.

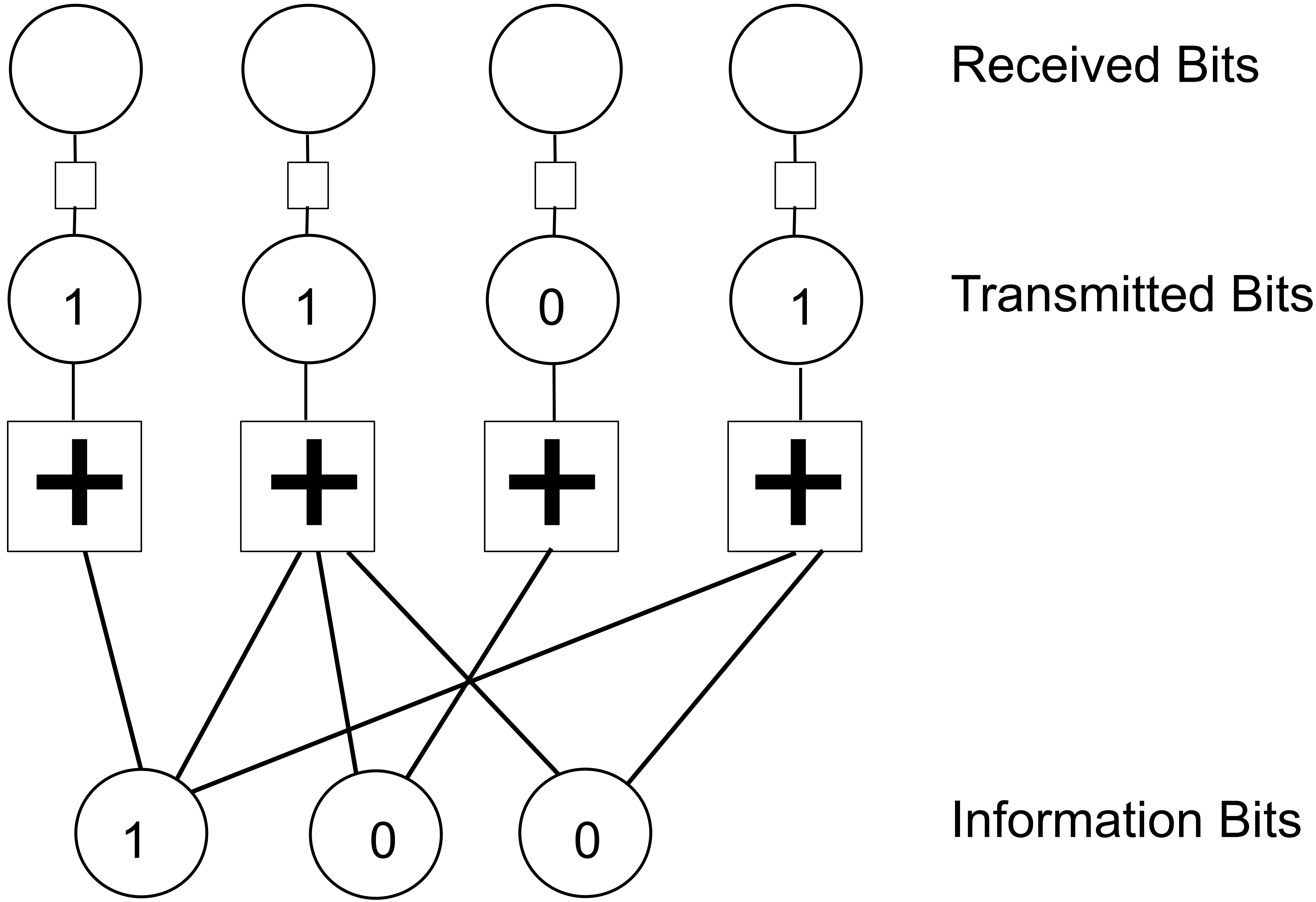


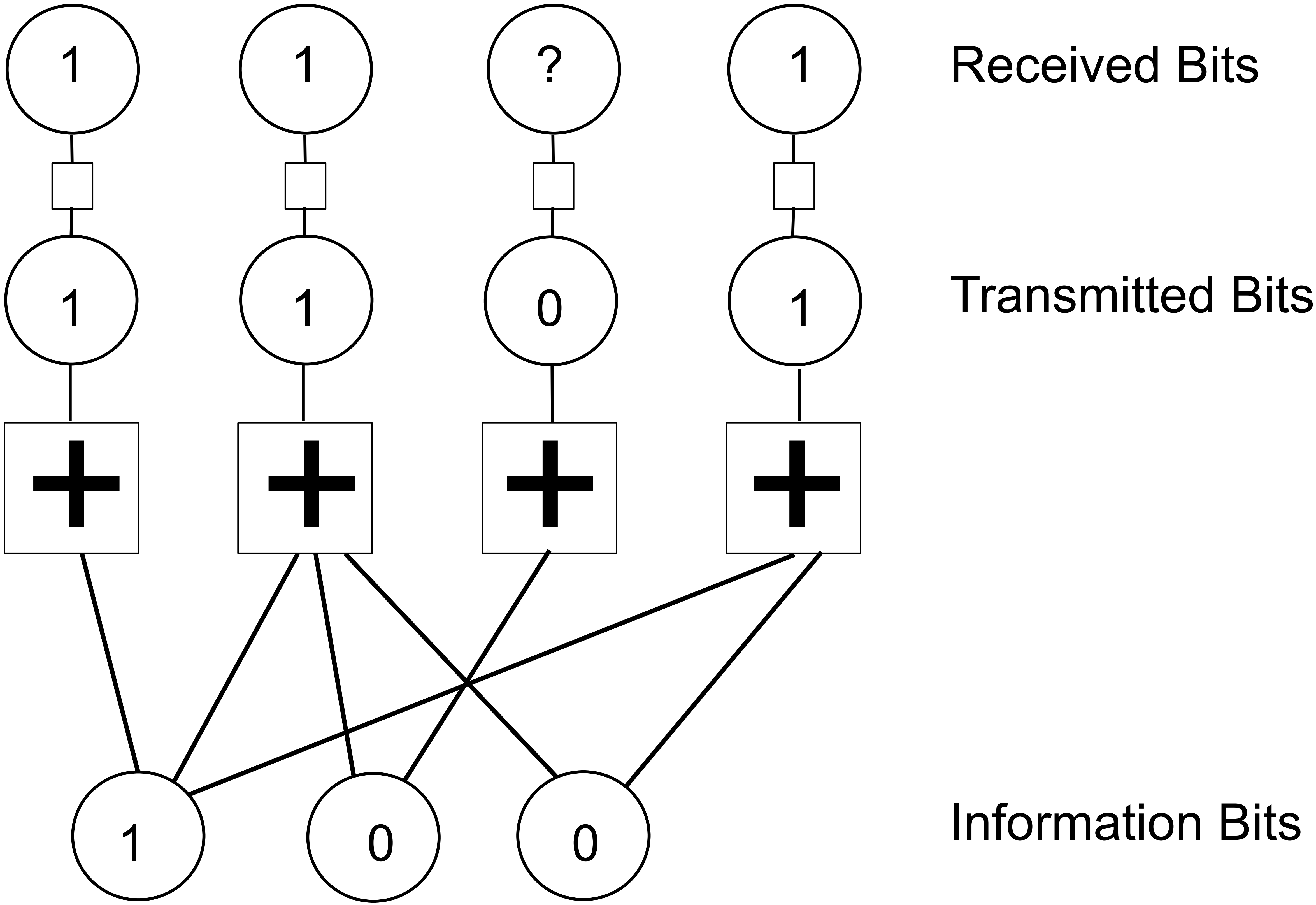
LT Fountain Codes

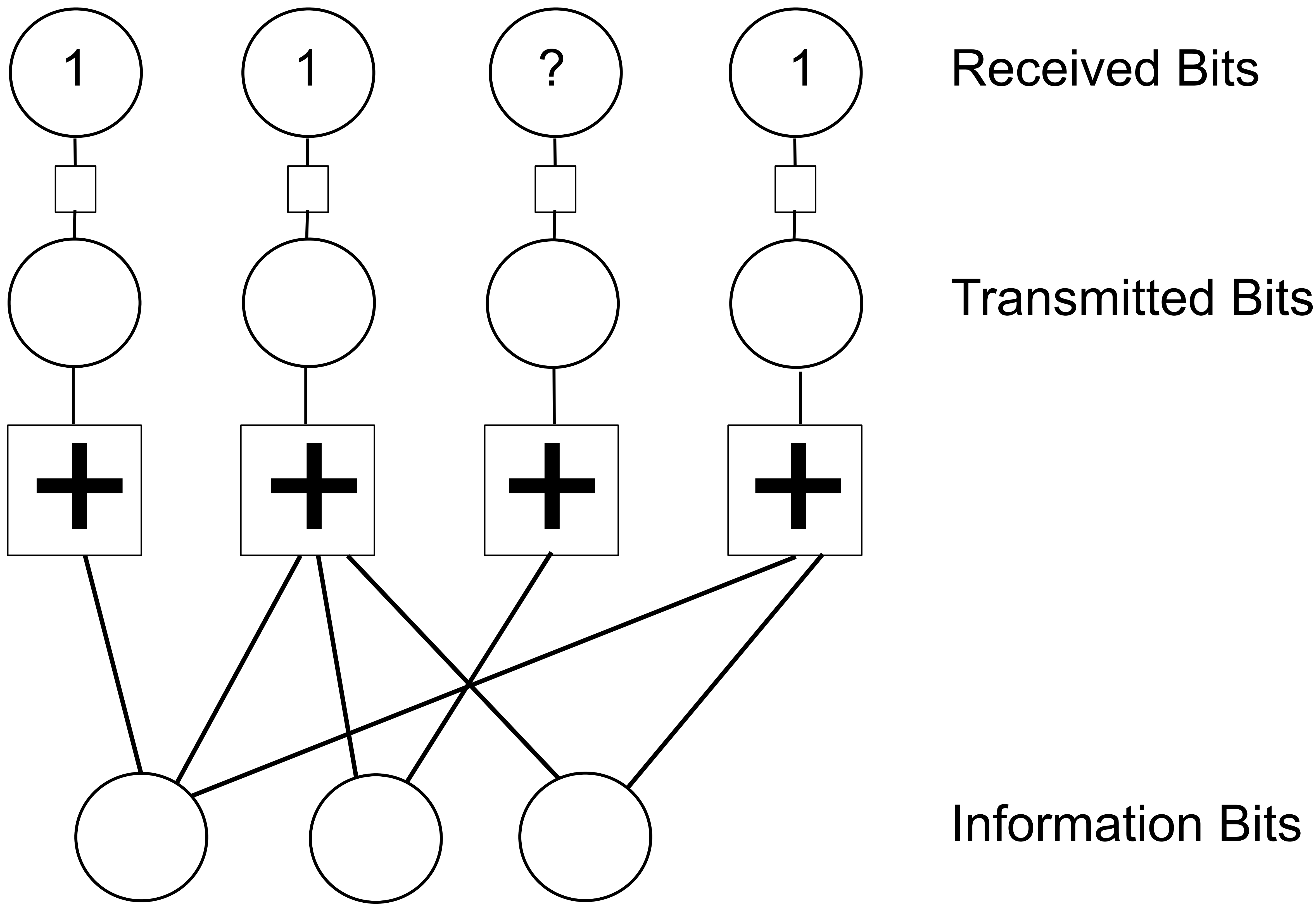
(Luby, 2002)

