Three Lectures on Loop Calculus Approach to Graphical Models of Statistical Inference

Michael Chertkov

Center for Nonlinear Studies & Theory Division, LANL

June 18-20, 2007
ITH, Zurich

Thanks to V. Chernyak (Wayne State, Detroit) & M. Stepanov (UofA, Tucson)

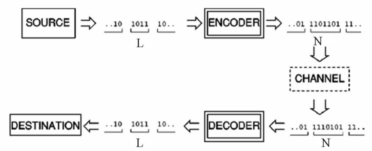
Outline

1. First Lecture
   - Error Correction. Statistical Inference.
   - Bethe Free Energy and Belief Propagation (BP)
   - Loop Calculus: Gauge Transformations & Loop Series
2. Second Lecture
   - Error-Floor. Pseudo-Codewords and Instantons.
   - Pseudo-Codeword Search. Spectra.
   - Analysis and Improvement of Decoding with Loop Calculus
3. Third Lecture
   - Loop Calculus for q-ary alphabet: Loop Tower
   - Long Correlations and Loops in Statistical Mechanics
   - Graphical Transformations

Error Correction

Cell phone | Hard disk | Optical disk | Fiber

Scheme:



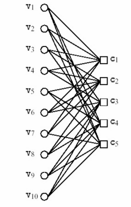
Example of Additive White Gaussian Channel:

$$P(\mathbf{x}_{out}|\mathbf{x}_{in}) = \prod_{i=bits} p(x_{out;i}|x_{in;i})$$

$$p(x|y) \sim \exp(-s^2(x-y)^2/2)$$

- Channel
  is noisy "black box" with only statistical information available
- Encoding:
  use redundancy to redistribute damaging effect of the noise
- Decoding [Algorithm]:
  reconstruct most probable codeword by noisy (polluted) channel

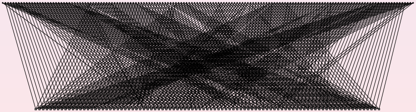
Low Density Parity Check Codes

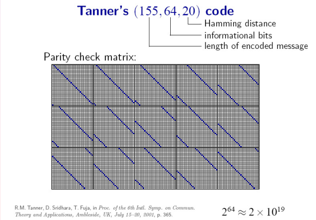


- N bits, M checks, $L = N - M$ information bits
  example: $N = 10$, $M = 5$, $L = 5$
- 2^L codewords of 2^N possible patterns
- Parity check: $\hat{H}\mathbf{v} = \mathbf{c} = \mathbf{0}$
  example:

$$\hat{H} = \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}$$

- LDPC = graph (parity check matrix) is sparse





Statistical Models

Ising model $\sigma_i = \pm 1$

$$\mathcal{P}(\boldsymbol{\sigma}) = Z^{-1} \exp\left(\sum_{i,j} J_{ij}\sigma_i\sigma_j\right)$$

J_{ij} define the graph (lattice)

Decoding $\sigma_i = \pm 1$

$$\mathcal{P}(\boldsymbol{\sigma}|\mathbf{x}) = Z^{-1}(\mathbf{x})\prod_{\alpha}\delta\left(\prod_{i\in\alpha}\sigma_i, +1\right)\prod_i p(x_i|\sigma_i)$$

Hard (check) constraints define the graph/code

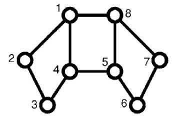
N.Sourlas '89; A.Montanari '00: Error-correction as a Statistical Mechanics

Graphical models

Factorization (Forney '01, Loeliger '01)

$$\mathcal{P}(\boldsymbol{\sigma}|\mathbf{x}) = Z^{-1}\prod_a f_a(\mathbf{x}_a|\boldsymbol{\sigma}_a)$$

$$\underbrace{Z(\mathbf{x}) = \sum_{\boldsymbol{\sigma}}\prod_a f_a(\mathbf{x}_a|\boldsymbol{\sigma}_a))}_{\text{partition function}}$$

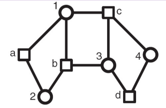


$$f_a \geq 0$$
$$\sigma_{ab} = \sigma_{ba} = \pm 1$$
$$\boldsymbol{\sigma}_1 = (\sigma_{12}, \sigma_{14}, \sigma_{18})$$
$$\boldsymbol{\sigma}_2 = (\sigma_{12}, \sigma_{13})$$

