

Predicting Activity in Low-Density Parity-Check Decoders Through Density Evolution

CMOSET

February 19, 2009

Brendan Crowley

University of Alberta

Outline

- **Background**
 - LDPC codes, decoding, density evolution
- **Research**
 - Density evolution to predict decoder activity
- **Results**
 - How accurate results are
 - How results are used
- **Future Work and Conclusion**

Low-Density Parity-Check codes

background

research

results

conclusion

- LDPC codes are a type of forward error correcting code
- Can have superior performance
- Supported by several standards:
 - IEEE 802.3an (10GBASE-T ethernet)
 - IEEE 802.11n (WLAN)
 - IEEE 802.16e (WiMAX)
 - DVB-S2

LDPC Codes

background

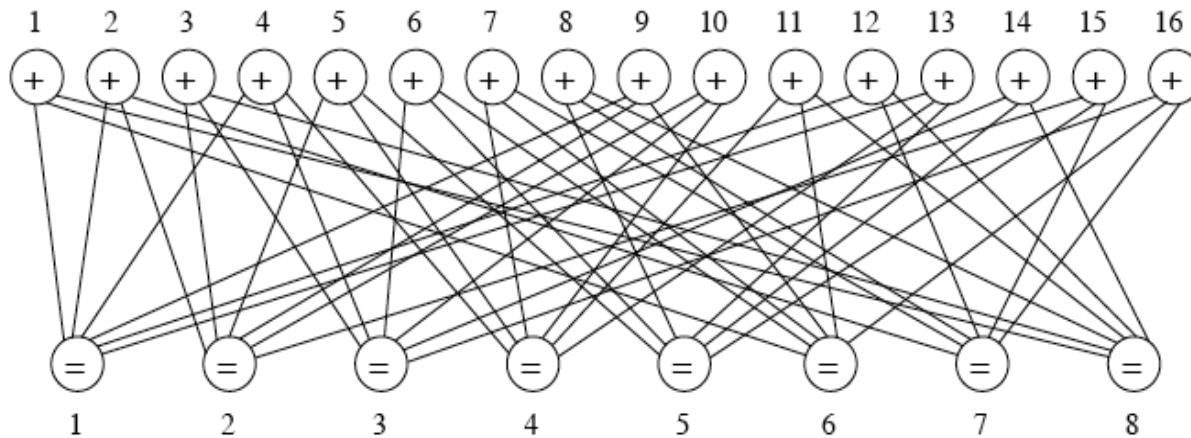
research

results

conclusion

$$H = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \end{bmatrix}$$

- Parity-check matrix (H)
- H is sparse and random
 - Reduces cycles
 - improves performance
- Longer codes perform better



