Graphical Models and Inference: Insights from Spatial Coupling

Henry D. Pfister

Electrical and Computer Engineering
Information Initiative (iiD)
Duke University

IEEE ITSOC Distinguished Lecture
Lund University, Sweden
September 16th, 2016

Acknowledgments

- Thanks to all my coauthors involved in this work
  - Krishna Narayanan
  - Phong Nguyen
  - Arvind Yedla
  - Yung-Yih Jian
  - Santhosh Kumar
- Thanks for the invitation and support
  - Michael Lentmaier
  - Lund University
  - IEEE Information Theory Society

Outline

Outline

Graphical Models

Point-to-Point Communication

Low-Density Parity-Check Codes

Compressed Sensing

Universality for Multiuser Scenarios

Natural Spatial Coupling in Cellular Systems

Abstract Formulation of Threshold Saturation

Graphical Models

- A graphical model provides a graphical representation of the local dependence structure for a set of random variables
  - factor graphs [KFL01], Bayesian networks [Pea88], etc...

Graphical Models

- A graphical model provides a graphical representation of the local dependence structure for a set of random variables
  - factor graphs [KFL01], Bayesian networks [Pea88], etc...
- Consider random variables $(X_1, X_2, \ldots, X_4) \in \mathcal{X}^4$ and Y where:

$$\begin{aligned} P(x_1, x_2, x_3, x_4) &\triangleq \mathbb{P}(X_1 = x_1, X_2 = x_2, \ldots, X_4 = x_4 | Y = y) \\ &\propto f(x_1, x_2, x_3, x_4) \\ &\triangleq f_1(x_1, x_2) f_2(x_2, x_3) f_3(x_3, x_4) \end{aligned}$$

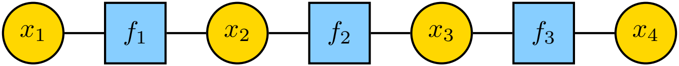
Graphical Models

- A graphical model provides a graphical representation of the local dependence structure for a set of random variables
  - factor graphs [KFL01], Bayesian networks [Pea88], etc...

- Consider random variables $(X_1, X_2, \ldots, X_4) \in \mathcal{X}^4$ and Y where:

$$\begin{aligned} P(x_1, x_2, x_3, x_4) &\triangleq \mathbb{P}(X_1 = x_1, X_2 = x_2, \ldots, X_4 = x_4 | Y = y) \\ &\propto f(x_1, x_2, x_3, x_4) \\ &\triangleq f_1(x_1, x_2) f_2(x_2, x_3) f_3(x_3, x_4) \end{aligned}$$

- Given $Y = y$, this describes a Markov chain whose factor graph is



Inference via Marginalization

- Marginalizing out all variables except X_1 gives

$$\mathbb{P}(X_1 = x_1 | Y = y) \propto g_1(x_1) \triangleq \sum_{(x_2,\ldots,x_4)\in\mathcal{X}^3} f(x_1, x_2, x_3, x_4)$$

- Thus, the maximum a posteriori decision for X_1 given $Y = y$ is

$$\hat{x}_1 = \arg\max_{x_1\in\mathcal{X}} \sum_{(x_2,\ldots,x_4)\in\mathcal{X}^3} f(x_1, x_2, x_3, x_4)$$

  - For a general function, this requires roughly $|\mathcal{X}|^4$ operations

Inference via Marginalization

- Marginalizing out all variables except X_1 gives

$$\mathbb{P}(X_1 = x_1 | Y = y) \propto g_1(x_1) \triangleq \sum_{(x_2,\ldots,x_4)\in\mathcal{X}^3} f(x_1, x_2, x_3, x_4)$$

- Thus, the maximum a posteriori decision for X_1 given $Y = y$ is

$$\hat{x}_1 = \arg\max_{x_1\in\mathcal{X}} \sum_{(x_2,\ldots,x_4)\in\mathcal{X}^3} f(x_1, x_2, x_3, x_4)$$

  - For a general function, this requires roughly $|\mathcal{X}|^4$ operations

- Marginalization is efficient for tree-structured factor graphs
  - For this Markov chain, roughly $5\,|\mathcal{X}|^2$ operations required

$$g_1(x_1) = \sum_{x_2\in\mathcal{X}} f_1(x_1, x_2) \sum_{x_3\in\mathcal{X}} f_2(x_2, x_3) \sum_{x_4\in\mathcal{X}} f_3(x_3, x_4)$$

Sudoku: A Well-Known Example

	2		5		1		9	
8			2		3			6
	3			6			7	
		1				6		
5	4						1	9
		2				7		
	9			3			8	
2			8		4			7
	1		9		7		6	

rows are permutations of $\{1, 2, \ldots, 9\}$

Sudoku: A Well-Known Example

	2		5		1		9	
8			2		3			6
	3			6			7	
		1				6		
5	4						1	9
		2				7		
	9			3			8	
2			8		4			7
	1		9		7		6	

rows are permutations of $\{1, 2, \ldots, 9\}$
columns are permutations of $\{1, 2, \ldots, 9\}$

Sudoku: A Well-Known Example

	2		5		1		9	
8			2		3			6
	3			6			7	
		1				6		
5	4						1	9
		2				7		
	9			3			8	
2			8		4			7
	1		9		7		6	

rows are permutations of $\{1, 2, \ldots, 9\}$
columns are permutations of $\{1, 2, \ldots, 9\}$
subblocks are permutations of $\{1, 2, \ldots, 9\}$

Sudoku: A Well-Known Example

	2		5		1		9	
8			2		3			6
	3			6			7	
		1				6		
5	4						1	9
		2				7		
	9			3			8	
2			8		4			7
	1		9		7		6	

rows are permutations of $\{1, 2, \ldots, 9\}$
columns are permutations of $\{1, 2, \ldots, 9\}$
subblocks are permutations of $\{1, 2, \ldots, 9\}$

x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}	x_{17}	x_{18}	x_{19}
x_{21}	x_{22}	x_{23}	x_{24}	x_{25}	x_{26}	x_{27}	x_{28}	x_{29}
x_{31}	x_{32}	x_{33}	x_{34}	x_{35}	x_{36}	x_{37}	x_{38}	x_{39}
x_{41}	x_{42}	x_{43}	x_{44}	x_{45}	x_{46}	x_{47}	x_{48}	x_{49}
x_{51}	x_{52}	x_{53}	x_{54}	x_{55}	x_{56}	x_{57}	x_{58}	x_{59}
x_{61}	x_{62}	x_{63}	x_{64}	x_{65}	x_{66}	x_{67}	x_{68}	x_{69}
x_{71}	x_{72}	x_{73}	x_{74}	x_{75}	x_{76}	x_{77}	x_{78}	x_{79}
x_{81}	x_{82}	x_{83}	x_{84}	x_{85}	x_{86}	x_{87}	x_{88}	x_{89}
x_{91}	x_{92}	x_{93}	x_{94}	x_{95}	x_{96}	x_{97}	x_{98}	x_{99}

implied factor graph has
81 variable and 27 factor nodes

$$f(\underline{x}) = \left(\prod_{i=1}^{9} f_\sigma(x_{i*})\right)\left(\prod_{j=1}^{9} f_\sigma(x_{*j})\right)\left(\prod_{k=1}^{9} f_\sigma(x_{B(k)})\right)\prod_{(i,j)\in O} \mathbb{I}(x_{ij} = y_{ij})$$

Solving Sudoku via Marginalization

- Consider any constraint satisfaction problem with erased entries
  - One can write $f(\underline{x})$ as the product of indicator functions
  - Some factors force $\underline{x}$ to be valid (i.e., satisfy constraints)
  - Other factors force $\underline{x}$ to be compatible with observed values
  - Summing over $\underline{x}$ counts the # of valid compatible sequences

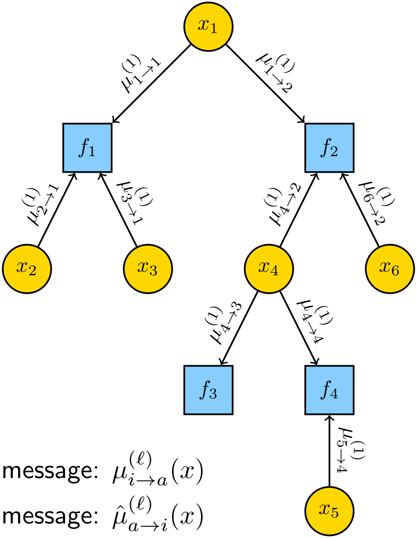
Solving Sudoku via Marginalization

- Consider any constraint satisfaction problem with erased entries
  - One can write $f(\underline{x})$ as the product of indicator functions
  - Some factors force $\underline{x}$ to be valid (i.e., satisfy constraints)
  - Other factors force $\underline{x}$ to be compatible with observed values
  - Summing over $\underline{x}$ counts the # of valid compatible sequences
- Marginalization allows uniform sampling from valid compatible set
  - Sample $x_1' \sim g_1(\cdot)$, fix $x_1 = x_1'$, sample $x_2' \sim g_2(\cdot|x_1)$, etc...
    - For Sudoku, this always works because only one solution!

Solving Sudoku via Marginalization

- Consider any constraint satisfaction problem with erased entries
  - One can write $f(\underline{x})$ as the product of indicator functions
  - Some factors force $\underline{x}$ to be valid (i.e., satisfy constraints)
  - Other factors force $\underline{x}$ to be compatible with observed values
  - Summing over $\underline{x}$ counts the # of valid compatible sequences
- Marginalization allows uniform sampling from valid compatible set
  - Sample $x_1' \sim g_1(\cdot)$, fix $x_1 = x_1'$, sample $x_2' \sim g_2(\cdot|x_1)$, etc...
    - For Sudoku, this always works because only one solution!
  - fast marginalization via BP if factor graph forms a tree
  - But, in general, marginalization is #P-complete

Marginalization via Belief Propagation

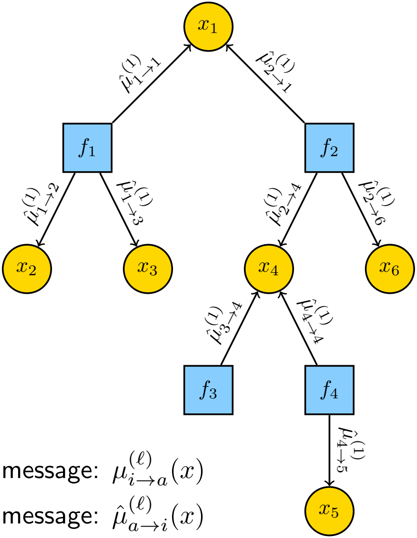


variable-to-factor message: $\mu^{(\ell)}_{i\to a}(x)$

factor-to-variable message: $\hat{\mu}^{(\ell)}_{a\to i}(x)$

$$f(x_1, x_2, x_3, x_4, x_5, x_6) = f_1(x_1, x_2, x_3) f_2(x_1, x_4, x_6) f_3(x_4) f_4(x_4, x_5)$$

Marginalization via Belief Propagation

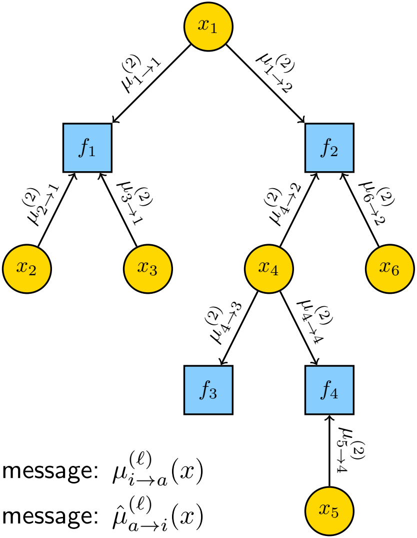


variable-to-factor message: $\mu^{(\ell)}_{i\to a}(x)$

factor-to-variable message: $\hat{\mu}^{(\ell)}_{a\to i}(x)$

$$f(x_1, x_2, x_3, x_4, x_5, x_6) = f_1(x_1, x_2, x_3) f_2(x_1, x_4, x_6) f_3(x_4) f_4(x_4, x_5)$$

Marginalization via Belief Propagation

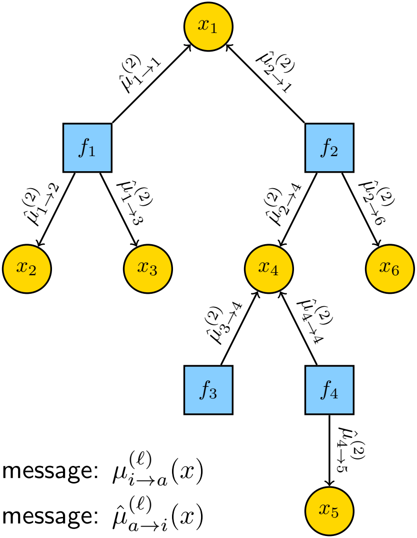


variable-to-factor message: $\mu^{(\ell)}_{i\to a}(x)$

factor-to-variable message: $\hat{\mu}^{(\ell)}_{a\to i}(x)$

$$f(x_1, x_2, x_3, x_4, x_5, x_6) = f_1(x_1, x_2, x_3) f_2(x_1, x_4, x_6) f_3(x_4) f_4(x_4, x_5)$$