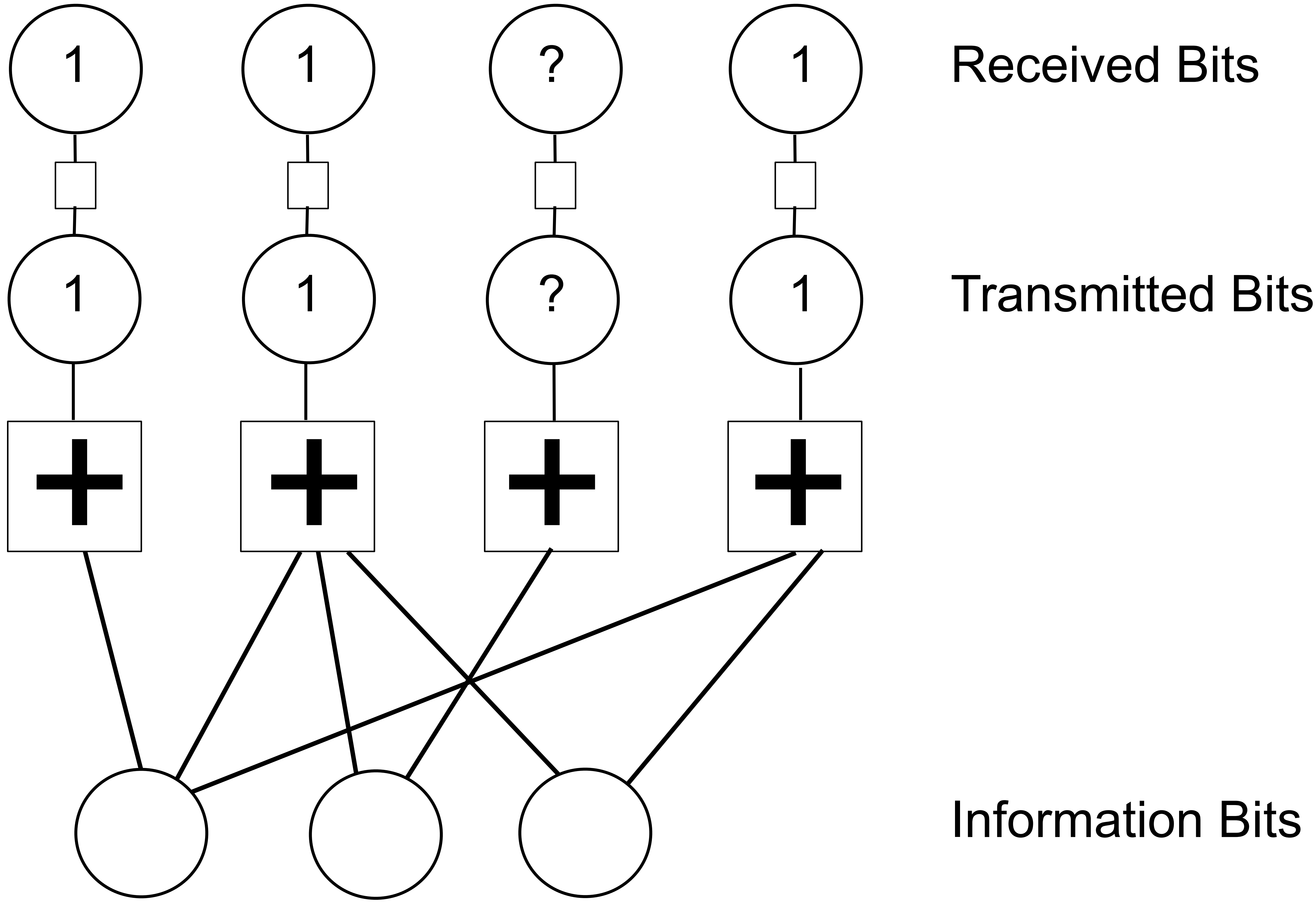


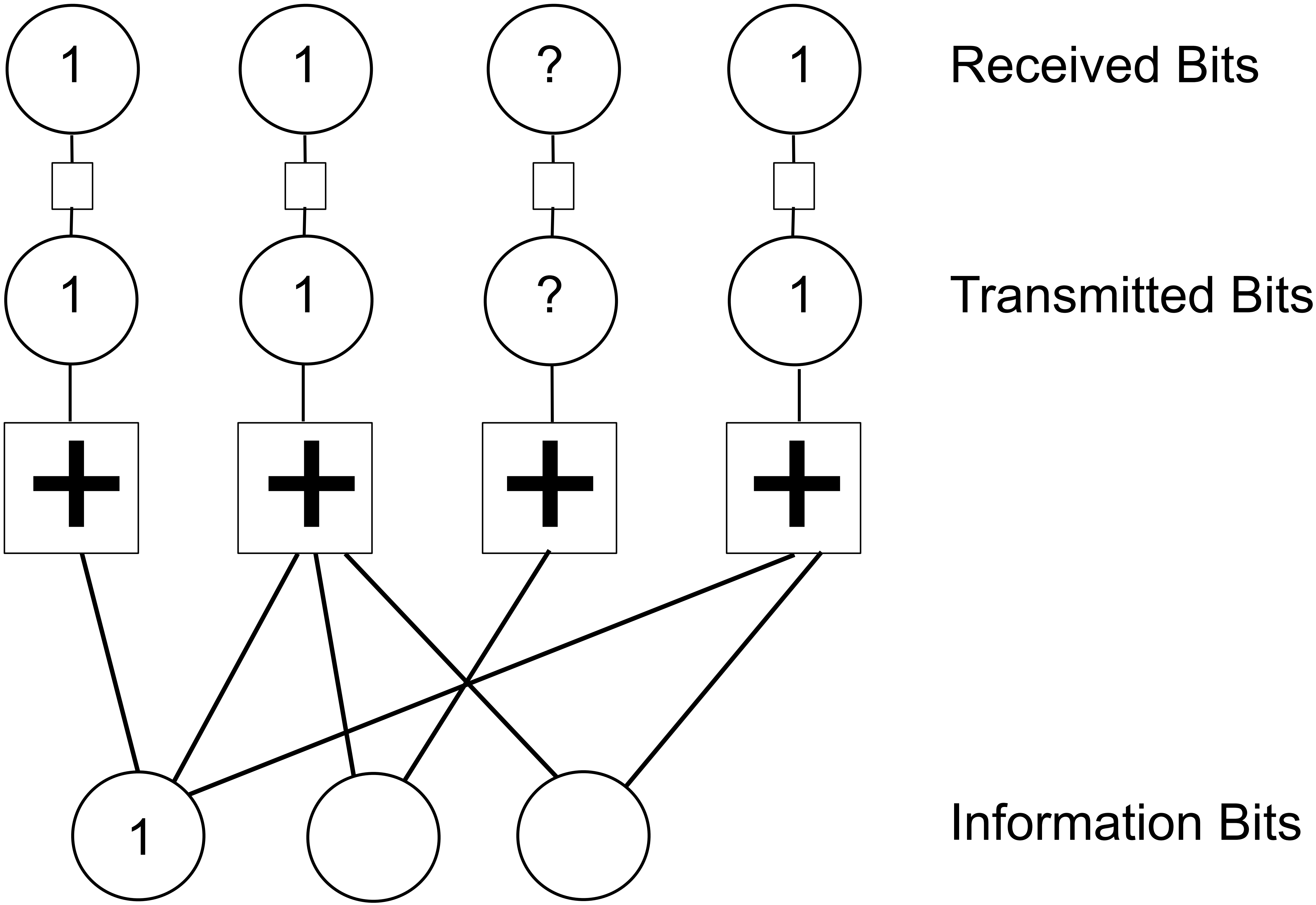

Received Bits
Transmitted Bits
+
+
+
+
1
0
0
Information Bits

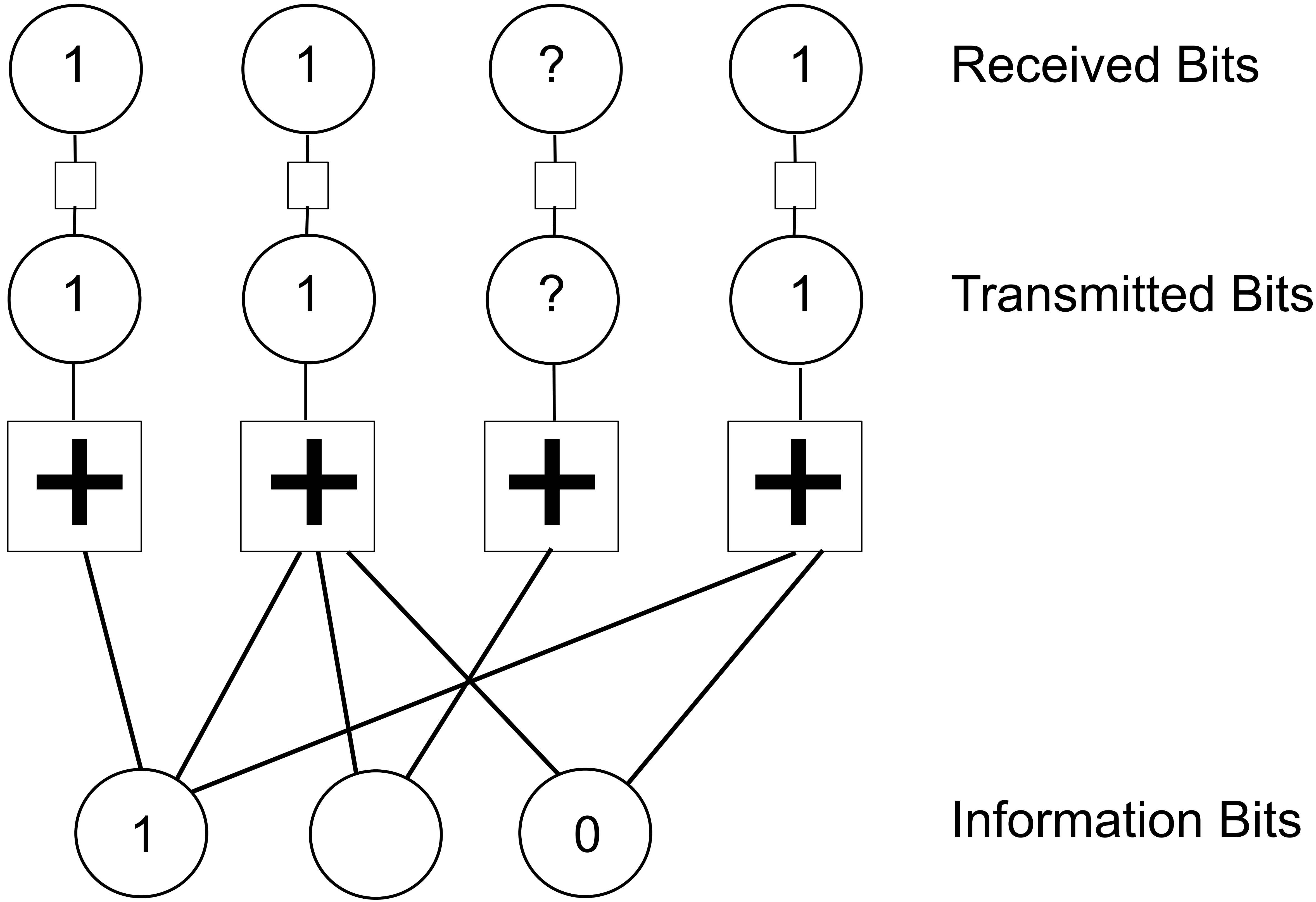

Received Bits
1
1
0
1
Transmitted Bits
+
+
+
+
1
0
0
Information Bits

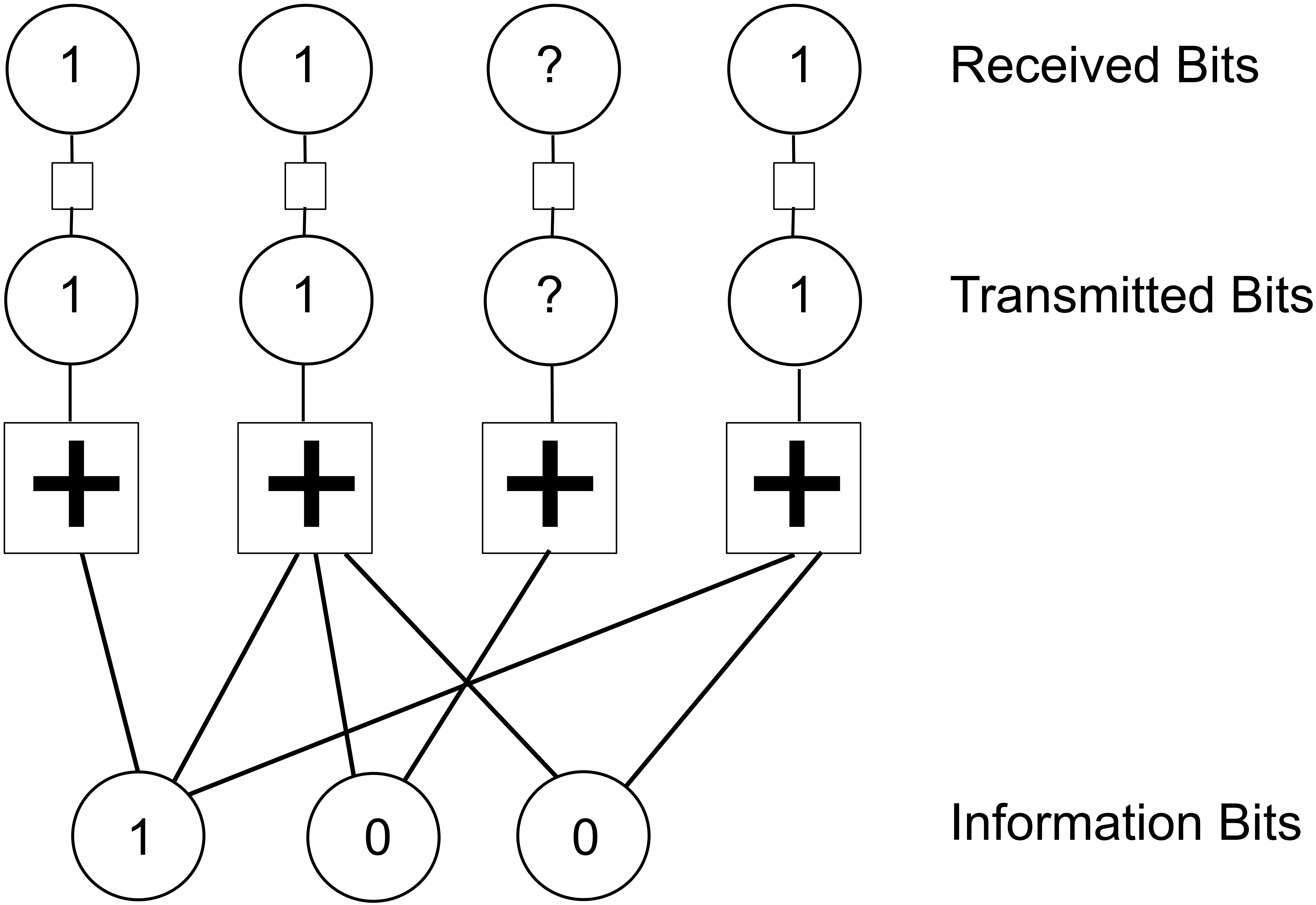

1
1
?
1
Received Bits
1
1
0