Factor graph and H for a (3,6) length-16 LDPC code

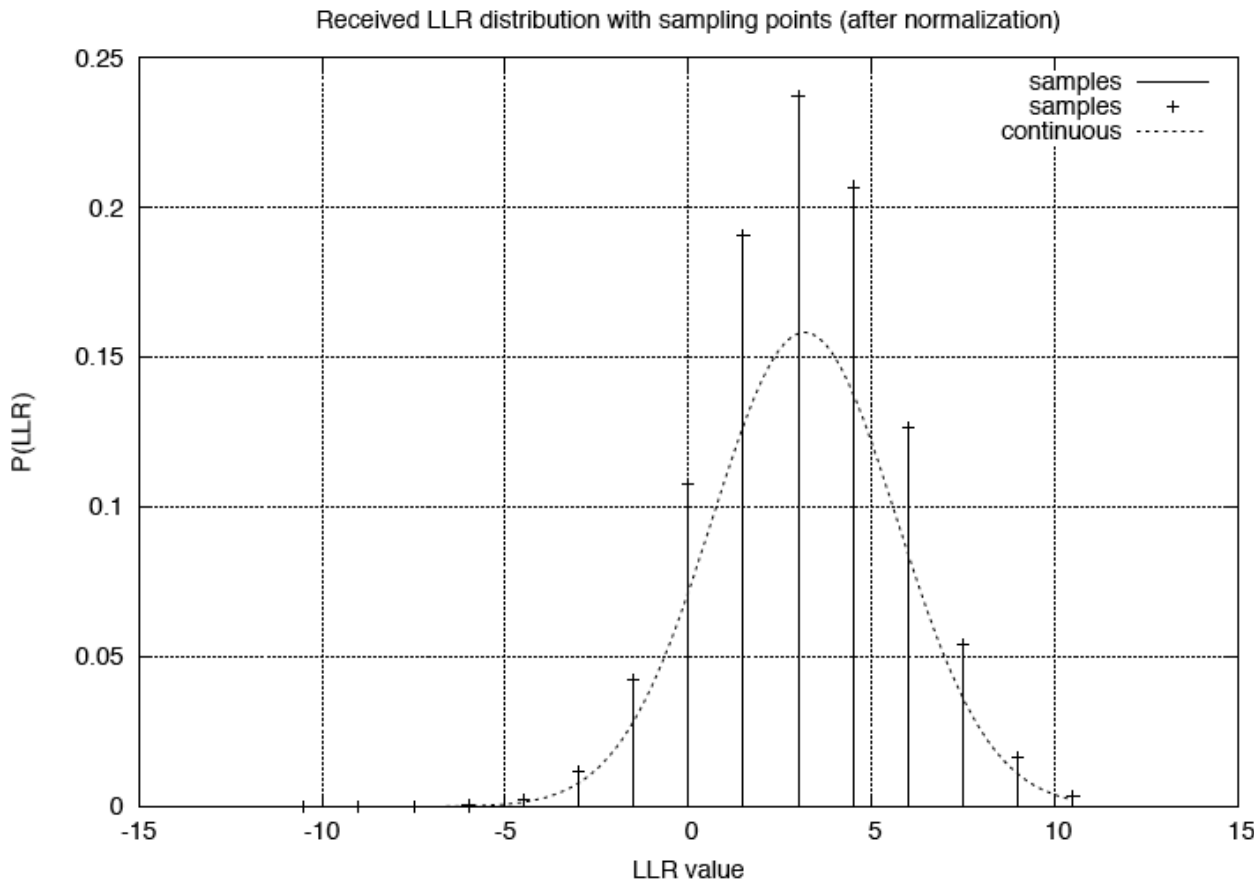
LLR Messages

background

research

results

conclusion



- Assume binary antipodal signaling
- Log-Likelihood-Ratio
- Sign:
 - Predicted bit
- Magnitude:
 - Reliability of bit prediction

$$\text{LLR} = \text{Ln} \left(\frac{P[+1]}{P[-1]} \right)$$

Min-Sum Algorithm

background

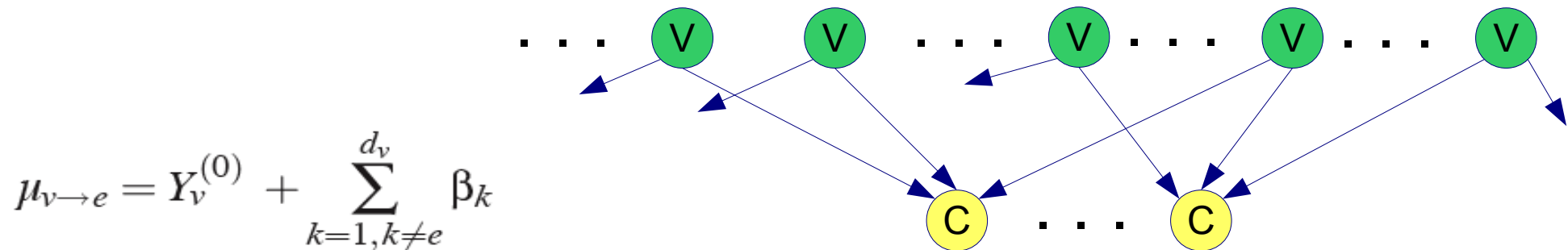
research

results

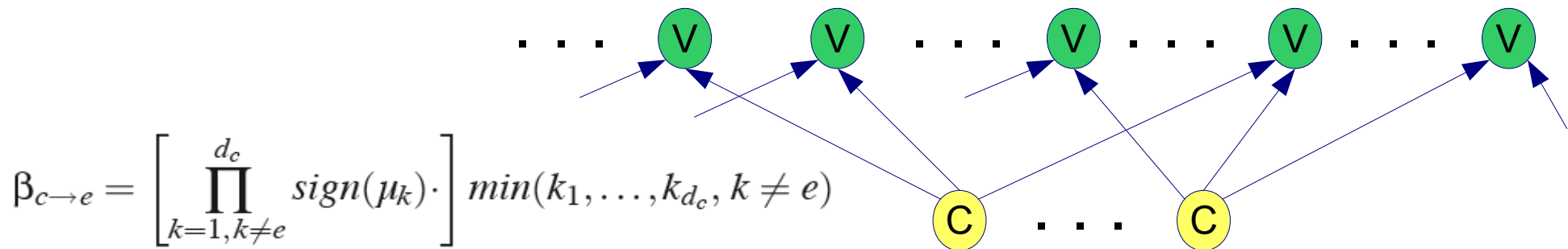
conclusion

- Iterative message-passing algorithm

Variable node \rightarrow Check node message-passing



Check node \rightarrow Variable node message-passing



- Requires a stopping criterion

Density Evolution (DE)