Marginalization via Belief Propagation



variable-to-factor message: $\mu^{(\ell)}_{i\to a}(x)$

factor-to-variable message: $\hat{\mu}^{(\ell)}_{a\to i}(x)$

$$f(x_1, x_2, x_3, x_4, x_5, x_6) = f_1(x_1, x_2, x_3) f_2(x_1, x_4, x_6) f_3(x_4) f_4(x_4, x_5)$$

Sudoku Example

	2		5		1		9	
8			2		3			6
	3			6			7	
		1				6		
5	4						1	9
		2				7		
	9			3			8	
2			8		4			7
	1		9		7		6	

Sudoku Example

	2		5		1		9	
8			2		3			6
	3		4	6			7	
		1				6		
5	4						1	9
		2				7		
	9			3			8	
2			8		4			7
	1		9		7		6	

Sudoku Example

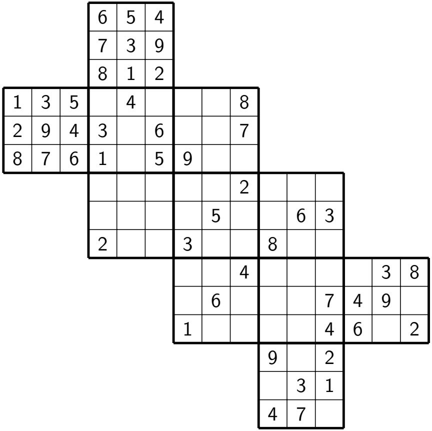
	2		5		1		9	
8			2		3			6
1	3		4	6			7	
		1				6		
5	4						1	9
		2				7		
	9			3			8	
2			8		4			7
	1		9		7		6	

Sudoku Example

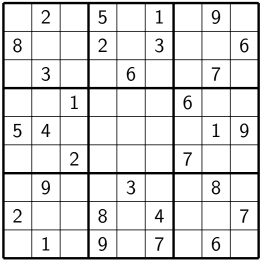
	2		5		1		9	
8			2		3			6
1	3		4	6	8		7	
		1				6		
5	4						1	9
		2				7		
	9			3			8	
2			8		4			7
	1		9		7		6	

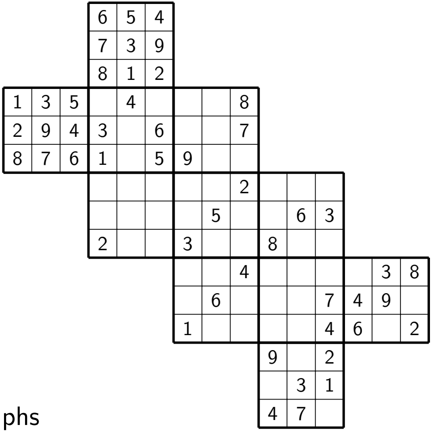
What is Spatial Coupling?

	2		5		1		9	
8			2		3			6
	3			6			7	
		1				6		
5	4						1	9
		2				7		
	9			3			8	
2			8		4			7
	1		9		7		6	



What is Spatial Coupling?





- Spatially-Coupled Factor Graphs
  - Variable nodes have a natural global orientation
  - Boundaries help variables to be recovered in an ordered fashion

Spatially-Coupled Sudoku Example

			6	5	4									
			7	3	9									
			8	1	2									
1	3	5		4				8						
2	9	4	3		6			7						
8	7	6	1		5	9								
								2						
							5			6	3			
			2			3			8					
								4					3	8
							6				7	4	9	
						1					4	6		2
									9		2			
										3	1			
									4	7				

Spatially-Coupled Sudoku Example

			6	5	4									
			7	3	9									
			8	1	2									
1	3	5	9	4	7			8						
2	9	4	3	8	6			7						
8	7	6	1	2	5	9								
								2						
							5			6	3			
			2			3			8					
								4					3	8
							6				7	4	9	
						1					4	6		2
									9		2			
										3	1			
									4	7				

Spatially-Coupled Sudoku Example

			6	5	4									
			7	3	9									
			8	1	2									
1	3	5	9	4	7	6	2	8						
2	9	4	3	8	6	5	1	7						
8	7	6	1	2	5	9	4	3						
								2						
							5			6	3			
			2			3			8					
								4					3	8
							6				7	4	9	
						1					4	6		2
									9		2			
										3	1			
									4	7				

Spatially-Coupled Sudoku Example

			6	5	4									
			7	3	9									
			8	1	2									
1	3	5	9	4	7	6	2	8						
2	9	4	3	8	6	5	1	7						
8	7	6	1	2	5	9	4	3						
			5		3			2						
			4		8		5			6	3			
			2		1	3			8					
								4					3	8
							6				7	4	9	
						1					4	6		2
									9		2			
										3	1			
									4	7				

Spatially-Coupled Sudoku Example

			6	5	4									
			7	3	9									
			8	1	2									
1	3	5	9	4	7	6	2	8						
2	9	4	3	8	6	5	1	7						
8	7	6	1	2	5	9	4	3						
			5		3	4	8	2						
			4		8	7	5	1		6	3			
			2		1	3	9	6	8					
								4					3	8
							6				7	4	9	
						1					4	6		2
									9		2			
										3	1			
									4	7				

Spatially-Coupled Sudoku Example

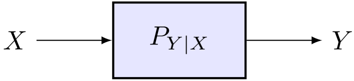
			6	5	4									
			7	3	9									
			8	1	2									
1	3	5	9	4	7	6	2	8						
2	9	4	3	8	6	5	1	7						
8	7	6	1	2	5	9	4	3						
			5	6	3	4	8	2						
			4	9	8	7	5	1		6	3			
			2	7	1	3	9	6	8					
								4					3	8
							6				7	4	9	
						1					4	6		2
									9		2			
										3	1			
									4	7				

Spatially-Coupled Sudoku Example

			6	5	4									
			7	3	9									
			8	1	2									
1	3	5	9	4	7	6	2	8						
2	9	4	3	8	6	5	1	7						
8	7	6	1	2	5	9	4	3						
			5	6	3	4	8	2	7	1	9			
			4	9	8	7	5	1	2	6	3			
			2	7	1	3	9	6	8	4	5			
								4					3	8
							6				7	4	9	
						1					4	6		2
									9		2			
										3	1			
									4	7				

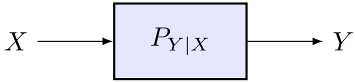
Outline

Capacity of Point-to-Point Communication



- Coding for Discrete-Time Memoryless Channels
  - Transition probability: $P_{Y|X}(y|x)$ for $x \in \mathcal{X}$ and $y \in \mathcal{Y}$
  - Transmit a length-n codeword $\underline{x} \in \mathcal{C} \subset \mathcal{X}^n$
  - Decode to most likely codeword given received $\underline{y}$

Capacity of Point-to-Point Communication



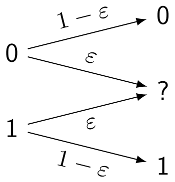
- Coding for Discrete-Time Memoryless Channels
  - Transition probability: $P_{Y|X}(y|x)$ for $x \in \mathcal{X}$ and $y \in \mathcal{Y}$
  - Transmit a length-n codeword $\underline{x} \in \mathcal{C} \subset \mathcal{X}^n$
  - Decode to most likely codeword given received $\underline{y}$

- Channel Capacity introduced by Shannon in 1948
  - Random code of rate $R \triangleq \frac{1}{n} \log_2 |\mathcal{C}|$ (bits per channel use)
  - As $n \to \infty$, reliable transmission possible if $R < C$ with

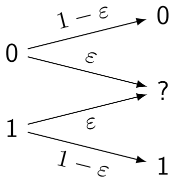
$$C \triangleq \max_{p(x)} I(X;Y)$$

The Binary Erasure Channel (BEC)

- Denoted BEC(ε) when erasure probability is ε
- $C = 1 - \varepsilon =$ expected fraction bits not erased

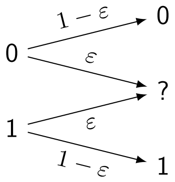


The Binary Erasure Channel (BEC)



- Denoted BEC(ε) when erasure probability is ε
- $C = 1 - \varepsilon =$ expected fraction bits not erased
- Coding with a binary linear code
  - Parity-check matrix $H \in \{0,1\}^{m \times n}$ with $m = (1-R)n$
  - Codebook $\mathcal{C} \triangleq \{\underline{x} \in \{0,1\}^n \mid H\underline{x} = \underline{0}\}$ has 2^{Rn} codewords

The Binary Erasure Channel (BEC)



- Denoted BEC(ε) when erasure probability is ε
- $C = 1 - \varepsilon =$ expected fraction bits not erased

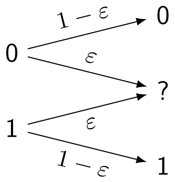
- Coding with a binary linear code
  - Parity-check matrix $H \in \{0,1\}^{m \times n}$ with $m = (1-R)n$
  - Codebook $\mathcal{C} \triangleq \{\underline{x} \in \{0,1\}^n \,|\, H\underline{x} = \underline{0}\}$ has 2^{Rn} codewords

  - Let $\mathcal{E}$ denote the index set of erased positions so that

$$H\underline{x} = \begin{bmatrix} H_{\mathcal{E}} & H_{\mathcal{E}^c} \end{bmatrix} \begin{bmatrix} \underline{x}_{\mathcal{E}} \\ \underline{y}_{\mathcal{E}^c} \end{bmatrix} = \underline{0} \quad \Leftrightarrow \quad H_{\mathcal{E}}\underline{x}_{\mathcal{E}} = -H_{\mathcal{E}^c}\underline{y}_{\mathcal{E}^c}$$

  - Decoding fails iff submatrix $H_{\mathcal{E}}$ is singular

The Binary Erasure Channel (BEC)



- Denoted BEC(ε) when erasure probability is ε
- $C = 1 - \varepsilon =$ expected fraction bits not erased
- Coding with a binary linear code
  - Parity-check matrix $H \in \{0,1\}^{m\times n}$ with $m = (1-R)n$
  - Codebook $\mathcal{C} \triangleq \{\underline{x} \in \{0,1\}^n \,|\, H\underline{x} = \underline{0}\}$ has 2^{Rn} codewords
  - Let $\mathcal{E}$ denote the index set of erased positions so that

$$H\underline{x} = \begin{bmatrix} H_{\mathcal{E}} & H_{\mathcal{E}^c} \end{bmatrix} \begin{bmatrix} \underline{x}_{\mathcal{E}} \\ \underline{y}_{\mathcal{E}^c} \end{bmatrix} = \underline{0} \quad \Leftrightarrow \quad H_{\mathcal{E}} \underline{x}_{\mathcal{E}} = -H_{\mathcal{E}^c} \underline{y}_{\mathcal{E}^c}$$

  - Decoding fails iff submatrix $H_{\mathcal{E}}$ is singular
  - One can achieve capacity by drawing H uniformly at random!

Some Early Milestones in Coding

- 1948: Shannon defines channel capacity and random codes
- 1950: Hamming formalizes linear codes and Hamming distance
- 1954: Reed-Muller codes (Muller gives codes, Reed the decoder)
- 1955: Elias introduces the erasure channel and convolutional codes; also shows random parity-check codes achieve capacity on the BEC
- 1959: BCH Codes (Hocquenghem'59 and Bose-Ray-Chaudhuri'60)
- 1960: Gallager introduces low-density parity-check (LDPC) codes and iterative decoding
- 1960: Reed-Solomon codes

Achieving Capacity in Practice

But, more than 35 years passed before we could:

- Achieve capacity in practice
- Provably achieve capacity with determinstic constructions

Achieving Capacity in Practice

But, more than 35 years passed before we could:

- Achieve capacity in practice
- Provably achieve capacity with determinstic constructions

Modern Milestones:

- 1993: Turbo Codes (Berrou, Glavieux, Thitimajshima)
- 1995: Rediscovery of LDPC codes (MacKay-Neal,Spielman)
- 1997: Optimized irregular LDPC codes for the BEC (LMSSS)
- 2001: Optimized irregular LDPC codes for BMS channels (RSU)
- 2008: Polar codes provable, low-complexity, deterministic (Arikan)
- 1999-2011: Understanding LDPC convolutional codes and coupling

Key Tools That Made the Difference

- Factor Graphs (FGs)
  - Compact description of joint distribution for random variables
  - Natural setup for inference problems with partial observations

Key Tools That Made the Difference

- Factor Graphs (FGs)
  - Compact description of joint distribution for random variables
  - Natural setup for inference problems with partial observations
- Belief-Propagation (BP)
  - Message-passing algorithm for inference on a FG
  - Probability estimates are passed along edges in the factor graph
  - Provides exact marginals if factor graph is a tree

Key Tools That Made the Difference

- Factor Graphs (FGs)
  - Compact description of joint distribution for random variables
  - Natural setup for inference problems with partial observations
- Belief-Propagation (BP)
  - Message-passing algorithm for inference on a FG
  - Probability estimates are passed along edges in the factor graph
  - Provides exact marginals if factor graph is a tree
- Density Evolution (DE)
  - Tracks distribution of messages passed by belief propagation
  - In some cases, allows rigorous analysis of BP-based inference

Key Tools That Made the Difference

- Factor Graphs (FGs)
  - Compact description of joint distribution for random variables
  - Natural setup for inference problems with partial observations
- Belief-Propagation (BP)
  - Message-passing algorithm for inference on a FG
  - Probability estimates are passed along edges in the factor graph
  - Provides exact marginals if factor graph is a tree
- Density Evolution (DE)
  - Tracks distribution of messages passed by belief propagation
  - In some cases, allows rigorous analysis of BP-based inference
- Spatial Coupling (SC): Enables near-optimal performance using BP

Applications of These Tools

- Error-Correcting Codes
  - Random code defined by random factor graph
  - Low-complexity decoding via belief propagation
  - Analysis of belief-propagation decoding via density evolution
  - Provides code constructions that provably achieve capacity!