1
Transmitted Bits
+
+
+
+
1
0
0
Information Bits


1
1
?
1
Received Bits
Transmitted Bits
Information Bits


1
1
?
1
Received Bits
1
1
?
1
Transmitted Bits
Information Bits


1
1
?
1
Received Bits
1
1
?
1
Transmitted Bits
1
Information Bits


1
1
?
1
Received Bits
1
1
?
1
Transmitted Bits
+
+
+
+
1
0
Information Bits


1
1
?
1
Received Bits
1
1
?
1
Transmitted Bits
1
0
0
Information Bits

Improvements on LT Codes

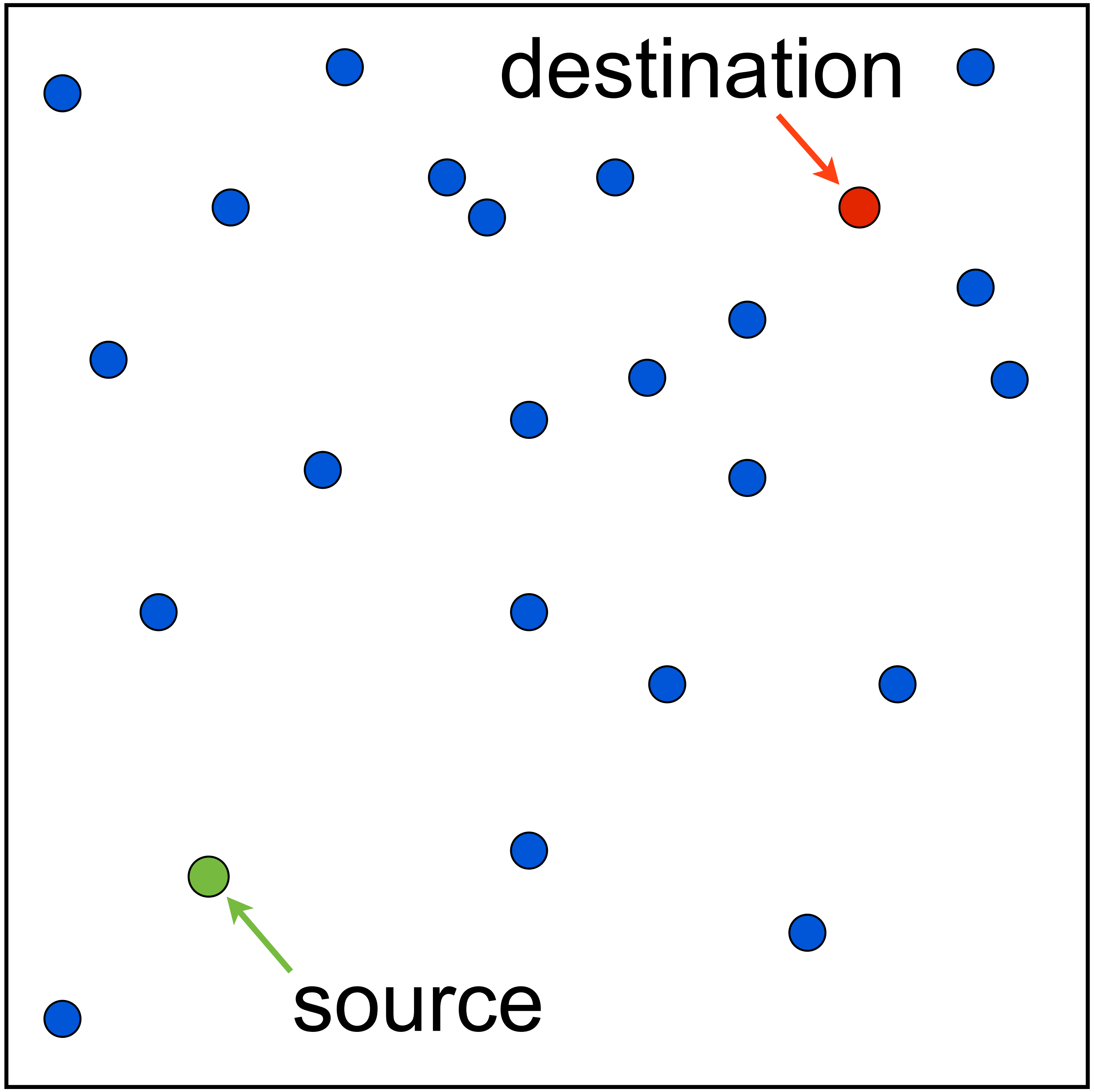
- Raptor Codes (Shokrollahi 2003): Use a pre-code which ensures that “missed bits” can be cleaned up.

- Codes for non-erasure channels (Palanki & Yedidia 2003, Estami Molkaraie and Shokrollahi 2003).