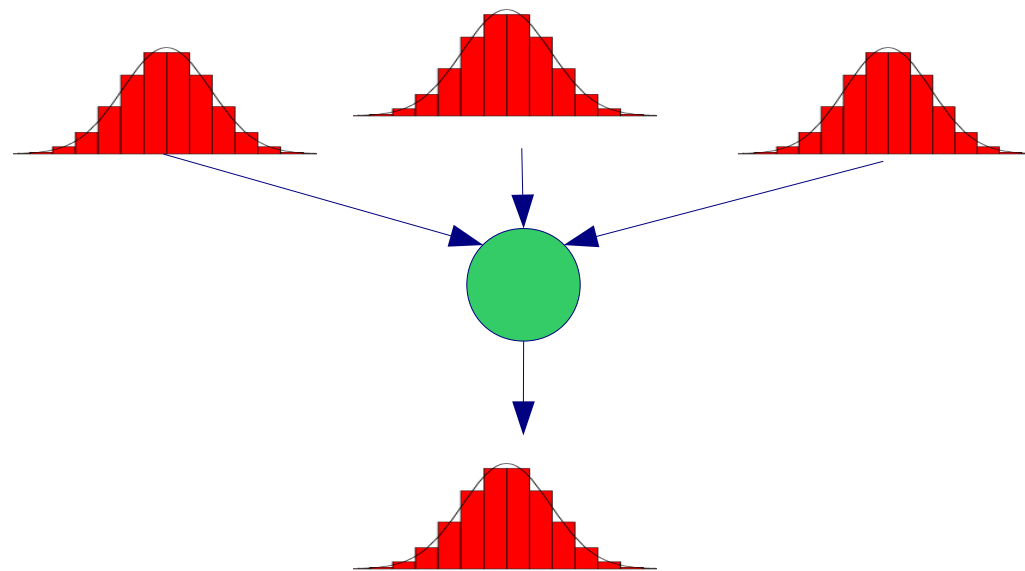
background

research

results

conclusion

- DE is used to analyze the performance of LDPC codes under message-passing decoding
 - PDF of LLR messages passed in each iteration
 - *Code threshold*
- Need only the initial message distribution, and node degrees



LDPC Decoder Power

background

research

results

conclusion

- Basic decoder is direct implementation of the code's factor graph:
 - Nodes (perform calculations)
 - Interconnect (wires transmit messages)
- Large codes are desired for their performance
 - Interconnect can become complex
 - Many long wires



Switching activity on interconnect can consume large amount of power

Factors Affecting Activity

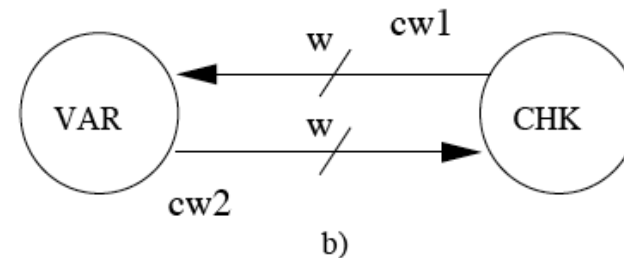
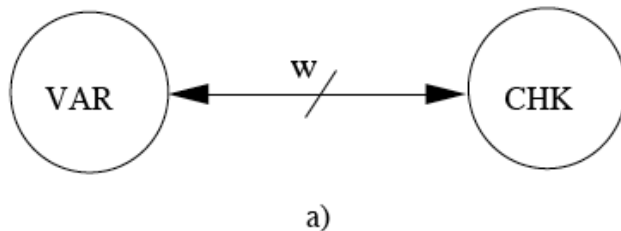
background

research

results

conclusion

- LLR Message bit precision
- One-wire (a) vs. Two-wire (b)



- Message encoding
 - sign+magnitude, two's complement, etc.
- Parallel vs. Serial LSB-first, MSB-first
- Stopping criterion

Counting Transitions

background

research

results

conclusion

- Find Joint PDF for each message change
- Given the message implementation, create a transitions matrix

	b1	b0
-1	1	1
0	0	0
+1	0	1

2-bit message encoding

		new		
		-1	0	+1
old	-1	0	2	1
	0	2	0	1
	+1	1	1	0

Parallel transitions matrix

		new		
		-1	0	+1
old	-1	0.15	0.20	0.05
	0	0.10	0.05	0.05
	+1	0.05	0.20	0.15

Joint PDF

Counting Transitions

background

research

results

conclusion

- To find average transitions on one graph edge in an iteration, do the following for *chk-var* and *var-chk* messages:

$$trans = \sum_{i=1}^{2^w-1} \sum_{j=1}^{2^w-1} \mathbf{J}(i, j) * \mathbf{T}(i, j)$$

old	new	old	new	trans	x	prob	=	result
-1	-1	11	11	0	x	0.15	=	0.00
-1	0	11	00	2	x	0.20	=	0.40
-1	+1	11	01	1	x	0.05	=	0.05
0	-1	00	11	2	x	0.10	=	0.20
0	0	00	00	0	x	0.05	=	0.00
0	+1	00	01	1	x	0.05	=	0.05
+1	-1	01	11	1	x	0.05	=	0.05
+1	0	01	00	1	x	0.20	=	0.20
+1	+1	01	01	0	x	0.15	=	0.00
						total	=	0.95

Power Stopping Criteria

background

research

results

conclusion

- A power stopping criterion monitors activity
 - Stop when decoder has converged
 - Monitor the total number of transitions in an iteration, stop when the number is zero
 - Stop when a certain amount of energy has been consumed
 - Monitor the total number of transitions in all iterations, stop when the number exceeds some threshold
 - Tradeoff with performance