Applications of These Tools

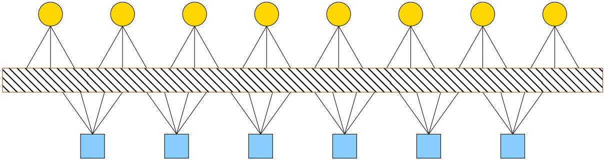
- Error-Correcting Codes
  - Random code defined by random factor graph
  - Low-complexity decoding via belief propagation
  - Analysis of belief-propagation decoding via density evolution
  - Provides code constructions that provably achieve capacity!
- Boolean Satisfiability: K-SAT
  - Random instance of K-SAT defined by random factor graph
  - Non-rigorous analysis via the cavity method
  - Predicted thresholds later proved exact!

Applications of These Tools

- Error-Correcting Codes
  - Random code defined by random factor graph
  - Low-complexity decoding via belief propagation
  - Analysis of belief-propagation decoding via density evolution
  - Provides code constructions that provably achieve capacity!
- Boolean Satisfiability: K-SAT
  - Random instance of K-SAT defined by random factor graph
  - Non-rigorous analysis via the cavity method
  - Predicted thresholds later proved exact!
- Compressed Sensing
  - Random measurement matrix defined by random factor graph
  - Low-complexity reconstruction via message passing
  - Schemes provably achieve the information-theoretic limit!

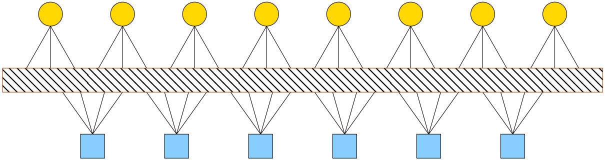
Outline

Low-Density Parity-Check (LDPC) Codes



- Linear codes defined by $\underline{x}H^T = \underline{0}$ for all c.w. $\underline{x} \in \mathcal{C} \subset \{0,1\}^n$
  - H is an $r \times n$ sparse parity-check matrix for the code
  - Code bits and parity checks associated with cols/rows of H

Low-Density Parity-Check (LDPC) Codes



- Linear codes defined by $\underline{x}H^T = \underline{0}$ for all c.w. $\underline{x} \in \mathcal{C} \subset \{0,1\}^n$
  - H is an $r \times n$ sparse parity-check matrix for the code
  - Code bits and parity checks associated with cols/rows of H
- Factor graph: H is the biadjacency matrix for variable/factor nodes
  - Ensemble defined by configuration model for random graphs
  - Checks define factors: $f_{\text{even}}(x_1^d) = \mathbb{I}(x_1 \oplus \cdots \oplus x_d = 0)$
  - Let $x_{F(a)}$ be the x-subvector for the a-th check and

$$f(x_1, \ldots, x_n) = \underbrace{\left(\prod_{a=1}^{r} f_{\text{even}}(x_{F(a)}) \right)}_{\mathbf{1}_{\mathcal{C}}(x_1^n)} \left(\prod_{i=1}^{n} P_{Y|X}(y_i|x_i) \right)$$

A Little History

Robert Gallager

introduced LDPC codes in 1962 paper

1962 *IRE TRANSACTIONS ON INFORMATION THEORY* 21

Low-Density Parity-Check Codes*

R. G. GALLAGER†

Summary—A low-density parity-check code is a code specified by a parity-check matrix with the following properties: each column contains a small fixed number $j \geq 3$ of 1's and each row contains a small fixed number $k > j$ of 1's. The typical minimum distance of these codes increases linearly with block length for a fixed rate and fixed j. When used with maximum likelihood decoding on a sufficiently quiet binary-input symmetric channel, the typical probability of decoding error decreases exponentially with block length for a fixed rate and fixed j.

A simple but nonoptimum decoding scheme operating directly from the channel a posteriori probabilities is described. Both the

equations. We call the set of digits contained in a parity-check equation a parity-check set. For example, the first parity-check set in Fig. 1 is the set of digits (1, 2, 3, 5).

The use of parity-check codes makes coding (as distinguished from decoding) relatively simple to implement. Also, as Elias [3] has shown, if a typical parity-check code of long block length is used on a binary symmetric channel, and if the code rate is between *critical rate* and channel capacity, then the probability of decoding error

Judea Pearl

defined general belief-propagation in 1986 paper

Fusion, Propagation, and Structuring in Belief Networks*

Judea Pearl
Cognitive Systems Laboratory, Computer Science Department, University of California, Los Angeles, CA 90024, U.S.A.

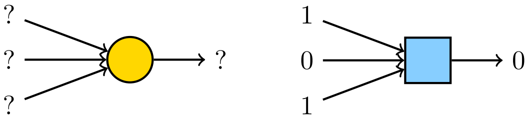
Recommended by Patrick Hayes

ABSTRACT

Belief networks are directed acyclic graphs in which the nodes represent propositions (or variables), the arcs signify direct dependencies between the linked propositions, and the strengths of these dependencies are quantified by conditional probabilities. A network of this sort can be used to represent the generic knowledge of a domain expert, and it turns into a computational architecture if the links are used not merely for storing factual knowledge but also for directing and activating the data flow in the computations which manipulate this knowledge.

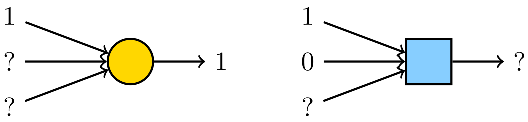
Simple Message-Passing Decoding for the BEC

- Constraint nodes define the valid patterns
  - Circles represent a single value shared by factors
  - Squares assert attached variables sum to 0 mod 2
- Iterative decoding on the binary erasure channel (BEC)
  - Messages passed in phases: bit-to-check and check-to-bit
  - Each output message depends on other input messages
  - Each message is either the correct value or an erasure
- Message passing rules for the BEC
  - Bits pass an erasure only if all other inputs are erased
  - Checks pass the correct value only if all other inputs are correct

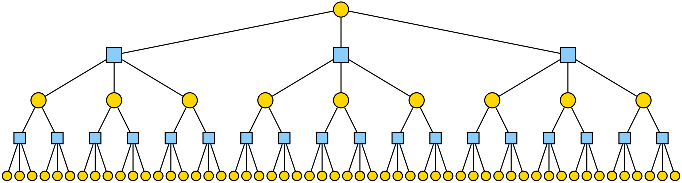


Simple Message-Passing Decoding for the BEC

- Constraint nodes define the valid patterns
  - Circles represent a single value shared by factors
  - Squares assert attached variables sum to 0 mod 2
- Iterative decoding on the binary erasure channel (BEC)
  - Messages passed in phases: bit-to-check and check-to-bit
  - Each output message depends on other input messages
  - Each message is either the correct value or an erasure
- Message passing rules for the BEC
  - Bits pass an erasure only if all other inputs are erased
  - Checks pass the correct value only if all other inputs are correct



Computation Graph and Density Evolution



$$\tilde{x}_3 = \varepsilon y_2^3$$

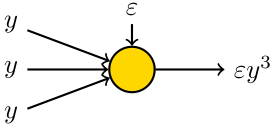
$$y_2 = 1-(1-x_2)^3$$

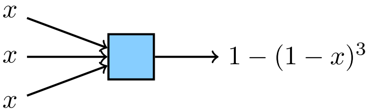
$$x_2 = \varepsilon y_1^2$$

$$y_1 = 1-(1-x_1)^3$$

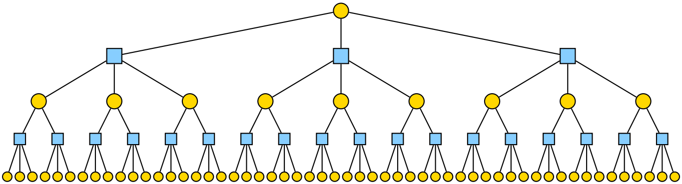
$$x_1 = \varepsilon$$

- Computation graph for a (3,4)-regular LDPC code
  - Illustrates decoding from the perspective of a single bit-node
  - For long random LDPC codes, the graph is typically a tree
  - Allows density evolution to track message erasure probability
  - If x/y are erasure prob. of bit/check output messages, then





Computation Graph and Density Evolution



$$\tilde{x}_3 = \varepsilon y_2^3$$

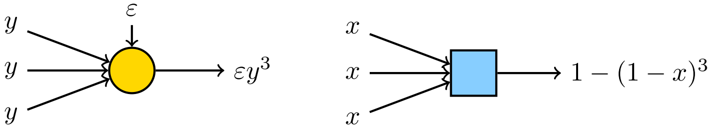
$$y_2 = 1-(1-x_2)^3$$

$$x_2 = \varepsilon y_1^2$$

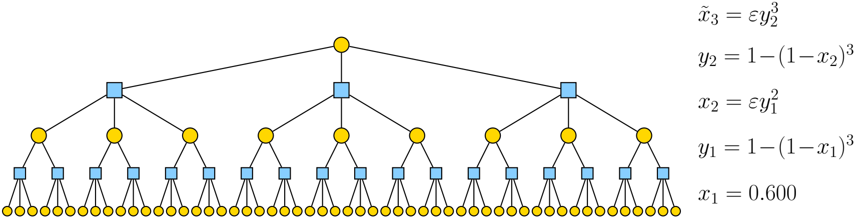
$$y_1 = 1-(1-x_1)^3$$

$$x_1 = \varepsilon$$

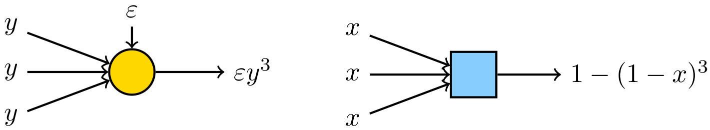
- Computation graph for a (3,4)-regular LDPC code
  - Illustrates decoding from the perspective of a single bit-node
  - For long random LDPC codes, the graph is typically a tree
  - Allows density evolution to track message erasure probability
  - If x/y are erasure prob. of bit/check output messages, then



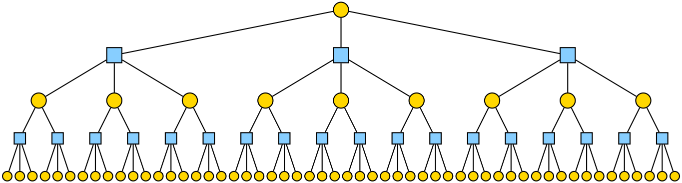
Computation Graph and Density Evolution



- Computation graph for a (3,4)-regular LDPC code
  - Illustrates decoding from the perspective of a single bit-node
  - For long random LDPC codes, the graph is typically a tree
  - Allows density evolution to track message erasure probability
  - If x/y are erasure prob. of bit/check output messages, then



Computation Graph and Density Evolution



$\tilde{x}_3 = \varepsilon y_2^3$

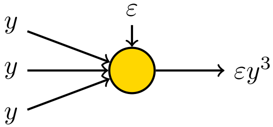
$y_2 = 1-(1-x_2)^3$

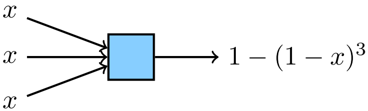
$x_2 = \varepsilon y_1^2$

$y_1 = 0.936$

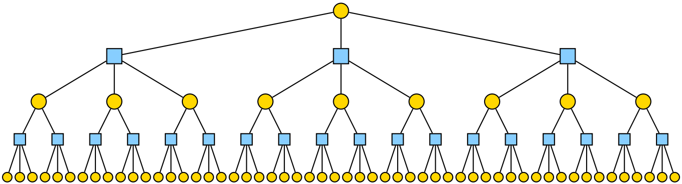
$x_1 = 0.600$

- Computation graph for a (3,4)-regular LDPC code
  - Illustrates decoding from the perspective of a single bit-node
  - For long random LDPC codes, the graph is typically a tree
  - Allows density evolution to track message erasure probability
  - If x/y are erasure prob. of bit/check output messages, then