Outline

- Motivations and Fountain Code Background
- Routing Problem Statement
- Resource Allocation Linear Program
- Decoding Order Revision
- Sample Results

Resource allocation in large networks



Objective:

- minimize delay

Constraints:

- energy
- band-width

Focus: resource allocation

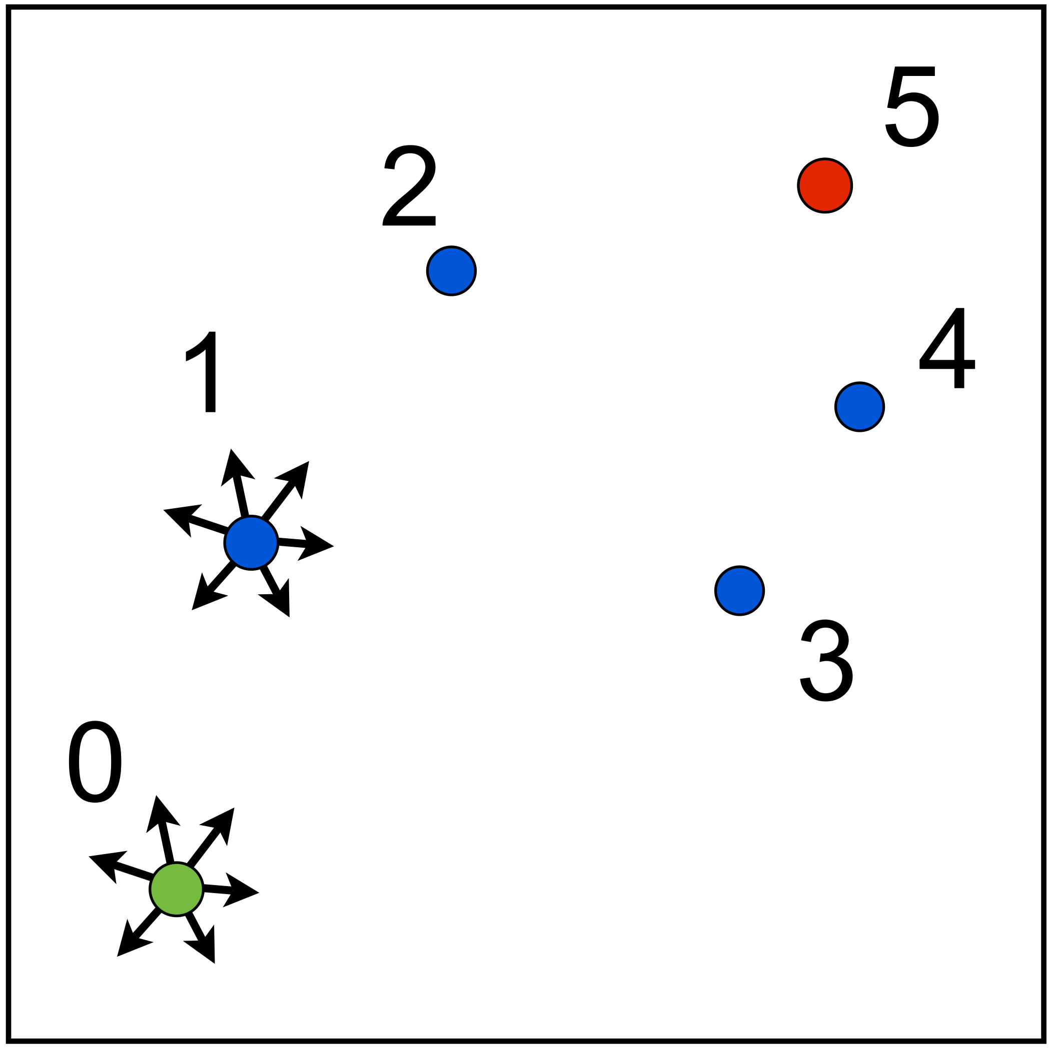
Simple physical layer:

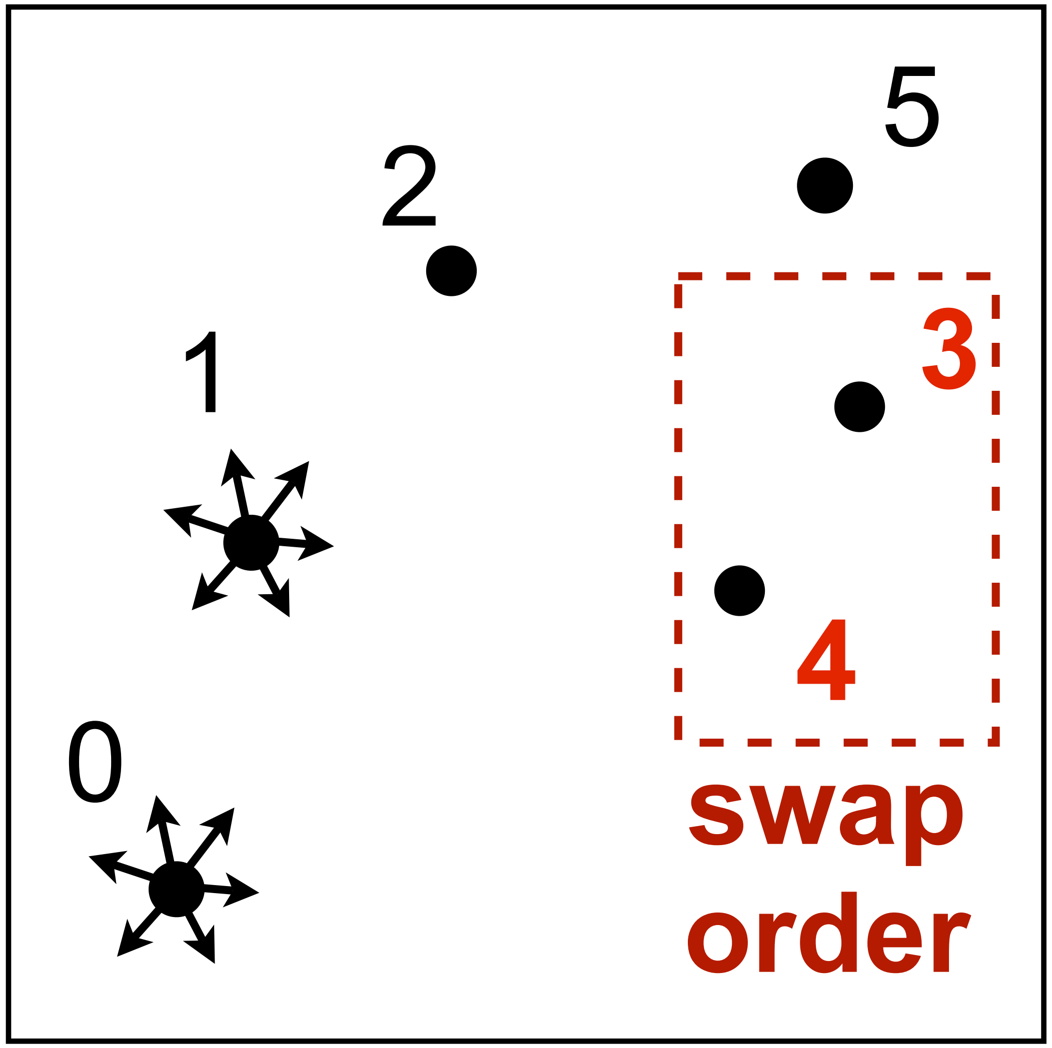
- fixed transmit power
- non-interfering channels
- “Perfect” fountain codes
- Receivers use mutual information (MI) accumulation

Problems are:

- Who transmits?
- When, for how long, and using how much band-width?

Break into two sub-problems





A) for fixed “decoding order” resource allocation is a Linear Program.