Regular DE Results

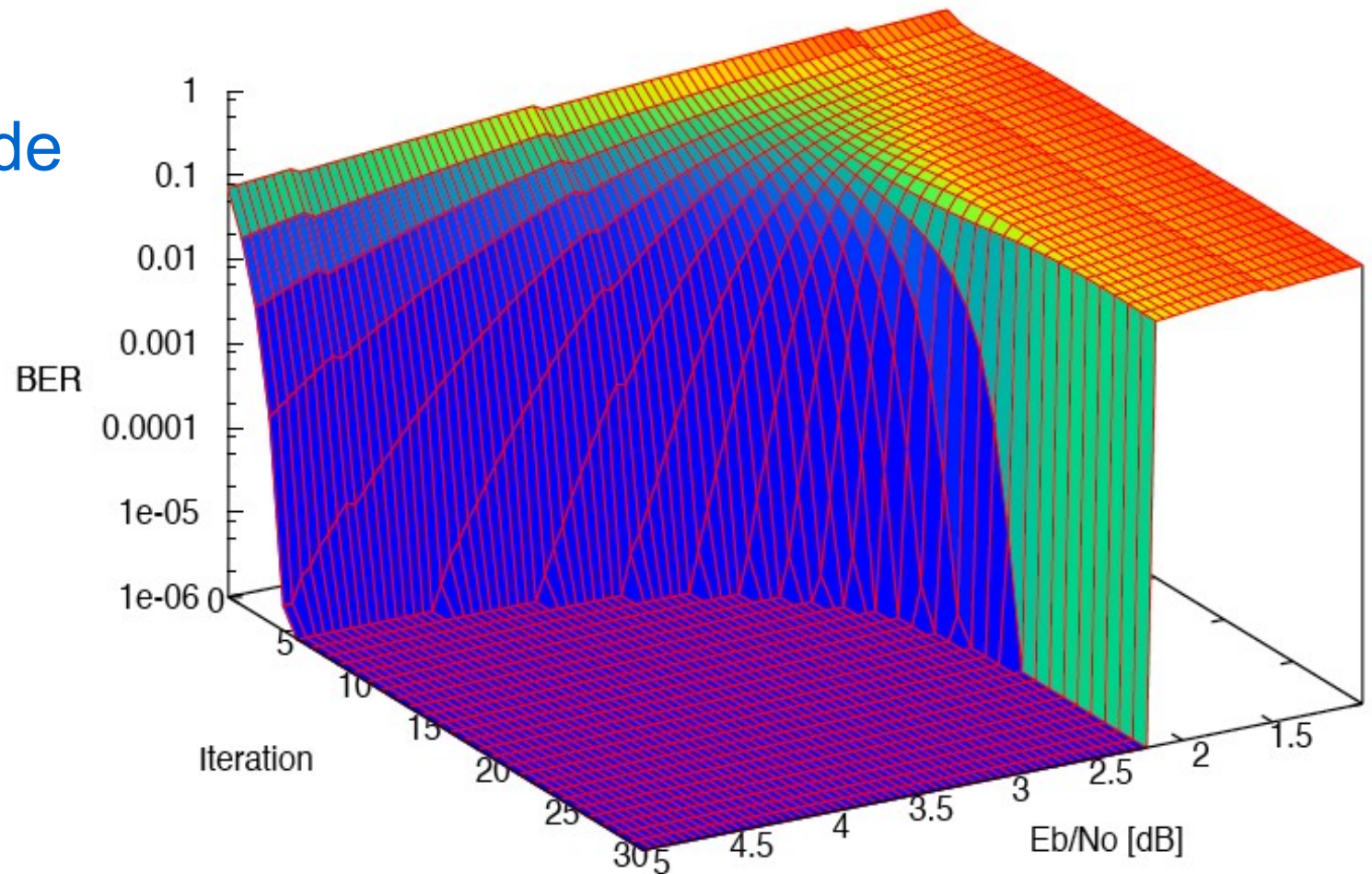
background

research

results

conclusion

- BER vs. Iteration vs. SNR
 - $w = 3$
 - (3,6) code



Reporting Activity Results

background

research

results

conclusion

- Typically activity is reported as the fraction of clock cycles in which a bit transition occurs
- Here it is reported as a number of transitions:
 - Transitions per edge per codeword (TEC)
 - TEC per iteration (TECI)
- Multiply TEC by the number of edges in the code to find the activity required to decode one codeword

Implementation Shorthand

background

research

results

conclusion

- Activity curves are labelled “WE” as follows:

Position	Description	Value	Description
W	Wire Type	P	Parallel
		L	Serial LSB first
		M	Serial MSB first
E	Encoding	S	Sign + Magnitude
		I	Sign + Magnitude, Inverted Sign
		T	Two's Complement