Computation Graph and Density Evolution



$\tilde{x}_3 = \varepsilon y_2^3$

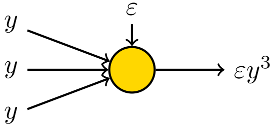
$y_2 = 1-(1-x_2)^3$

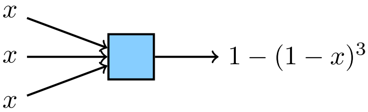
$x_2 = 0.526$

$y_1 = 0.936$

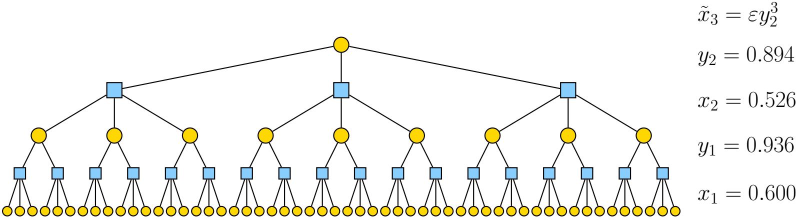
$x_1 = 0.600$

- Computation graph for a (3,4)-regular LDPC code
  - Illustrates decoding from the perspective of a single bit-node
  - For long random LDPC codes, the graph is typically a tree
  - Allows density evolution to track message erasure probability
  - If x/y are erasure prob. of bit/check output messages, then

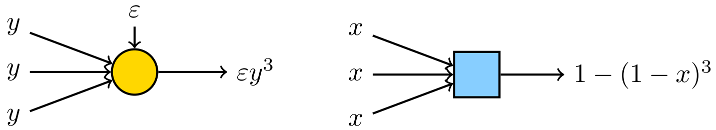




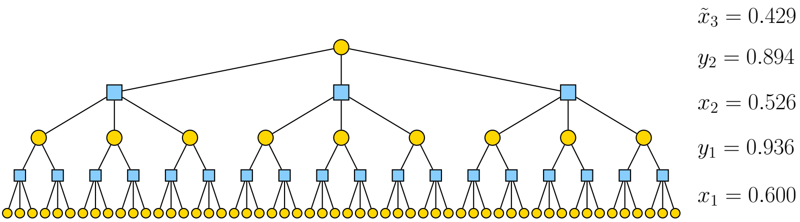
Computation Graph and Density Evolution



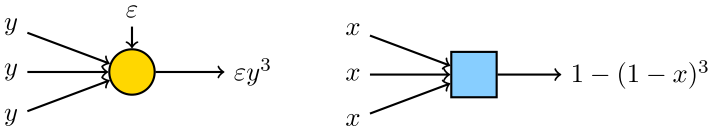
- Computation graph for a (3,4)-regular LDPC code
  - Illustrates decoding from the perspective of a single bit-node
  - For long random LDPC codes, the graph is typically a tree
  - Allows density evolution to track message erasure probability
  - If x/y are erasure prob. of bit/check output messages, then



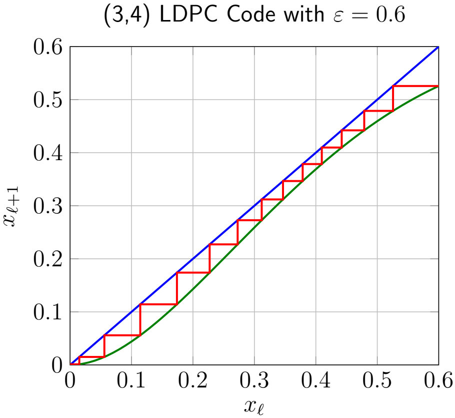
Computation Graph and Density Evolution



- Computation graph for a (3,4)-regular LDPC code
  - Illustrates decoding from the perspective of a single bit-node
  - For long random LDPC codes, the graph is typically a tree
  - Allows density evolution to track message erasure probability
  - If x/y are erasure prob. of bit/check output messages, then



Density Evolution (DE) for LDPC Codes



Density evolution for a $(3,4)$-regular LDPC code:

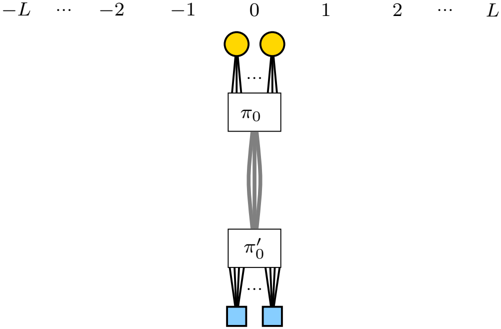
$$x_{\ell+1} = \varepsilon \left(1 - (1 - x_\ell)^3\right)^2$$

Decoding Thresholds:

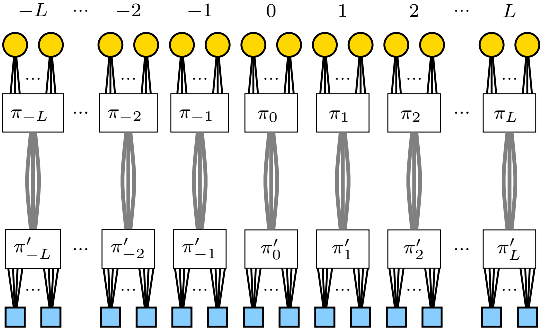
$$\varepsilon^{\mathrm{BP}} \approx 0.647$$
$$\varepsilon^{\mathrm{MAP}} \approx 0.746$$
$$\varepsilon^{\mathrm{Sh}} = 0.750$$

- Binary erasure channel (BEC) with erasure prob. ε
- DE tracks bit-to-check msg erasure rate x_ℓ after ℓ iterations
- Defines noise threshold $\varepsilon^{\mathrm{BP}}$ for the large system limit
  - Easily computed numerically for given code ensemble

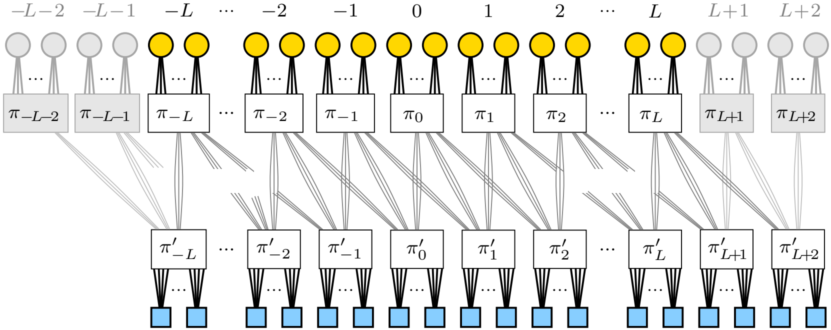
Spatially-Coupled (or Convolutional) LDPC Codes



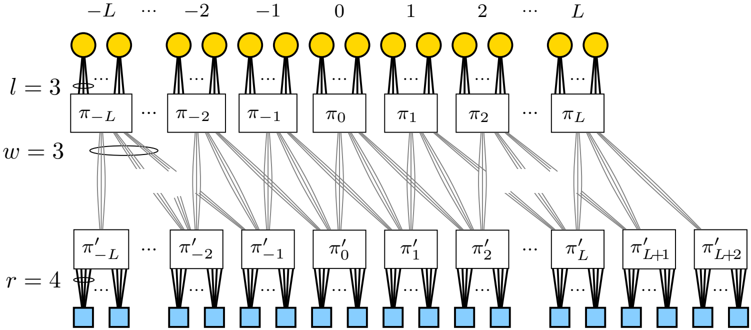
Spatially-Coupled (or Convolutional) LDPC Codes



Spatially-Coupled (or Convolutional) LDPC Codes



Spatially-Coupled (or Convolutional) LDPC Codes



- Historical Notes
  - LDPC convolutional codes introduced in [FZ99]
  - Shown to have near optimal noise thresholds in [LSZC05]
  - (l, r, L, w) ensemble proven to achieve capacity in [KRU11]

The LDPCC Gang

5274 IEEE TRANSACTIONS ON INFORMATION THEORY, VOL. 56, NO. 10, OCTOBER 2010

Iterative Decoding Threshold Analysis for LDPC Convolutional Codes

Michael Lentmaier, *Member, IEEE*, Arvind Sridharan, *Member, IEEE*, Daniel J. Costello, Jr., *Life Fellow, IEEE*, and Kamil Sh. Zigangirov, *Fellow, IEEE*




The Spatial Coupling KRU

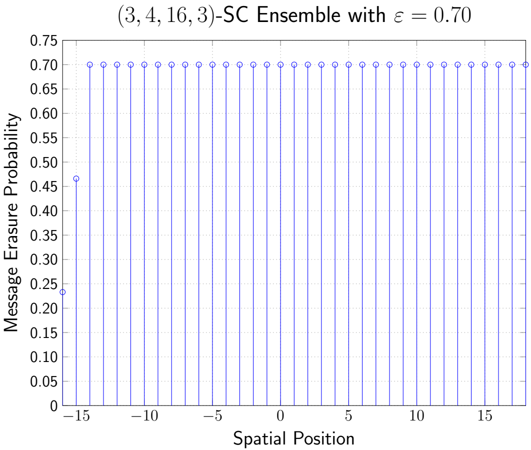
IEEE TRANSACTIONS ON INFORMATION THEORY, VOL. 57, NO. 2, FEBRUARY 2011 803

Threshold Saturation via Spatial Coupling: Why Convolutional LDPC Ensembles Perform So Well over the BEC

Shrinivas Kudekar, *Member, IEEE*, Thomas J. Richardson, *Fellow, IEEE*, and Rüdiger L. Urbanke

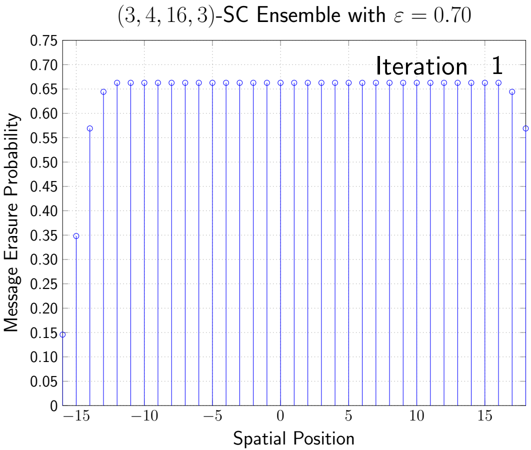


Density Evolution for the (l, r, L, w)-SC LDPC Ensemble



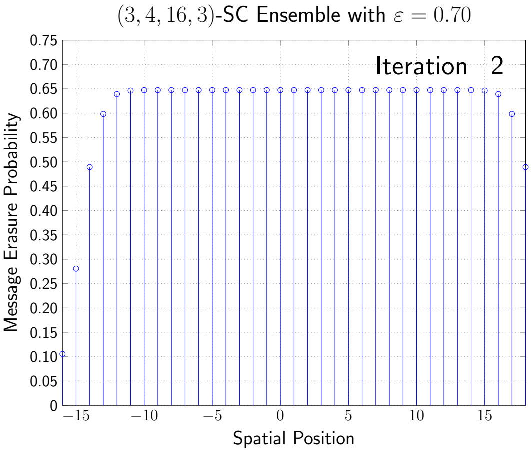
$$z_i^{(\ell+1)} = \varepsilon \left(1 - \frac{1}{w} \sum_{j=0}^{w-1} \left(1 - \frac{1}{w} \sum_{k=0}^{w-1} z_{i+j-k}^{(\ell)} \right)^{r-1} \right)^{l-1}$$

Density Evolution for the (l, r, L, w)-SC LDPC Ensemble



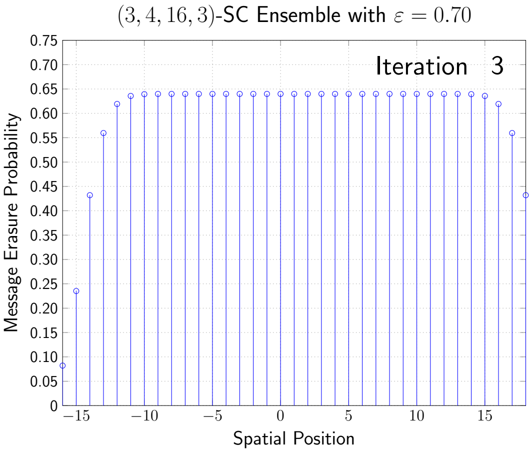
$$z_i^{(\ell+1)} = \varepsilon \left(1 - \frac{1}{w} \sum_{j=0}^{w-1} \left(1 - \frac{1}{w} \sum_{k=0}^{w-1} z_{i+j-k}^{(\ell)} \right)^{r-1} \right)^{l-1}$$