• computationally quick

B) revise decoding order based on LP optimum

• for 50 nodes 10^{63} orderings

(Some) related work

Maric & Yates JSAC ’04, JSAC ’05

- also decouple problem and pose a Linear Program
- “energy-accumulation” rather than “MI accumulation” similar at low-SNR, different at high-SNR

Yang & Host-Madsen EURASIP ’06

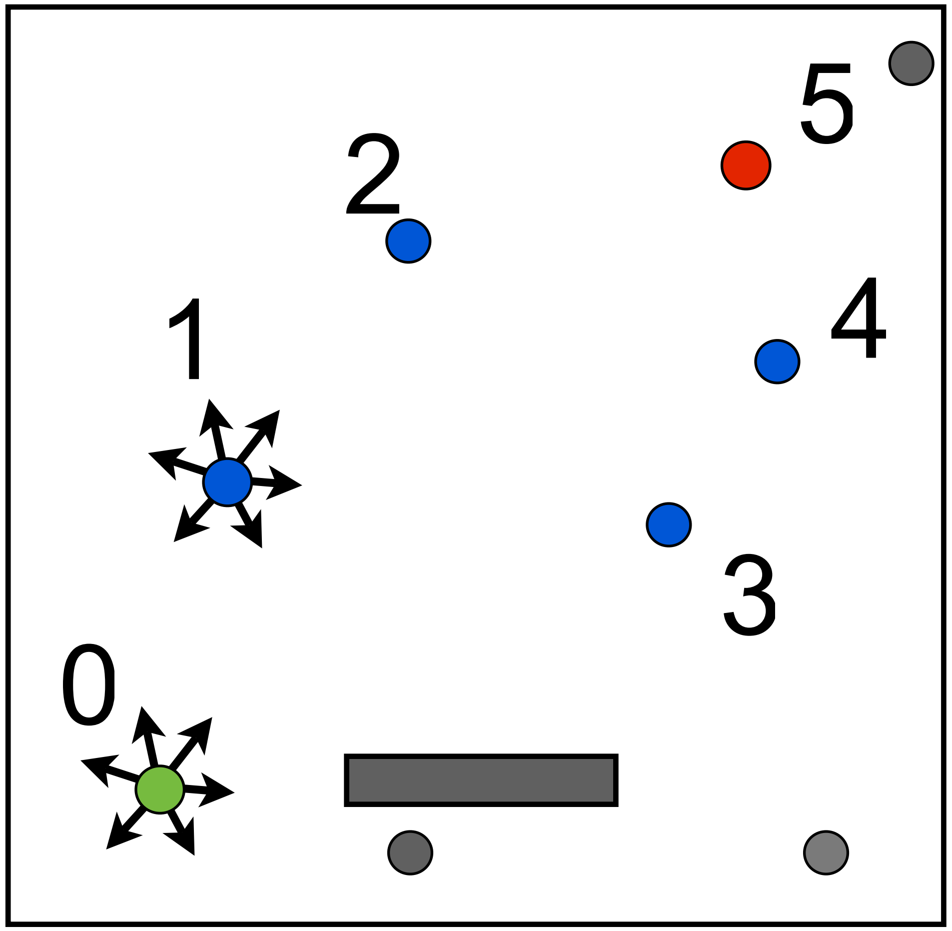
- power-allocation for selected routes

Neither explores using result of optimization for given decoding order to revise decoding order

Outline

- Motivations and Fountain Code Background
- Routing Problem Statement
- Resource Allocation Linear Program
- Decoding Order Revision
- Sample Results

Decoding order

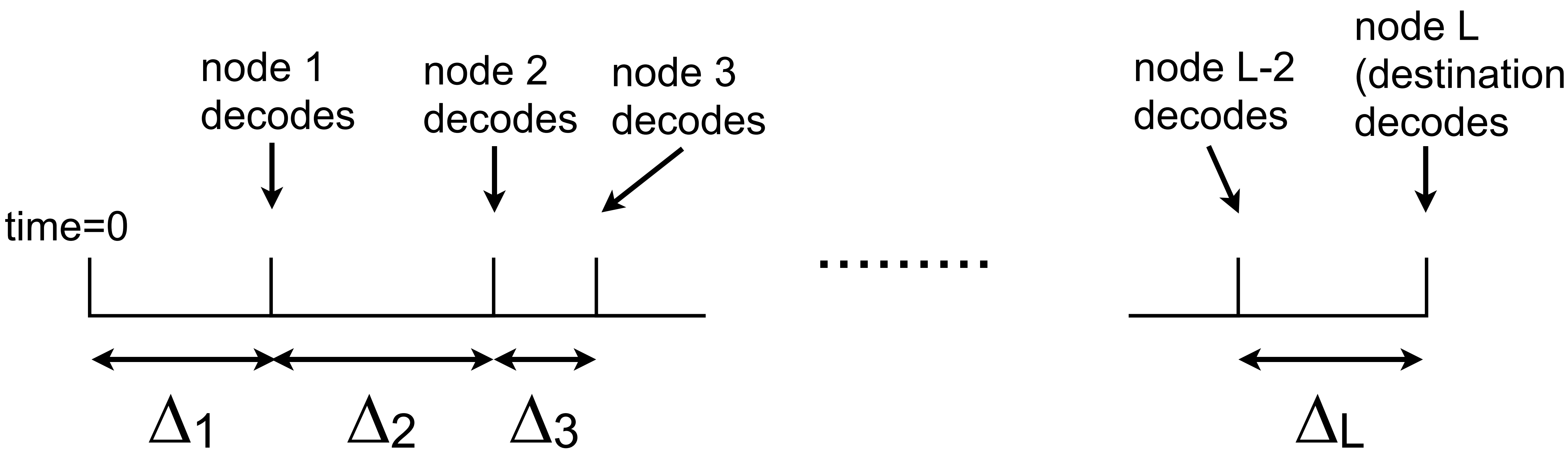


The “decoding order” is the order in which nodes are able to come on-line as relays

- Always starts with source & finishes with destination
- Need not include all nodes, e.g.,

Parameterization: inter-node delays

Inter-node decoding delay (node i-1 to i) = Δ_i



Minimize delay = min $\sum_{i=1}^{L} \Delta_i$

Decoding order induces linear constraints

Pairwise capacities :

$$C_{i,j} = \log_2\left[1 + \frac{h_{i,j}P_iW_i}{N_0W_i}\right] = \log_2\left[1 + \frac{h_{i,j}P_i}{N_0}\right] \text{ bits/s/Hz},$$

Decoding constraints:

$$\sum_{i=0}^{k-1}\sum_{j=i+1}^{k} A_{i,j}C_{i,k} \geq B \quad \text{for all} \quad k \in \{1, 2, \ldots, L\}$$

$A_{i,j}$ is the transmission-time band-width product used by node *i* in time-slot *j*

Resource constraints also linear

Per-node band-width constraints:

$$A_{i,j} \leq \Delta_j W_{\text{node}} \quad \text{for all} \quad \begin{array}{l} i \in \{0, 1, \ldots L-1\} \\ j \in \{1, 2, \ldots, L\} \end{array}$$

Sum-energy constraint:

$$\sum_{i=0}^{L-1} \sum_{j=1}^{L} A_{i,j} P_i = \sum_{i=0}^{L-1} \sum_{j=i+1}^{L} A_{i,j} P_i \leq E_{\mathrm{T}}$$

Variety of other scenarios also linear

Constraints:

- Band-width: per-node or sum-across-nodes
- Energy: per-node or sum-across-nodes

Alternate objective functions:

- Min energy subject to delay: $\sum_{i=1}^{L} \Delta_i \leq \tau_{tot}$
- Min time-BW product: $\sum_{i=0}^{L-1} \sum_{j=i+1}^{L} A_{i,j} P_i$

For fixed decoding order LP solution is optimum resource allocation

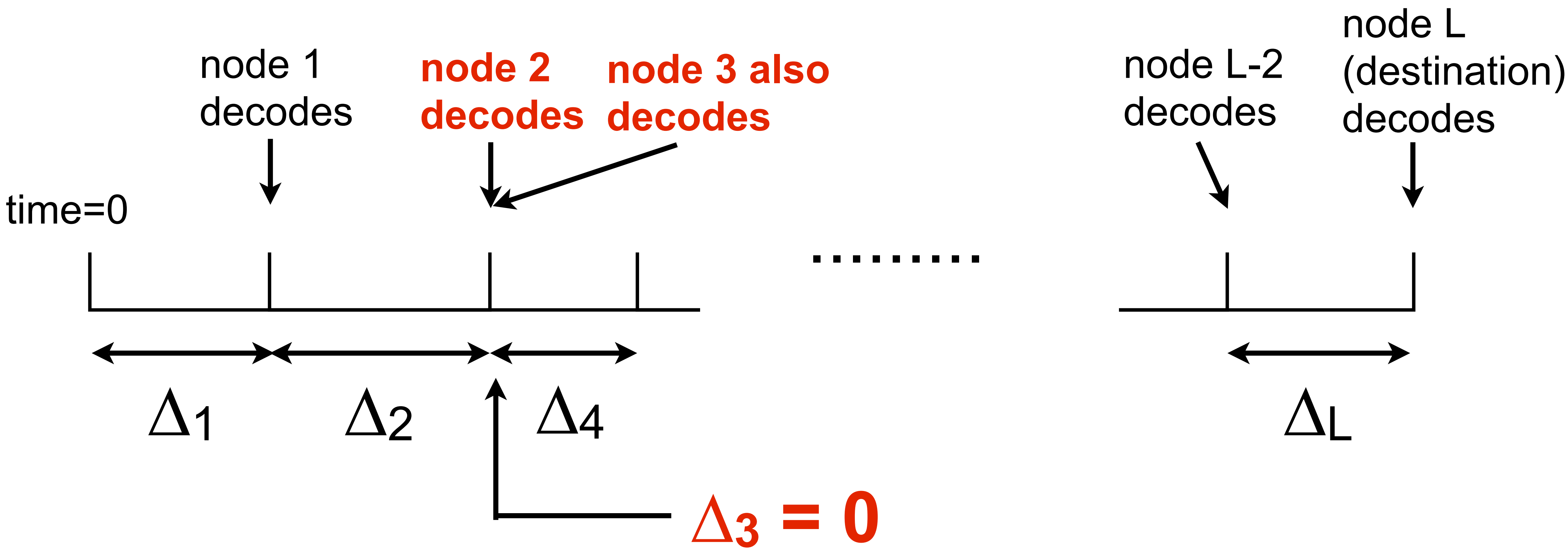
But, there are a massive number of orderings