- Ex. LS = Serial LSB-first, sign+magnitude

DE Activity Curves

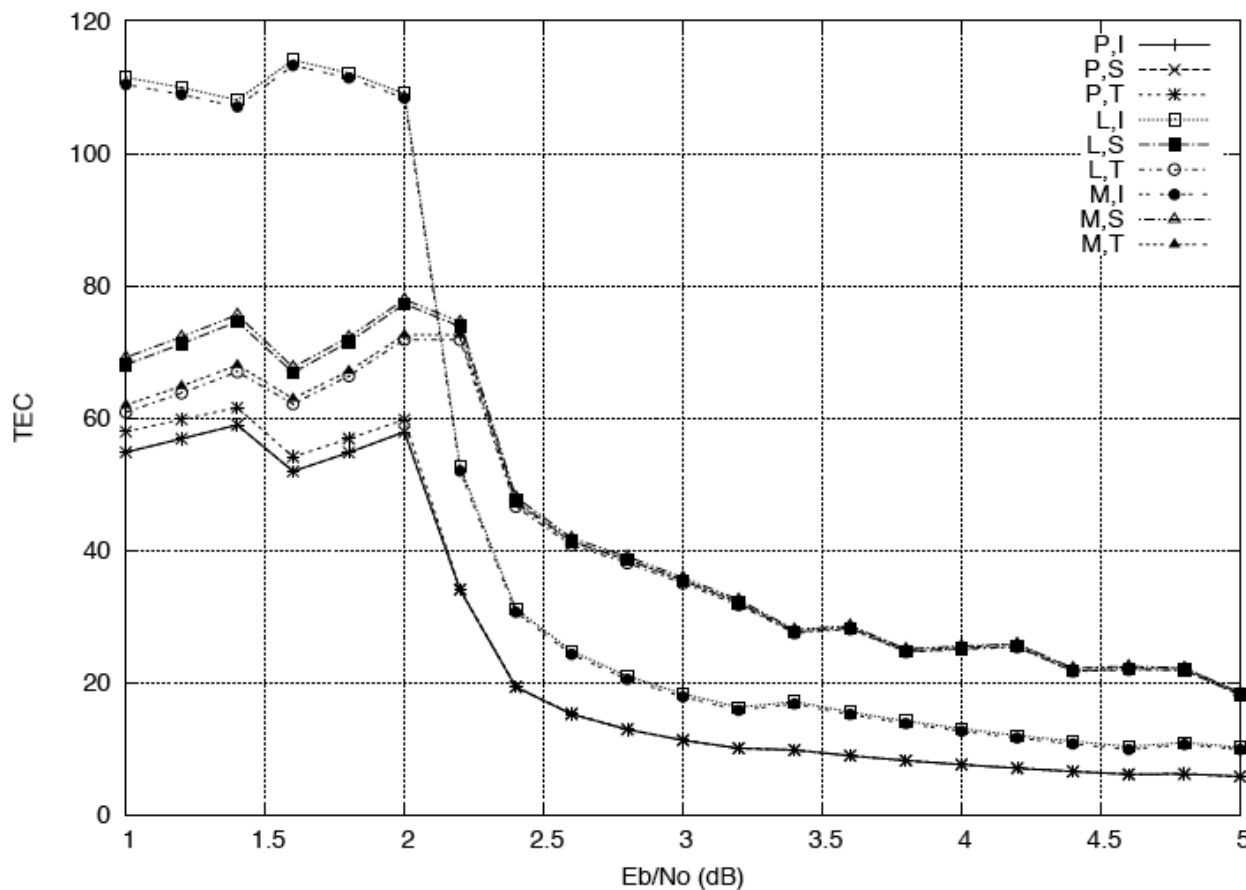
background

research

results

conclusion

- **TEC vs. SNR**
 - $w = 3$, all formats



TEC relative to PS

WE	2.2dB	3.0dB	4.0dB
PS	1.000	1.000	1.000
PI	1.000	1.000	1.000
PT	1.004	1.004	1.003
MS	2.194	3.189	3.372
MI	1.523	1.584	1.670
MT	2.135	3.155	3.359
LS	2.173	3.148	3.323
LI	1.550	1.625	1.719
LT	2.114	3.113	3.310