Density Evolution for the (l, r, L, w)-SC LDPC Ensemble



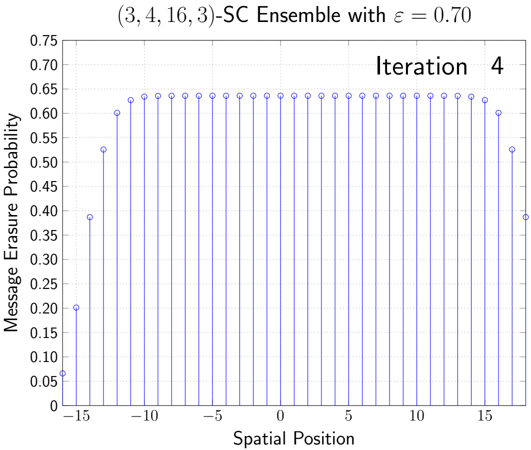
$$z_i^{(\ell+1)} = \varepsilon \left(1 - \frac{1}{w} \sum_{j=0}^{w-1} \left(1 - \frac{1}{w} \sum_{k=0}^{w-1} z_{i+j-k}^{(\ell)}\right)^{r-1}\right)^{l-1}$$

Density Evolution for the (l, r, L, w)-SC LDPC Ensemble



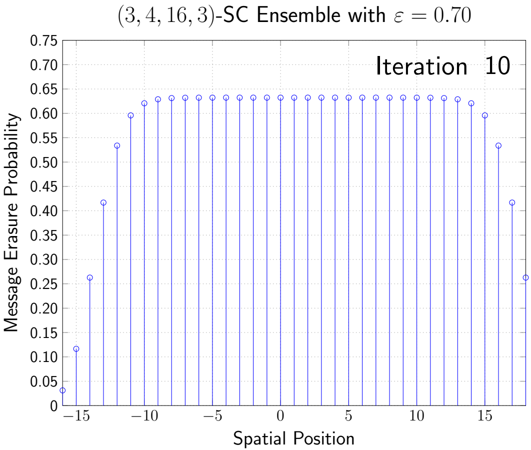
$$z_i^{(\ell+1)} = \varepsilon \left(1 - \frac{1}{w} \sum_{j=0}^{w-1} \left(1 - \frac{1}{w} \sum_{k=0}^{w-1} z_{i+j-k}^{(\ell)} \right)^{r-1} \right)^{l-1}$$

Density Evolution for the (l, r, L, w)-SC LDPC Ensemble



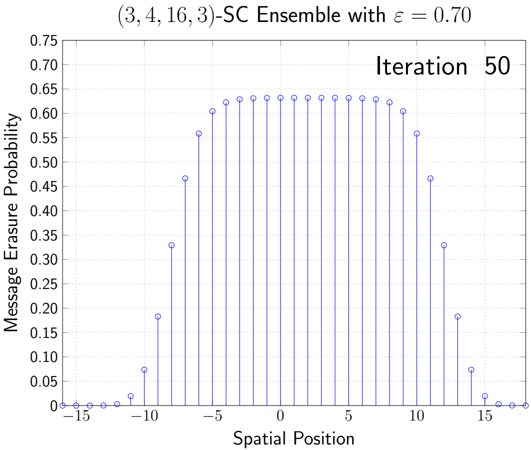
$$z_i^{(\ell+1)} = \varepsilon \left(1 - \frac{1}{w} \sum_{j=0}^{w-1} \left(1 - \frac{1}{w} \sum_{k=0}^{w-1} z_{i+j-k}^{(\ell)}\right)^{r-1}\right)^{l-1}$$

Density Evolution for the (l, r, L, w)-SC LDPC Ensemble



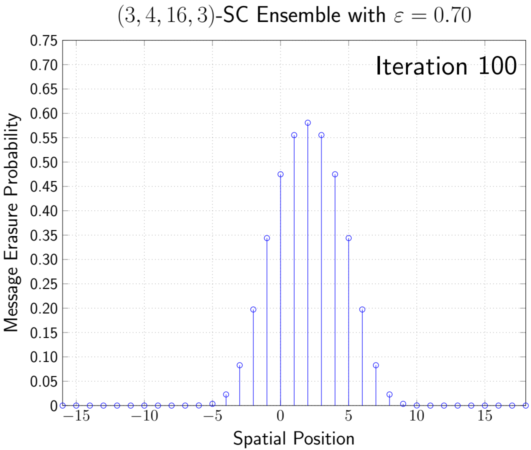
$$z_i^{(\ell+1)} = \varepsilon \left(1 - \frac{1}{w} \sum_{j=0}^{w-1} \left(1 - \frac{1}{w} \sum_{k=0}^{w-1} z_{i+j-k}^{(\ell)}\right)^{r-1}\right)^{l-1}$$

Density Evolution for the (l, r, L, w)-SC LDPC Ensemble



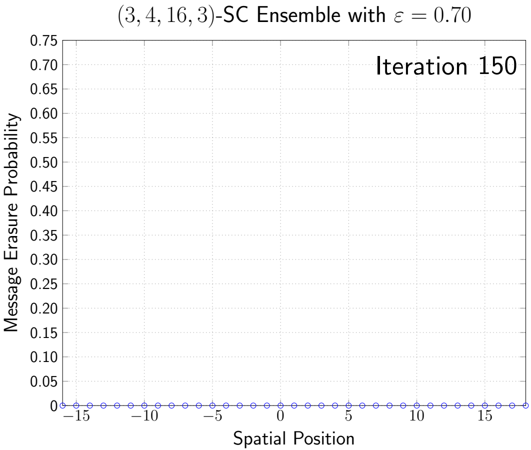
$$z_i^{(\ell+1)} = \varepsilon \left(1 - \frac{1}{w} \sum_{j=0}^{w-1} \left(1 - \frac{1}{w} \sum_{k=0}^{w-1} z_{i+j-k}^{(\ell)} \right)^{r-1} \right)^{l-1}$$

Density Evolution for the (l, r, L, w)-SC LDPC Ensemble



$$z_i^{(\ell+1)} = \varepsilon \left(1 - \frac{1}{w} \sum_{j=0}^{w-1} \left(1 - \frac{1}{w} \sum_{k=0}^{w-1} z_{i+j-k}^{(\ell)} \right)^{r-1} \right)^{l-1}$$

Density Evolution for the (l, r, L, w)-SC LDPC Ensemble



$$z_i^{(\ell+1)} = \varepsilon \left(1 - \frac{1}{w} \sum_{j=0}^{w-1} \left(1 - \frac{1}{w} \sum_{k=0}^{w-1} z_{i+j-k}^{(\ell)} \right)^{r-1} \right)^{l-1}$$

Threshold Saturation via Spatial Coupling

- **General Phenomenon** (observed by Kudekar, Richardson, Urbanke)
  - BP threshold of the spatially-coupled system converges to the MAP threshold of the uncoupled system
  - Can be proven rigorously in many cases!

Threshold Saturation via Spatial Coupling

- **General Phenomenon** (observed by Kudekar, Richardson, Urbanke)
  - BP threshold of the spatially-coupled system converges to the MAP threshold of the uncoupled system
  - Can be proven rigorously in many cases!
- Connection to statistical physics
  - Factor graph defines system of coupled particles
  - Valid sequences are ordered crystalline structures

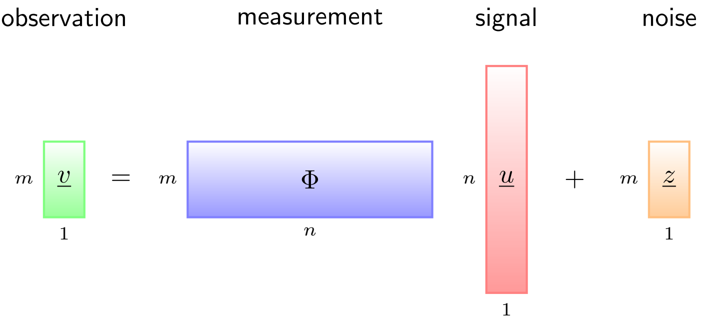
Threshold Saturation via Spatial Coupling

- **General Phenomenon** (observed by Kudekar, Richardson, Urbanke)
  - BP threshold of the spatially-coupled system converges to the MAP threshold of the uncoupled system
  - Can be proven rigorously in many cases!
- Connection to statistical physics
  - Factor graph defines system of coupled particles
  - Valid sequences are ordered crystalline structures
- Between BP and MAP threshold, system acts as supercooled liquid
  - Correct answer (crystalline state) has minimum energy.
  - Spontaneous crystallization (i.e., decoding) does not occur

http://www.youtube.com/watch?v=Xe8vJrIvDQM

Outline

Compressed Sensing (CS)



- For a signal vector in $\underline{u} \in \mathbb{R}^n$ (e.g., drawn iid from $P_U(u)$)
- Let $\Phi \in \mathbb{R}^{m \times n}$ be an $m \times n$ measurement matrix
- Let $\underline{z} \in \mathbb{R}^m$ be a noise vector (e.g., Gaussian noise)
- Problem: Reconstruct $\underline{u}$ from the observation $\underline{v} = \Phi \underline{u} + \underline{z} \in \mathbb{R}^m$

Signal Reconstruction

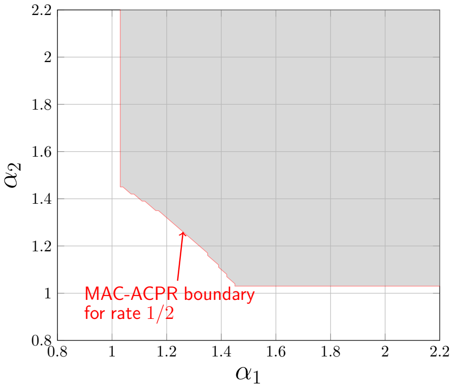
$$p_{\underline{U},\underline{V}}(\underline{u},\underline{v}) = \left(\prod_{i=1}^{m} \exp\left(-\frac{1}{2\sigma^2}\left|v_i - \sum\nolimits_{j=1}^{n} \Phi_{i,j} u_j\right|^2\right)\right)\left(\prod_{j=1}^{n} P_U(u_j)\right)$$

- Joint distribution factors naturally using continuous variables
  - Standard BP defined using pdf messages $\Rightarrow$ impractical
  - Gaussian approx. leads to Relaxed Belief Propagation (RBP)
  - Simplification leads to Approximate Message Passing (AMP)
  - For random Φ, the "density evolution" is called state evolution
- Spatially-Coupled Measurement Matrices
  - Introduced by [KP10] and analyzed by [KMS$^+$12]
  - Are essentially equal to random band-diagonal matrices
  - Shown to be information-theoretically optimal by [DJM13]

Outline

Universality over Unknown Parameters

- The Achievable Channel Parameter Region (ACPR)
  - For a sequence of coding schemes involving parameters, the parameter region where decoding succeeds in the limit
  - In contrast, a capacity region is a rate region for fixed channels



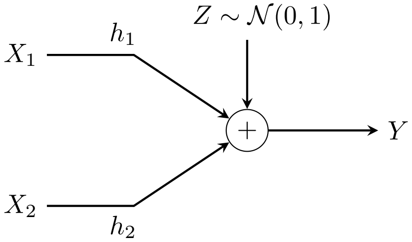
Universality over Unknown Parameters

- The Achievable Channel Parameter Region (ACPR)
  - For a sequence of coding schemes involving parameters, the parameter region where decoding succeeds in the limit
  - In contrast, a capacity region is a rate region for fixed channels
- Properties
  - For fixed encoders, the ACPR depends on the decoders
  - For example, one has BP-ACPR $\subseteq$ MAP-ACPR
  - Often, $\exists$ unique maximal ACPR given by information theory

Universality over Unknown Parameters

- The Achievable Channel Parameter Region (ACPR)
  - For a sequence of coding schemes involving parameters, the parameter region where decoding succeeds in the limit
  - In contrast, a capacity region is a rate region for fixed channels
- Properties
  - For fixed encoders, the ACPR depends on the decoders
  - For example, one has BP-ACPR $\subseteq$ MAP-ACPR
  - Often, $\exists$ unique maximal ACPR given by information theory
- Universality
  - A sequence of encoding/decoding schemes is called universal if:
    its ACPR equals the optimal ACPR
  - Channel parameters are assumed unknown at the transmitter
  - At the receiver, the channel parameters are easily estimated

2-User Binary-Input Gaussian Multiple Access Channel



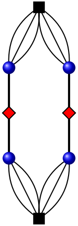
- Fixed noise variance
- Real channel gains h_1 and h_2 not known at transmitter
- Users encode separately with rate-R codes