How do we search that space efficiently?

Outline

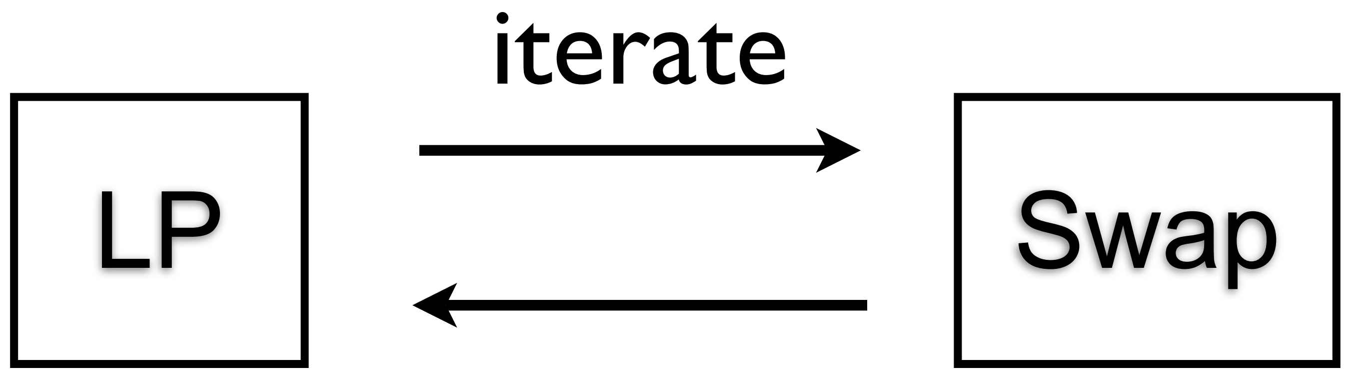
- Motivations and Fountain Code Background
- Routing Problem Statement
- Resource Allocation Linear Program
- Decoding Order Revision
- Sample Results

LP solution suggests a revised order



- If $\Delta_3 = 0$ then "swap" ordering of nodes 2 & 3
- Old solution is feasible for new order
- Re-run LP, delay can only get better or stay same
- If swap nodes L-1 & L (destination), "drop" L-1 from order

Iterative algorithm

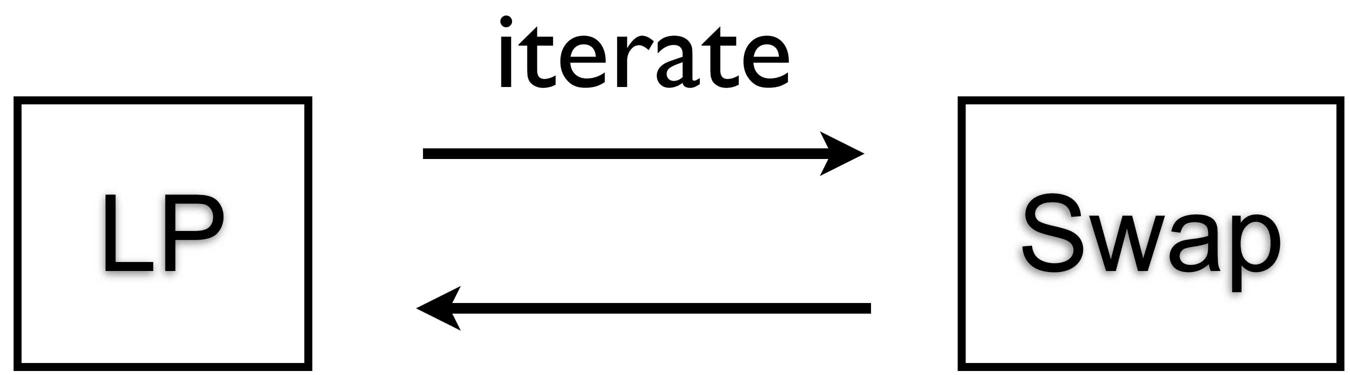


until LP solution satisfies

$\Delta_i \geq 0$ for all i

- Only necessarily local optimum. For small networks (8-10 nodes) optimum often global
- Problem when multiple $\Delta_i = 0$; which swaps to make?
- Start from minimum delay “flooding” order and sequentially tighten energy constraint

Multicasting formulation



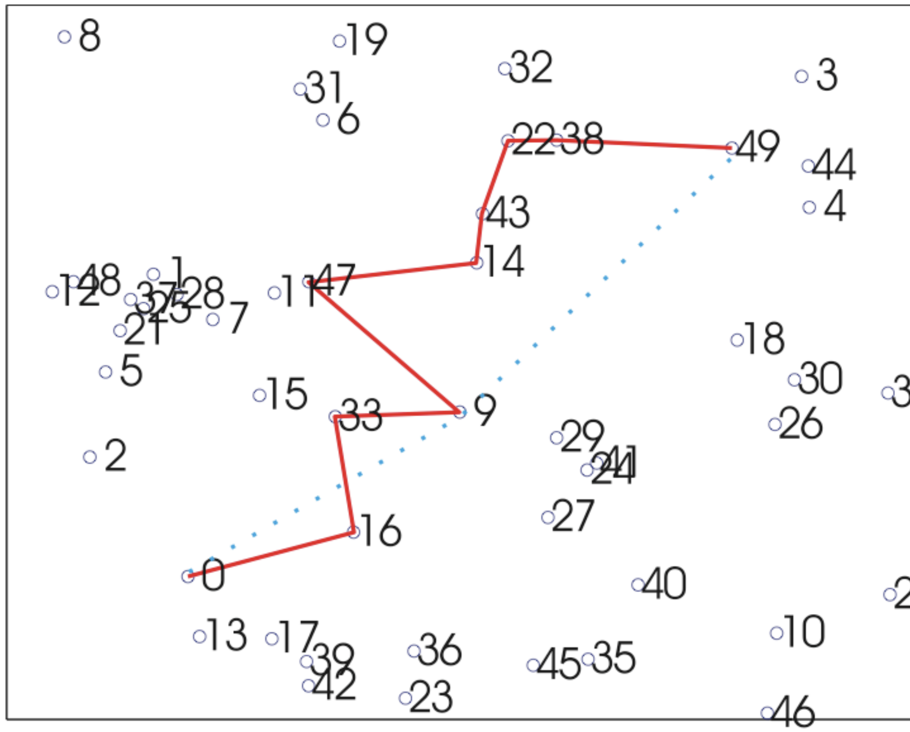
until LP solution satisfies
$\Delta_i \geq 0$ for all i