DE Activity Curves

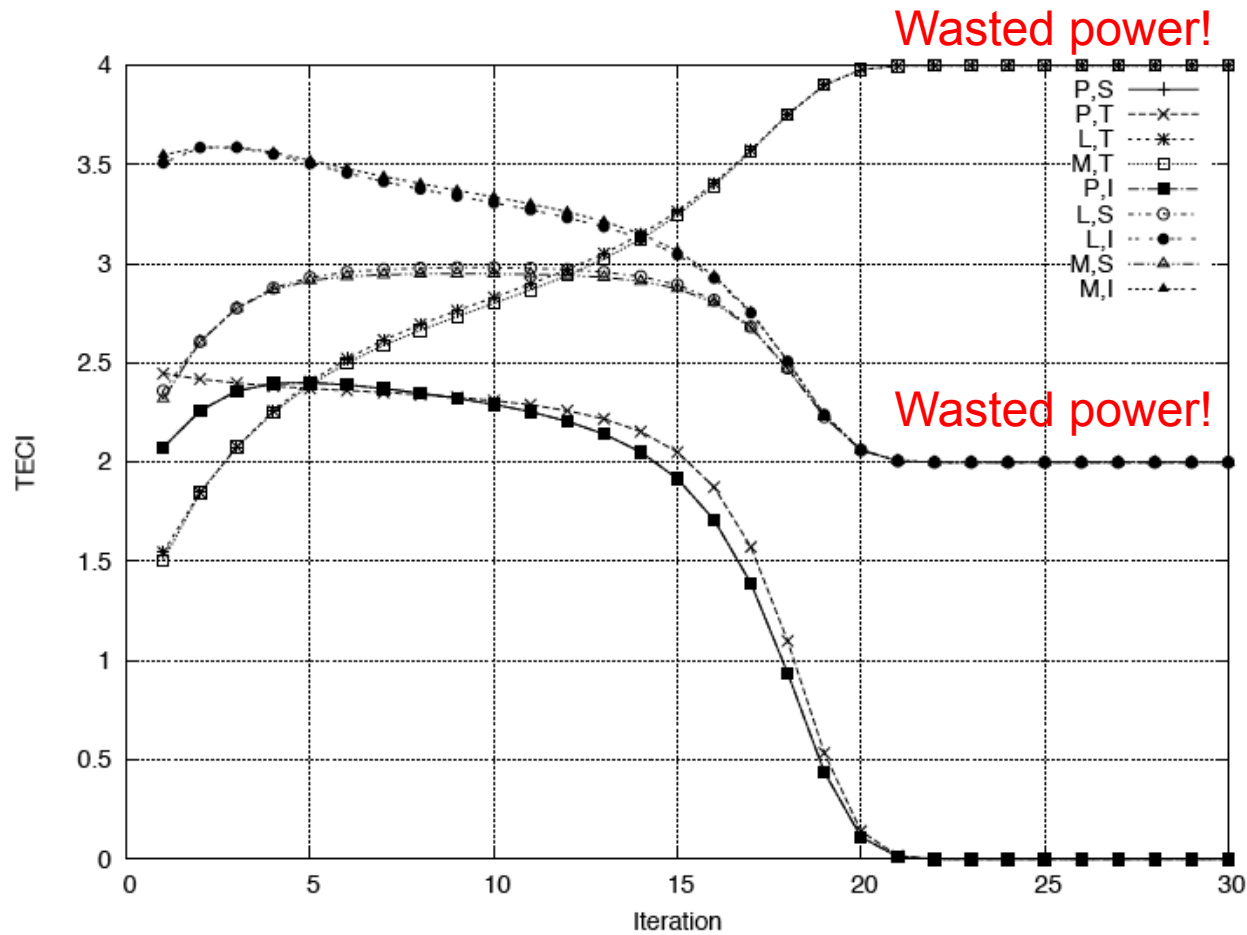
background

research

results

conclusion

- **TECI vs. Iteration**
 - SNR = 2.2dB, $w = 3$, all formats



DE Activity Curves

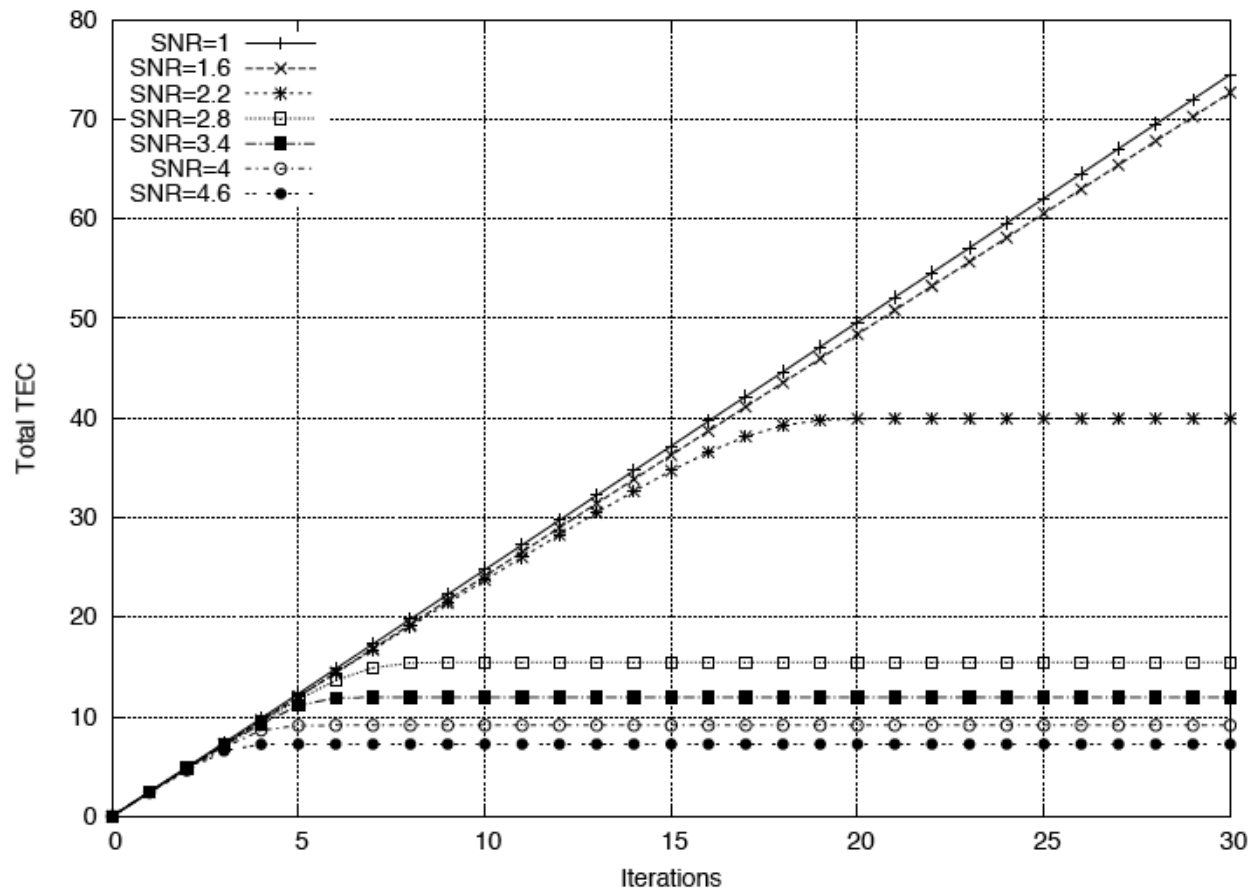
background

research

results

conclusion

- Cumulative Total TEC vs. Iteration
 - $w = 3, PT$



DE BER Curves

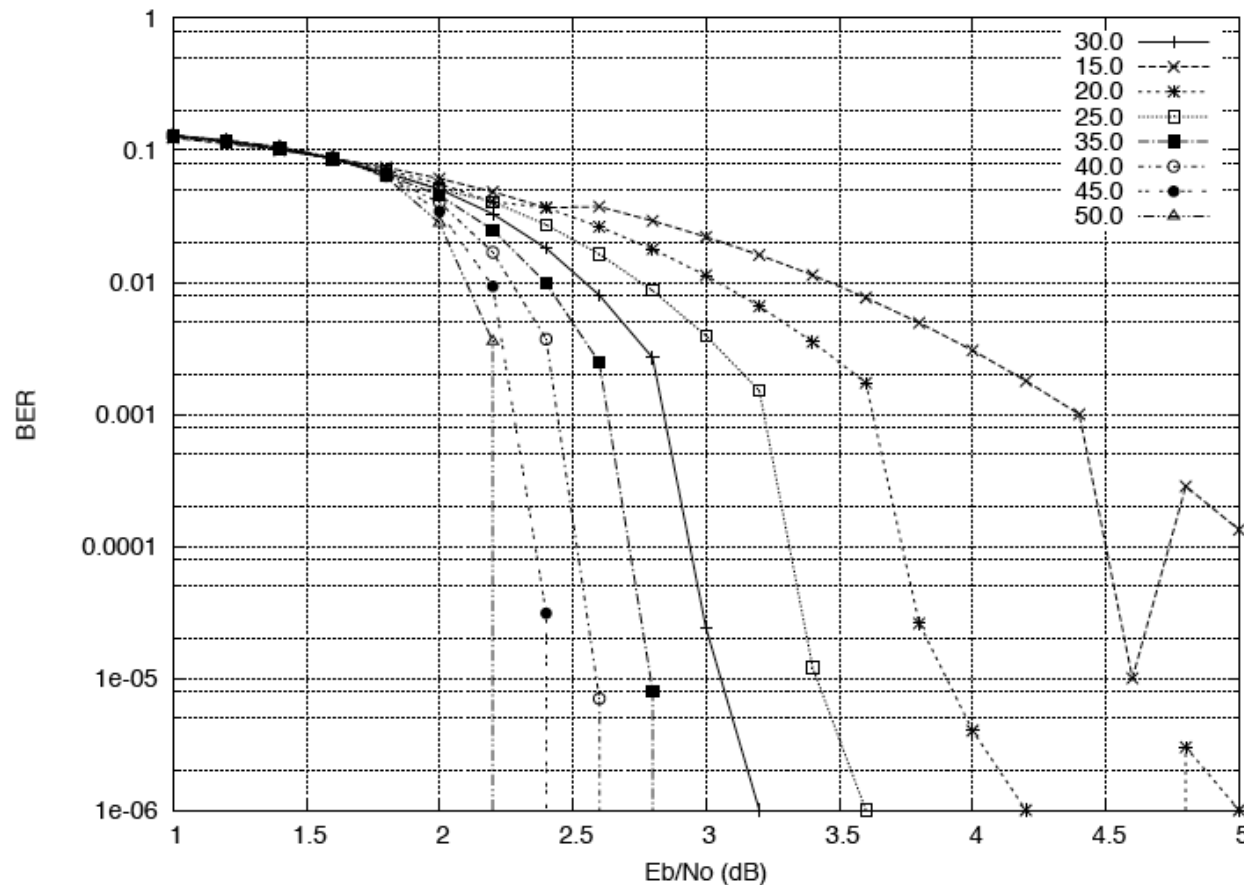
background

research

results

conclusion

- BER vs. SNR using power stopping criterion
 - $w = 5, MI$



MC vs. DE Activity Curves

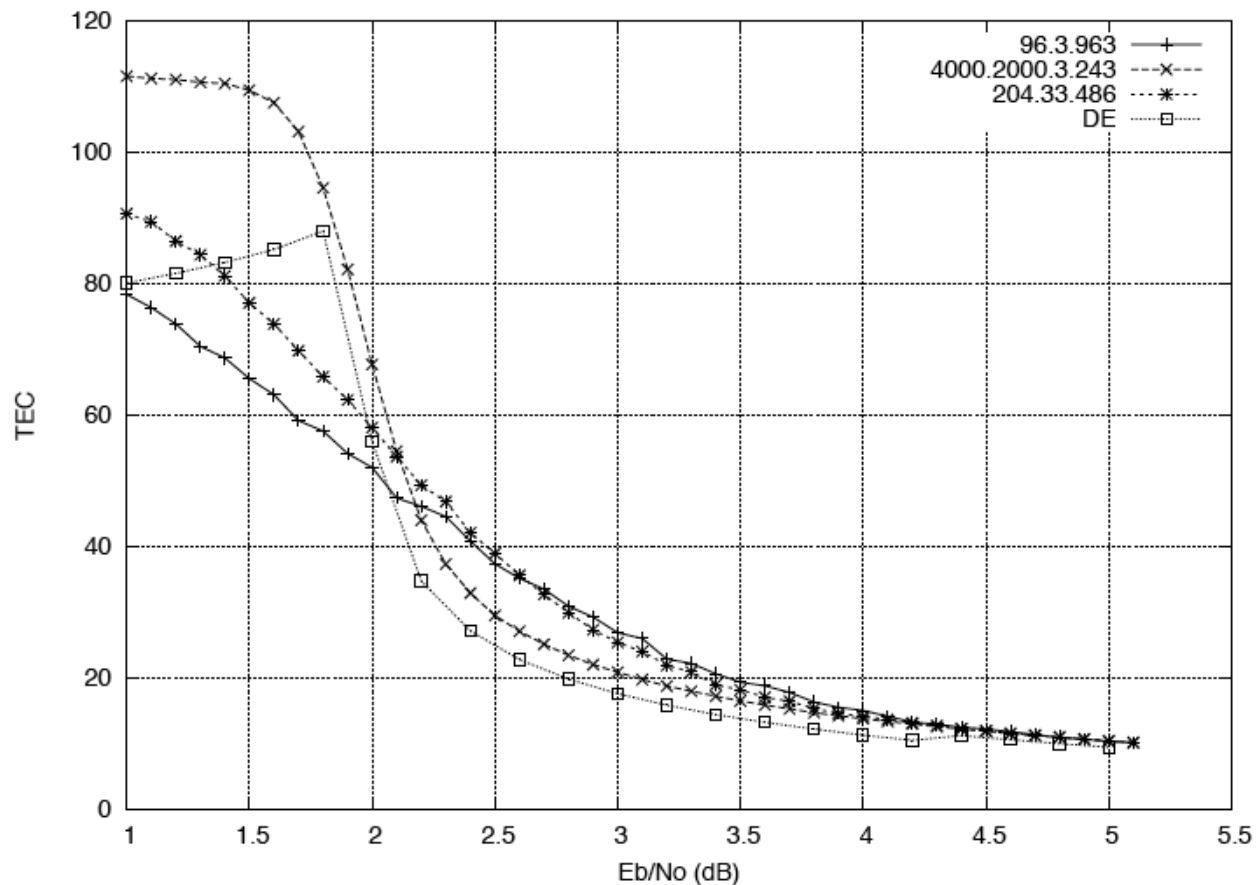
background

research

results

conclusion

- TEC vs. SNR
 - $w = 4, PT$



TEC relative to DE

SNR	2.6	3.6	4.6
MC-96	1.54	1.42	1.11
MC-204	1.56	1.29	1.08
MC-4000	1.19	1.20	1.09
DE	1.00	1.00	1.00

MC vs. DE Activity Curves