- MAC-ACPR denotes the information-theoretic optimal region

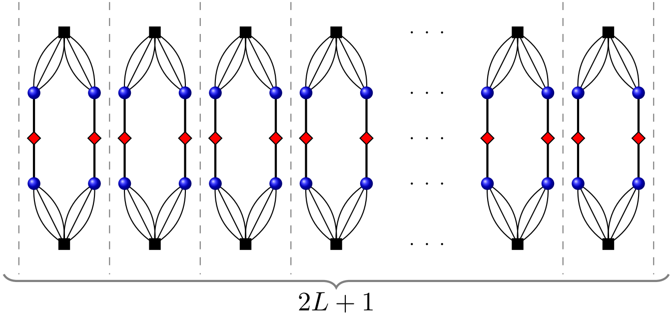
A Little History: SC for Multiple-Access (MAC) Channels

- [KK11] considers a binary-adder erasure channel
  - SC exhibits threshold saturation for the joint decoder
- [YPN11] consider the Gaussian MAC
  - SC exhibits threshold saturation for the joint decoder
  - For channel gains h_1, h_2 unknown at transmitter, SC provides universality
- Others consider CDMA systems without coding
  - [TTK11] shows SC improves BP demod of standard CDMA
  - [ST11] proves saturation for a SC protograph-style CDMA

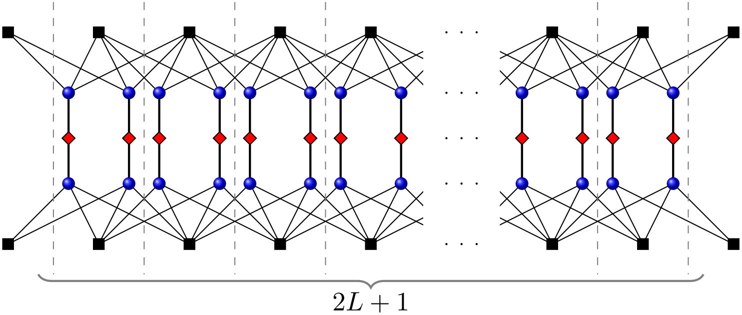
Spatially-Coupled Factor Graph for Joint Decoder



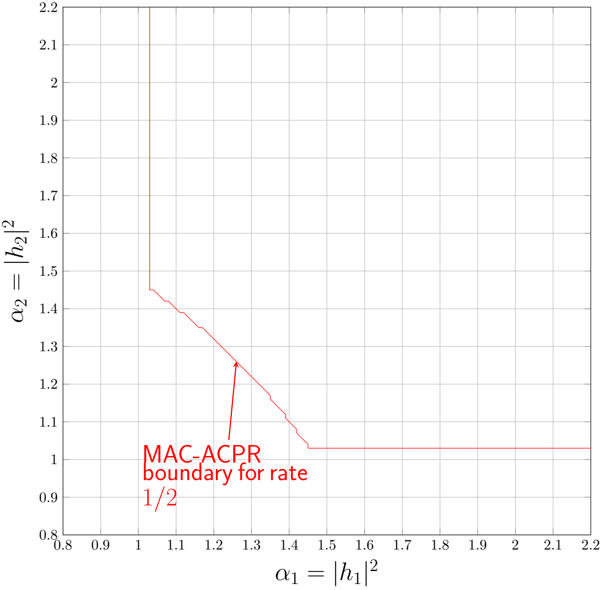
Spatially-Coupled Factor Graph for Joint Decoder



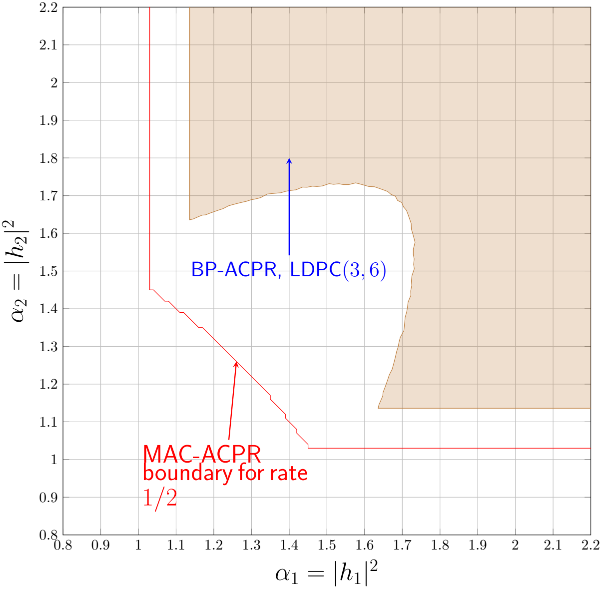
Spatially-Coupled Factor Graph for Joint Decoder



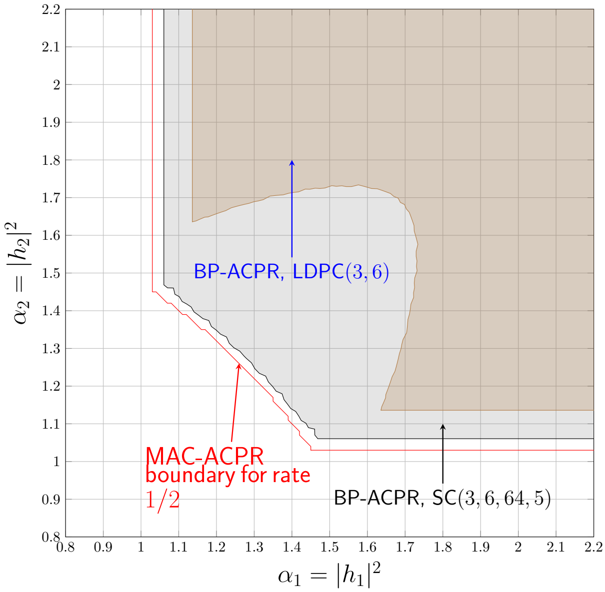
DE Performance of the Joint Decoder



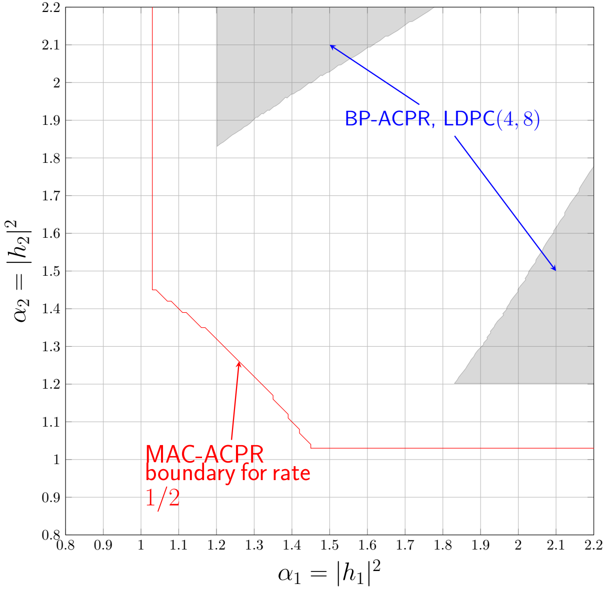
DE Performance of the Joint Decoder



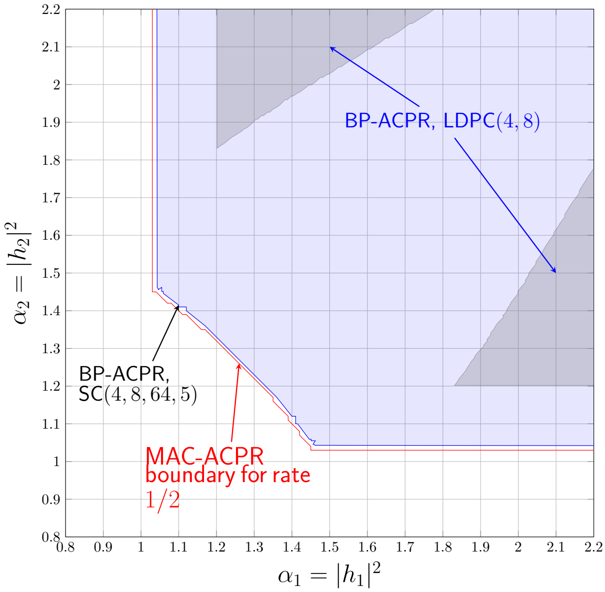
DE Performance of the Joint Decoder



DE Performance of the Joint Decoder

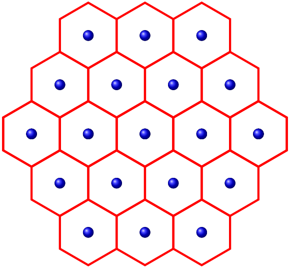


DE Performance of the Joint Decoder



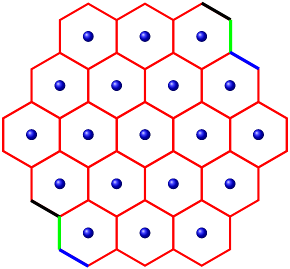
Outline

Cellular Multiple Access: 19-Cell Topology



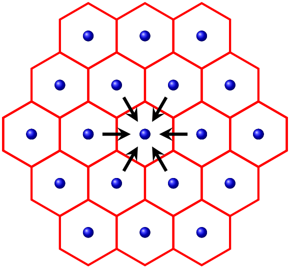
19 cell model

Cellular Multiple Access: 19-Cell Topology



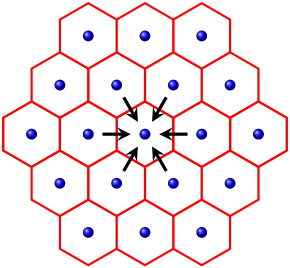
19 cell model
with wraparound

Cellular Multiple Access: 19-Cell Topology



intercell interference

Cellular Multiple Access: 19-Cell Topology



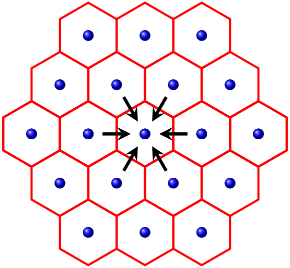
intercell interference

passive interference management

- soft handoff

- user scheduling

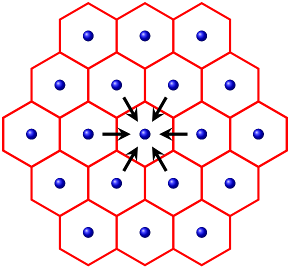
Cellular Multiple Access: 19-Cell Topology



intercell interference

interference-aware multicell coordination and joint decoding

Cellular Multiple Access: 19-Cell Topology

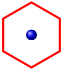


intercell interference

interference-aware multicell coordination and joint decoding

- backhaul links, data center

Cellular Multiple Access: 19-Cell Topology



~~intercell interference~~

interference-aware multicell coordination and joint decoding

- backhaul links, data center

- Ultimate goal is to achieve single-cell performance with optimal power control

Natural Spatial Coupling in Cellular Systems

- Spatially-coupled codes show that
  - Edge effects can make a big difference!
  - Wrap-around may not provide an accurate picture

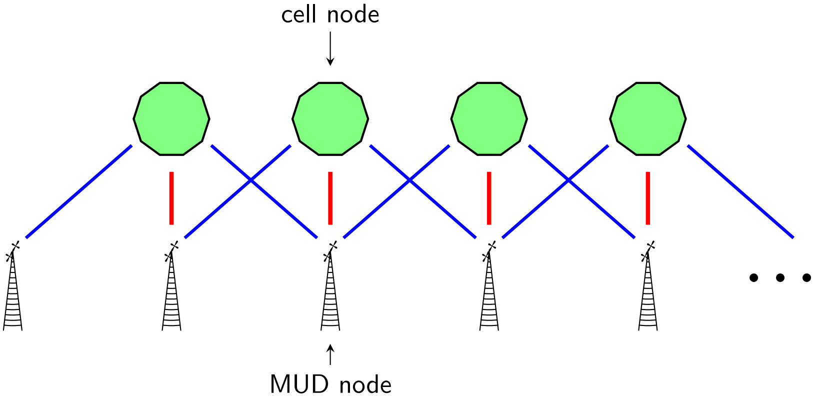
Natural Spatial Coupling in Cellular Systems

- Spatially-coupled codes show that
  - Edge effects can make a big difference!
  - Wrap-around may not provide an accurate picture
- Edge effects are common in real systems
  - Regional and city boundaries
  - Lightly loaded cells due to random loading
  - Periodic scheduling for interference reduction

Natural Spatial Coupling in Cellular Systems

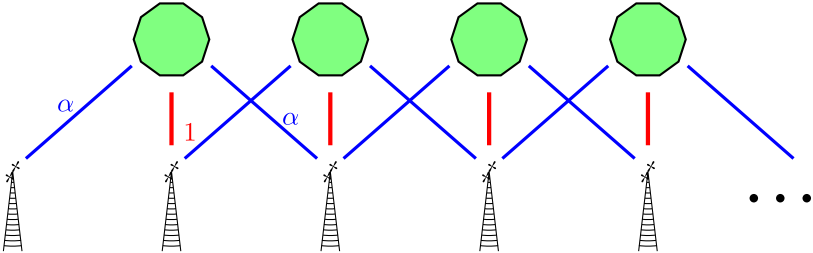
- Spatially-coupled codes show that
  - Edge effects can make a big difference!
  - Wrap-around may not provide an accurate picture
- Edge effects are common in real systems
  - Regional and city boundaries
  - Lightly loaded cells due to random loading
  - Periodic scheduling for interference reduction
- Can this be used to improve cellular systems?
  - Wyner's 1D model: N-chip spreading and $K = \beta N$ users/cell