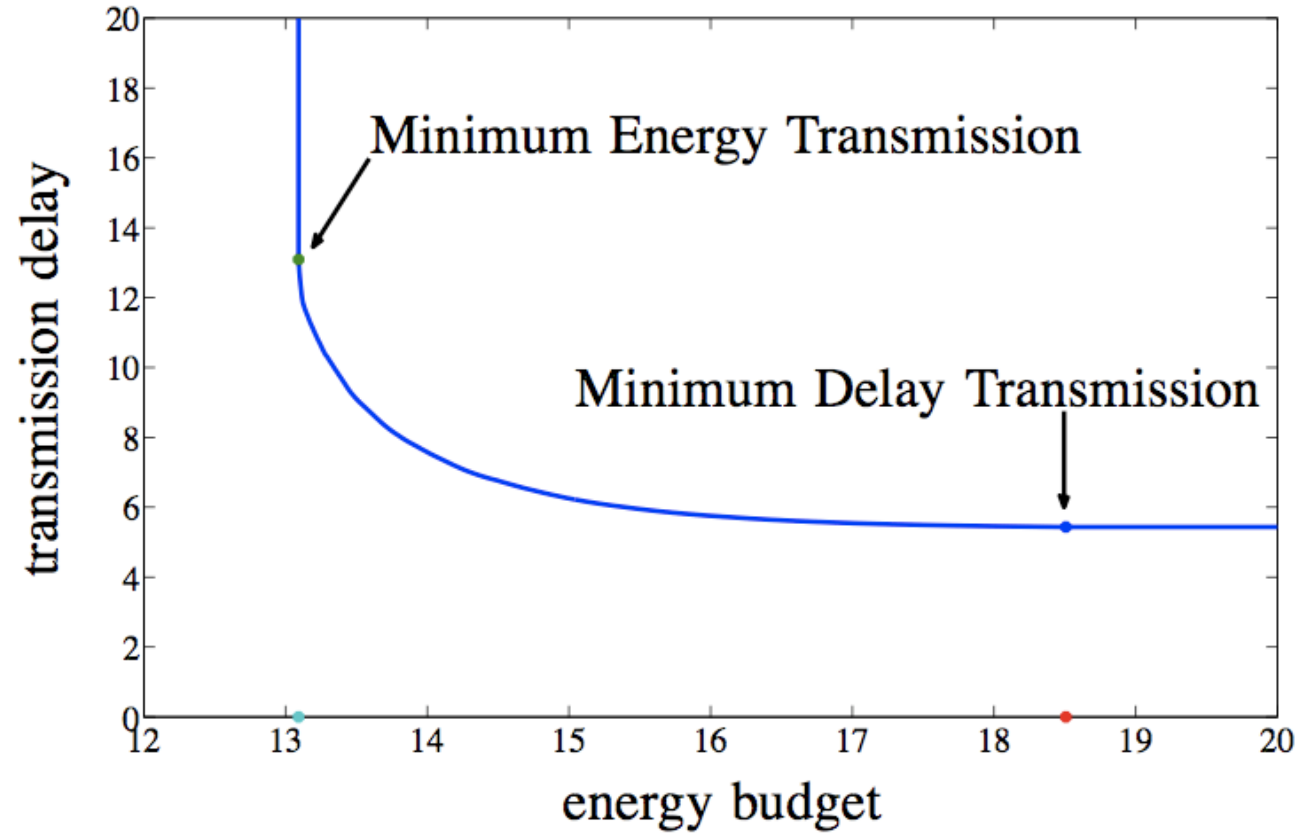
Same algorithm, simply never drop the (now multiple) “destination nodes” from the decoding order

Outline

- Motivations and Fountain Code Background
- Routing Problem Statement
- Resource Allocation Linear Program
- Decoding Order Revision
- Sample Results

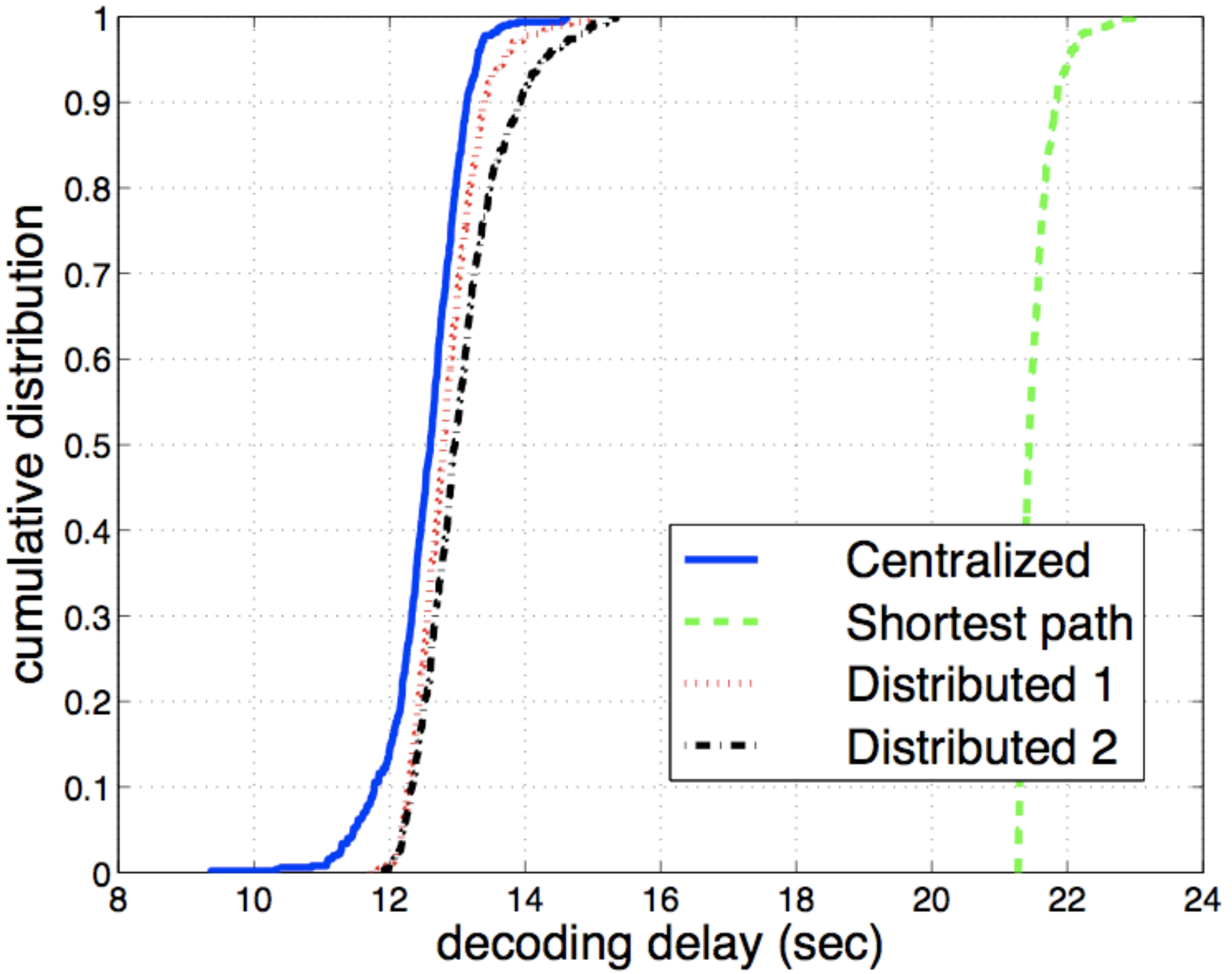
Results: 50 nodes, per-node BW





- Node numbering arbitrary, channel quality $h_{i,j} = (d_{i,j})^{-2}$
- At min delay all nodes except 3, 4, 44 relay
- At min energy cooperative route follows red line. N.B.: at minimum energy, only one node transmits in each time-slot.
- Compare with Dijkstra (dotted), 21.4 sec vs. 13.1 sec., also uses comparably less energy.
- Half of gain is from MI accumulation, half is from using appropriate route

Averaged over 500 node placements



- Mean delay 12.5 sec vs 21.5 sec
- Distributed algorithm 2: whenever a node with a better channel to the destination decodes, it takes over.

Conclusions etc.

Summary

- Fountain codes enable more efficient communication systems, including cooperative systems.
- We have shown how to optimize routes for wireless cooperative networks that use MI accumulation.
- The routing problem is broken into two sub-problems, (a) decoding order, (b) resource allocation given order, iterate between

Future work

- adjusting power levels
- multiple flows
- building prototype