background

research

results

conclusion

- Cumulative TEC vs. Iteration (relative to DE)
 - $w = 3, PT$

SNR = 2.8dB

Iteration	1	3	5	10	15	20	25
MC-96	1.088	1.091	1.025	1.072	1.263	1.427	1.568
MC-4000	1.092	1.101	1.103	1.192	1.201	1.202	1.202
DE	1.000	1.000	1.000	1.000	1.000	1.000	1.000

SNR = 4.0dB

Iteration	1	3	5	10	15	20	25
MC-96	1.094	1.082	1.057	1.157	1.199	1.230	1.256
MC-4000	1.094	1.090	1.121	1.125	1.125	1.125	1.125
DE	1.000	1.000	1.000	1.000	1.000	1.000	1.000

Summary of Results

background

research

results

conclusion

- DE is useful for quick activity prediction, and the selection of power stopping criterion
- Use model very early in design process to make implementation choices
- With appropriate scaling factors, DE could be used to predict real power consumption as a function of code size

Future Work

background

research

results

conclusion

- Test other message encodings
- Test irregular codes
- Verify with fabricated decoders
- Design circuits to monitor power/activity
- Apply to other iteratively decoded codes
- Create activity models for other components of the decoder

Conclusion

background

research

results

conclusion

- Combined knowledge of message statistics from DE with knowledge of physical decoder implementation to create a model for activity prediction in message-passing decoders
- Introduced stopping criterion based on activity
- Resulting DE model is quite accurate
 - More accurate as SNR and code size increase
- Can be used early in the code design process to aid low-power design