Example: Wyner's 1D Cellular Model

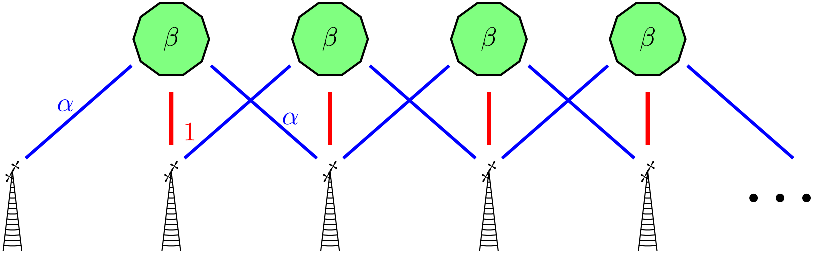


a natural "spatially-coupled" structure

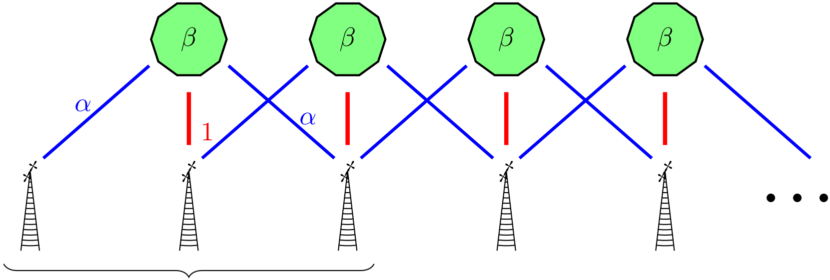
Example: Wyner's 1D Cellular Model



Example: Wyner's 1D Cellular Model

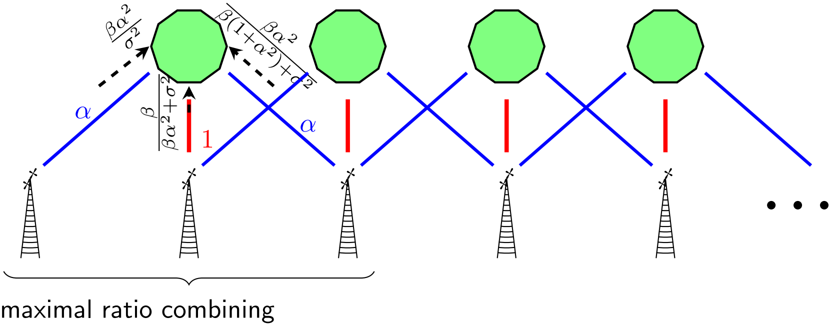


Example: Wyner's 1D Cellular Model

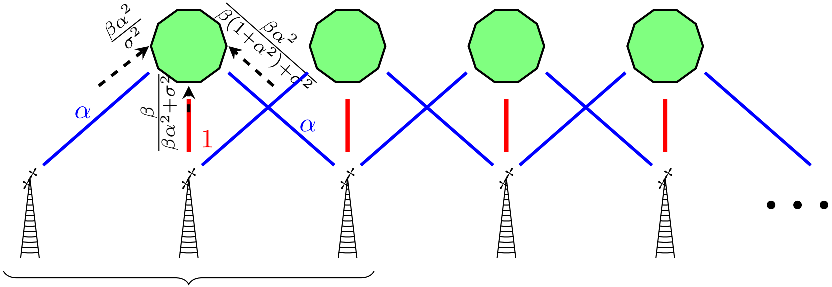


suboptimal decoding

Example: Wyner's 1D Cellular Model



Example: Wyner's 1D Cellular Model

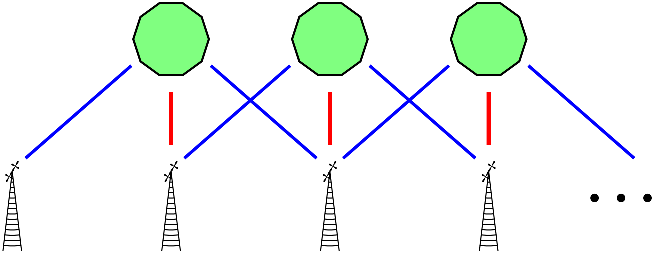


maximal ratio combining +

optimal single cell decoding

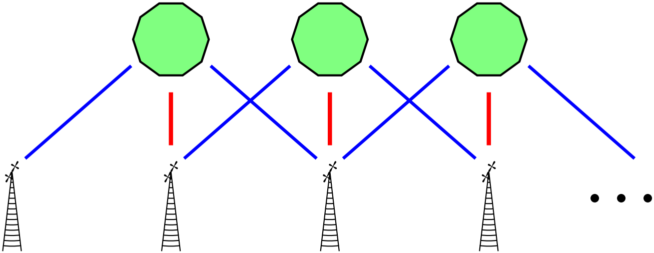
$$\frac{1}{R\beta}\left(\frac{\alpha^2\beta}{\sigma^2}+\frac{\beta}{\beta\alpha^2+\sigma^2}+\frac{\alpha^2\beta}{\beta(1+\alpha^2)+\sigma^2}\right) > \frac{4^R-1}{2R}$$

Example: Wyner's 1D Cellular Model



left-to-right peeling

Example: Wyner's 1D Cellular Model



left-to-right peeling

$\Rightarrow$ lower-bound on system load

Outline

Graphical Models

Point-to-Point Communication

Low-Density Parity-Check Codes

Compressed Sensing

Universality for Multiuser Scenarios

Natural Spatial Coupling in Cellular Systems

Abstract Formulation of Threshold Saturation

Proving Threshold Saturation: General Approach

Let $f\colon \mathcal{X} \to \mathcal{X}$ and $g\colon \mathcal{X} \to \mathcal{X}$ be strictly increasing smooth functions on $\mathcal{X} = [0, 1]$. Then, the scalar recursion (from $x^{(0)} = 1$)

$$y^{(\ell+1)} = g\left(x^{(\ell)}\right)$$
$$x^{(\ell+1)} = f\left(y^{(\ell+1)}\right)$$

Proving Threshold Saturation: General Approach

Let $f\colon \mathcal{X}\to\mathcal{X}$ and $g\colon \mathcal{X}\to\mathcal{X}$ be strictly increasing smooth functions on $\mathcal{X}=[0,1]$. Then, the scalar recursion (from $x^{(0)}=1$)

$$y^{(\ell+1)} = g\left(x^{(\ell)}\right) = 1-(1-x)^3$$
$$x^{(\ell+1)} = f\left(y^{(\ell+1)}\right) = \varepsilon x^2$$

Ex. (3,4) LDPC

Proving Threshold Saturation: General Approach

Let $f\colon \mathcal{X} \to \mathcal{X}$ and $g\colon \mathcal{X} \to \mathcal{X}$ be strictly increasing smooth functions on $\mathcal{X} = [0,1]$. Then, the scalar recursion (from $x^{(0)} = 1$)

$$y^{(\ell+1)} = g\left(x^{(\ell)}\right) = 1 - (1-x)^3$$
$$x^{(\ell+1)} = f\left(y^{(\ell+1)}\right) = \varepsilon x^2$$

Ex. (3,4) LDPC

characterizes fixed point of the coupled recursion ($x_i^{(0)} = 1, i \in [N+w-1]$)

$$y_i^{(\ell+1)} = g\left(x_i^{(\ell)}\right)$$
$$x_i^{(\ell+1)} = \sum_{j=1}^{N+w-1} A_{j,i}\, f\left(\sum_{k=1}^{N} A_{j,k}\, y_k^{(\ell+1)}\right)$$
$$[A_{j,k}] = \boldsymbol{A} = \frac{1}{w}\begin{bmatrix} 1 & 1 & \cdots & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & \ddots & 1 & 0 & 0 \\ 0 & 0 & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & 0 & 1 & 1 & \cdots & 1 \end{bmatrix}$$

Proving Threshold Saturation: General Approach

Let $f\colon \mathcal{X}\to\mathcal{X}$ and $g\colon \mathcal{X}\to\mathcal{X}$ be strictly increasing smooth functions on $\mathcal{X}=[0,1]$. Then, the scalar recursion (from $x^{(0)}=1$)

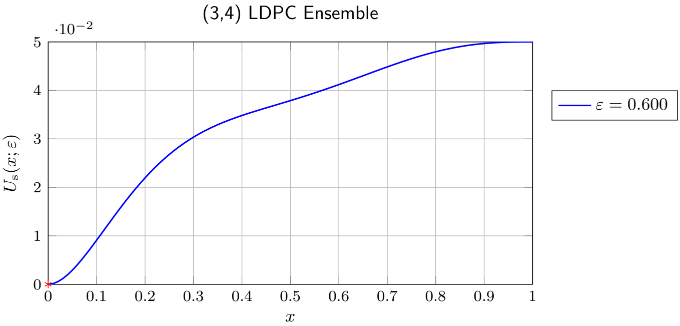
$$y^{(\ell+1)} = g\left(x^{(\ell)}\right) = 1-(1-x)^3$$
$$x^{(\ell+1)} = f\left(y^{(\ell+1)}\right) = \varepsilon x^2$$

Ex. (3,4) LDPC

characterizes fixed point of the coupled recursion ($\boldsymbol{x}^{(0)}=\mathbf{1}$)

$$\boldsymbol{y}^{(\ell+1)} = \boldsymbol{g}\left(\boldsymbol{x}^{(\ell)}\right)$$
$$\boldsymbol{x}^{(\ell+1)} = \boldsymbol{A}^\top \boldsymbol{f}\left(\boldsymbol{A}\,\boldsymbol{y}^{(\ell+1)}\right)$$
$$\boldsymbol{A} = \frac{1}{w}\begin{bmatrix} 1 & 1 & \cdots & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & \ddots & 1 & 0 & 0 \\ 0 & 0 & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & 0 & 1 & 1 & \cdots & 1 \end{bmatrix}$$

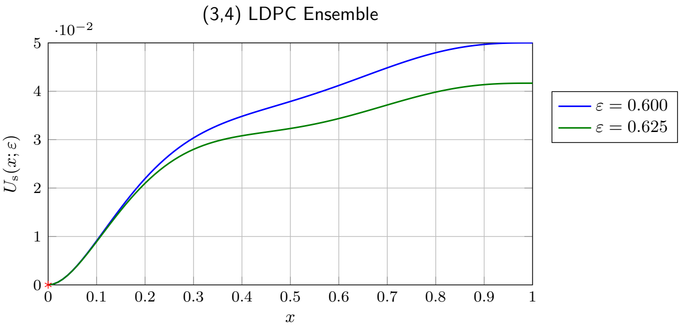
The Potential Function of the Scalar Recursion



Let the potential function $U_{\mathrm{s}} \colon \mathcal{X} \to \mathbb{R}$ of the scalar recursion be

$$U_{\mathrm{s}}(x) \triangleq \int_0^x \big(z - f\left(g(z)\right)\big) g'(z) \mathrm{d}z.$$

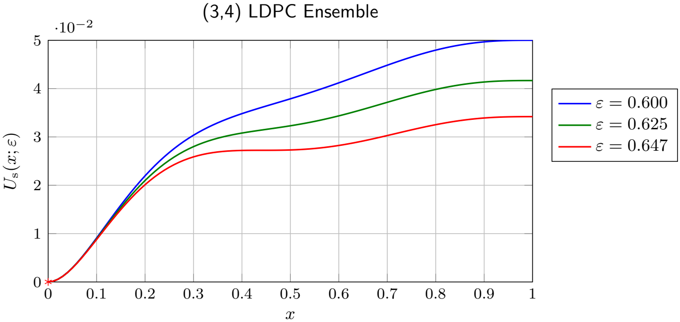
The Potential Function of the Scalar Recursion



Let the potential function $U_s \colon \mathcal{X} \to \mathbb{R}$ of the scalar recursion be

$$U_s(x) \triangleq \int_0^x \big(z - f\left(g(z)\right)\big) g'(z) \mathrm{d}z.$$

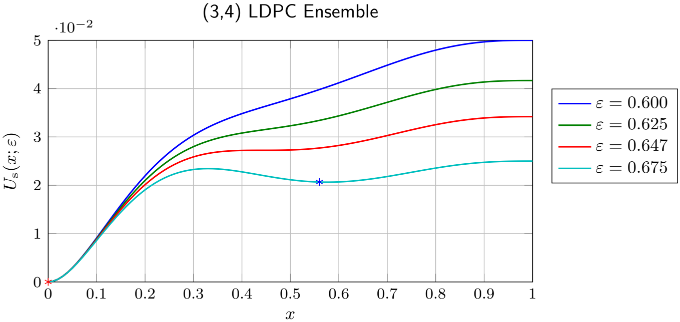
The Potential Function of the Scalar Recursion



Let the potential function $U_{\mathrm{s}} \colon \mathcal{X} \to \mathbb{R}$ of the scalar recursion be

$$U_{\mathrm{s}}(x) \triangleq \int_0^x \big(z - f\left(g(z)\right)\big) g'(z) \mathrm{d}z.$$

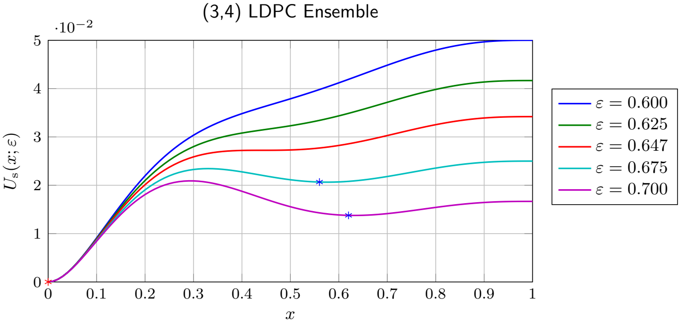
The Potential Function of the Scalar Recursion



Let the potential function $U_s \colon \mathcal{X} \to \mathbb{R}$ of the scalar recursion be

$$U_s(x) \triangleq \int_0^x \big(z - f\,(g(z))\,\big)g'(z)\mathrm{d}z.$$

The Potential Function of the Scalar Recursion



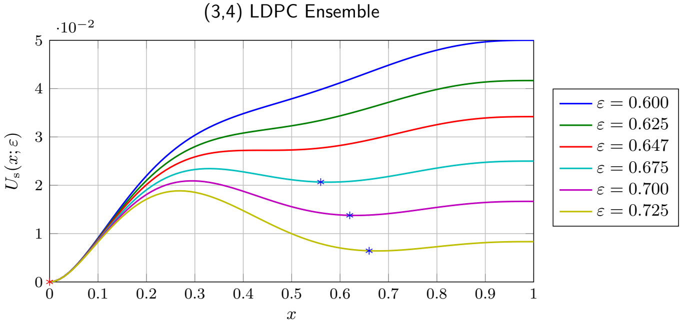
Let the potential function $U_s \colon \mathcal{X} \to \mathbb{R}$ of the scalar recursion be

$$U_s(x) \triangleq \int_0^x \big(z - f\,(g(z))\,\big) g'(z) \mathrm{d}z.$$

Theorem [YJNP14] (arXiv:1309.7910)

$$\lim_{w\to\infty} \lim_{M\to\infty} \max_{i\in\{1,\dots,M\}} x_i^{(\infty)} \leq \max\left(\arg\min_{x\in\mathcal{X}} U_s(x)\right)$$

The Potential Function of the Scalar Recursion



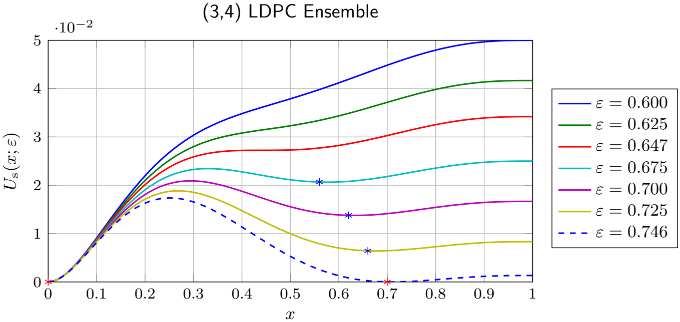
Let the potential function $U_s : \mathcal{X} \to \mathbb{R}$ of the scalar recursion be

$$U_s(x) \triangleq \int_0^x \big(z - f\left(g(z)\right)\big) g'(z) \mathrm{d}z.$$

Theorem [YJNP14] (arXiv:1309.7910)

$$\lim_{w\to\infty} \lim_{M\to\infty} \max_{i\in\{1,\ldots,M\}} x_i^{(\infty)} \leq \max\left(\arg\min_{x\in\mathcal{X}} U_s(x)\right)$$

The Potential Function of the Scalar Recursion



Let the potential function $U_s : \mathcal{X} \to \mathbb{R}$ of the scalar recursion be

$$U_s(x) \triangleq \int_0^x \big(z - f\,(g(z))\,\big) g'(z) \mathrm{d}z.$$

Theorem [YJNP14] (arXiv:1309.7910)

$$\lim_{w\to\infty} \lim_{M\to\infty} \max_{i\in\{1,\ldots,M\}} x_i^{(\infty)} \leq \max\left(\arg\min_{x\in\mathcal{X}} U_s(x)\right)$$

Spatially-Coupled (SC) Compressed Sensing

- Compressive sensing reconstruction of a length-n signal
  - whose entries are i.i.d. copies of a r.v. U with $\mathbb{E}[U^2] < \infty$
  - from δn linear measurements with i.i.d. noise $Z \sim \mathcal{N}(0, \sigma^2)$
  - Assume SC measurements with chain length N and width w

- The MSE x^* for SC measurements with BP reconstruction [DJM13][KMS+12] satisfies (for $M \gg w \to \infty$)

$$x^* \leq \max\left\{ \operatorname*{argmin}_{x \in \mathcal{X}} \left(-\frac{x}{\sigma^2 + \frac{1}{\delta}x} + \delta \ln\left(1 + \frac{x}{\delta\sigma^2}\right) + 2I\left(U; \sqrt{\frac{1}{\sigma^2 + x/\delta}}U + Z\right)\right)\right\}$$

- RHS matches the replica method prediction for the optimal MSE

History of Threshold Saturation Proofs

- the BEC in 2010 [KRU11]
  - Established many properties and tools used by later approaches
- the Curie-Weiss model of physics in 2010 [HMU12]
- CDMA using a GA in 2011 [TTK12]
- CDMA with outer code via GA in 2011 [Tru12]
- compressive sensing using a GA in 2011 [DJM13]
- regular codes on BMS channels in 2012 [KRU13]
- increasing scalar and vector recursions in 2012 [YJNP14]
- irregular LDPC codes on BMS channels in 2012 [KYMP14]
- non-decreasing scalar recursions in 2012 [KRU15]
- non-binary LDPC codes on the BEC in 2014 [AG16]
- and more since 2014...

Summary and Open Problems

- Factor Graphs
  - Useful tool for modeling dependent random variables
  - Low-complexity algorithms for approximate inference
  - Density evolution can be used to analyze performance
- Spatial Coupling
  - Powerful technique for designing and understanding FGs.
  - Related to the statistical physics of supercooled liquids
  - Simple proof of threshold saturation for scalar recursions
- Interesting Open Problems
  - Finding new problems where SC provides benefits
  - Code constructions that reduce the rate-loss due to termination

Thanks for your attention

References I

[AG16] Iryna Andriyanova, Alexandre Graell i Amat. Threshold saturation for nonbinary SC-LDPC codes on the binary erasure channel.
arXiv preprint arXiv:1311.2003v4, 2016.

[DJM13] D.L. Donoho, A. Javanmard, A. Montanari.
Information-theoretically optimal compressed sensing via spatial coupling and approximate message passing.
IEEE Trans. Inform. Theory, 59(11):7434–7464, Nov. 2013.

[FZ99] J. Felstrom, K. S. Zigangirov.
Time-varying periodic convolutional codes with low-density parity-check matrix.
IEEE Trans. Inform. Theory, 45(6):2181–2191, Sept. 1999.

[Gal63] Robert G. Gallager.
Low-Density Parity-Check Codes.
The M.I.T. Press, Cambridge, MA, USA, 1963.

[HMU12] S. H. Hassani, N. Macris, R. Urbanke.
Chains of mean-field models.
J. Stat. Mech., strona P02011, 2012.

[KFL01] Frank R. Kschischang, Brendan J. Frey, Hans-Andrea Loeliger.
Factor graphs and the sum-product algorithm.
IEEE Trans. Inform. Theory, 47(2):498–519, Feb. 2001.

References II

[KK11] S. Kudekar, K. Kasai.
Spatially coupled codes over the multiple access channel.
Proc. IEEE Int. Symp. Inform. Theory, strony 2816–2820, St. Petersburg, Russia, July 2011.

[KMS+12] Florent Krzakala, Marc Mézard, Francois Sausset, Yifan Sun, Lenka Zdeborová.
Probabilistic reconstruction in compressed sensing: algorithms, phase diagrams, and threshold achieving matrices.
Journal of Statistical Mechanics: Theory and Experiment, (08):P08009, 2012.

[KP10] S. Kudekar, H. D. Pfister.
The effect of spatial coupling on compressive sensing.
Proc. Annual Allerton Conf. on Commun., Control, and Comp., strony 347–353, Monticello, IL, USA, Oct. 2010.

[KRU11] S. Kudekar, T. J. Richardson, R. L. Urbanke.
Threshold saturation via spatial coupling: Why convolutional LDPC ensembles perform so well over the BEC.
IEEE Trans. Inform. Theory, 57(2):803–834, Feb. 2011.

References III

[KRU13] S. Kudekar, T. Richardson, R. L. Urbanke.
Spatially coupled ensembles universally achieve capacity under belief propagation.
IEEE Trans. Inform. Theory, 59(12):7761–7813, Dec. 2013.

[KRU15] Shrinivas Kudekar, Thomas J Richardson, Rudiger L Urbanke.
Wave-like solutions of general 1-D spatially coupled systems.
IEEE Trans. Inform. Theory, 61(8):4117–4157, 2015.

[KYMP14] Santhosh Kumar, Andrew J. Young, Nicolas Macris, Henry D. Pfister.
Threshold saturation for spatially-coupled LDPC and LDGM codes on BMS channels.
IEEE Trans. Inform. Theory, 60(12):7389–7415, Dec. 2014.

[LMSS01] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, D. A. Spielman.
Efficient erasure correcting codes.
IEEE Trans. Inform. Theory, 47(2):569–584, Feb. 2001.

[LSZC05] M. Lentmaier, A. Sridharan, K. S. Zigangirov, D. J. Costello.
Terminated LDPC convolutional codes with thresholds close to capacity.
Proc. IEEE Int. Symp. Inform. Theory, strony 1372–1376, Adelaide, Australia, Sept. 2005.

References IV

[Mac99] David J. C. MacKay.
Good error-correcting codes based on very sparse matrices.
IEEE Trans. Inform. Theory, 45(2):399–431, March 1999.

[MM09] M. Mezard, A. Montanari.
Information, Physics, and Computation.
Oxford University Press, New York, NY, 2009.

[Pea88] Judea Pearl.
Probabilistic reasoning in intelligent systems: Networks of plausible reasoning, 1988.

[RSU01] Thomas J. Richardson, M. Amin Shokrollahi, Rüdiger L. Urbanke.
Design of capacity-approaching irregular low-density parity-check codes.
IEEE Trans. Inform. Theory, 47(2):619–637, Feb. 2001.

[RU01] Thomas J. Richardson, Rüdiger L. Urbanke.
The capacity of low-density parity-check codes under message-passing decoding.
IEEE Trans. Inform. Theory, 47(2):599–618, Feb. 2001.

[RU08] Thomas J. Richardson, Rüdiger L. Urbanke.
Modern Coding Theory.
Cambridge University Press, New York, NY, 2008.

References V

[ST11] Christian Schlegel, Dmitri Truhachev.
Multiple access demodulation in the lifted signal graph with spatial coupling.
Proc. IEEE Int. Symp. Inform. Theory, strony 2989–2993, St. Petersburg, Russia, July 2011.

[Tru12] Dmitri Truhachev.
Achieving AWGN multiple access channel capacity with spatial graph coupling.
IEEE Commun. Letters, 16(5):585–588, May 2012.

[TTK11] K. Takeuchi, T. Tanaka, T. Kawabata.
Improvement of BP-based CDMA multiuser detection by spatial coupling.
Proc. IEEE Int. Symp. Inform. Theory, strony 1489–1493, St. Petersburg, Russia, July 2011.

[TTK12] Keigo Takeuchi, Toshiyuki Tanaka, Tsutomu Kawabata.
A phenomenological study on threshold improvement via spatial coupling.
IEICE Trans. Fundamentals, E95-A(5):974–977, 2012.

[YJNP14] A. Yedla, Y.-Y. Jian, P. S. Nguyen, H. D. Pfister.
A simple proof of Maxwell saturation for coupled scalar recursions.
IEEE Trans. Inform. Theory, 60(11):6943–6965, Nov. 2014.

References VI

[YPN11] A. Yedla, H. D. Pfister, K. R. Narayanan.
Universality for the noisy Slepian-Wolf problem via spatial coupling.
Proc. IEEE Int. Symp. Inform. Theory, strony 2567–2571, St. Petersburg, Russia, July 2011.