Example: Error-Correction (linear code, bipartite Tanner graph)

$$f_i(h_i|\boldsymbol{\sigma}_i) = \exp(\sigma_i h_i)\cdot\begin{cases} 1, & \forall \alpha,\beta \ni i, \quad \sigma_{i\alpha} = \sigma_{i\beta} \\ 0, & \text{otherwise} \end{cases}$$

$$f_\alpha(\boldsymbol{\sigma}_\alpha) = \delta\left(\prod_{i\in\alpha}\sigma_i, +1\right)$$



h_i - log-likelihoods

Statistical Inference

$\boldsymbol{\sigma}_{\text{orig}}$	$\Rightarrow$	$\mathbf{x}$	$\Rightarrow$	$\boldsymbol{\sigma}$
original *data* $\boldsymbol{\sigma}_{\text{orig}} \in \mathcal{C}$ codeword	noisy channel $\mathcal{P}(\mathbf{x}\|\boldsymbol{\sigma})$	corrupted data: log-likelihood magnetic field	statistical inference	possible preimage $\boldsymbol{\sigma} \in \mathcal{C}$

Maximum Likelihood [ground state] Maximum-a-Posteriori [magnetization]

$$\text{ML} = \arg\max_{\boldsymbol{\sigma}} \mathcal{P}(\mathbf{x}|\boldsymbol{\sigma}) \qquad \text{MAP}_i = \arg\max_{\sigma_i} \sum_{\boldsymbol{\sigma}\setminus\sigma_i} \mathcal{P}(\mathbf{x}|\boldsymbol{\sigma})$$

Exhaustive search is generally expensive:
complexity of the algorithm $\sim 2^N$

Statistical Inference

$\boldsymbol{\sigma}_{\text{orig}}$	$\Rightarrow$	$\mathbf{x}$	$\Rightarrow$	$\boldsymbol{\sigma}$
original *data* $\boldsymbol{\sigma}_{\text{orig}} \in \mathcal{C}$ codeword	noisy channel $\mathcal{P}(\mathbf{x}\|\boldsymbol{\sigma})$	corrupted data: log-likelihood magnetic field	statistical inference	possible preimage $\boldsymbol{\sigma} \in \mathcal{C}$

Maximum Likelihood [ground state] Maximum-a-Posteriori [magnetization]

$$\text{ML} = \arg\max_{\boldsymbol{\sigma}} \mathcal{P}(\mathbf{x}|\boldsymbol{\sigma}) \qquad \text{MAP}_i = \arg\max_{\sigma_i} \sum_{\boldsymbol{\sigma}\setminus\sigma_i} \mathcal{P}(\mathbf{x}|\boldsymbol{\sigma})$$

Exhaustive search is generally expensive:
complexity of the algorithm $\sim 2^N$

Statistical Inference

σ_{orig}	$\Rightarrow$	$\mathbf{x}$	$\Rightarrow$	$\boldsymbol{\sigma}$
original *data* $\sigma_{\text{orig}} \in \mathcal{C}$ codeword	noisy channel $\mathcal{P}(\mathbf{x}\|\boldsymbol{\sigma})$	corrupted data: log-likelihood magnetic field	statistical inference	possible preimage $\boldsymbol{\sigma} \in \mathcal{C}$

Maximum Likelihood [ground state] Maximum-a-Posteriori [magnetization]

$$\text{ML} = \arg\max_{\boldsymbol{\sigma}} \mathcal{P}(\mathbf{x}|\boldsymbol{\sigma}) \qquad \text{MAP}_i = \arg\max_{\sigma_i} \sum_{\boldsymbol{\sigma}\setminus\sigma_i} \mathcal{P}(\mathbf{x}|\boldsymbol{\sigma})$$

Exhaustive search is generally expensive:
complexity of the algorithm $\sim 2^N$

Statistical Inference

$\boldsymbol{\sigma}_{\text{orig}}$	$\Rightarrow$	$\mathbf{x}$	$\Rightarrow$	$\boldsymbol{\sigma}$
original *data* $\boldsymbol{\sigma}_{\text{orig}} \in \mathcal{C}$ codeword	noisy channel $\mathcal{P}(\mathbf{x}\|\boldsymbol{\sigma})$	corrupted data: log-likelihood magnetic field	statistical inference	possible preimage $\boldsymbol{\sigma} \in \mathcal{C}$

$$\boldsymbol{\sigma} = (\sigma_1, \cdots, \sigma_N), \quad N \textit{ finite}, \quad \sigma_i = \pm 1 \ (\textit{example})$$

Maximum Likelihood [ground state]
Maximum-a-Posteriori [magnetization]

$$\text{ML} = \arg\max_{\boldsymbol{\sigma}} \mathcal{P}(\mathbf{x}|\boldsymbol{\sigma}) \qquad \text{MAP}_i = \arg\max_{\sigma_i} \sum_{\boldsymbol{\sigma}\setminus\sigma_i} \mathcal{P}(\mathbf{x}|\boldsymbol{\sigma})$$

Exhaustive search is generally expensive:
complexity of the algorithm $\sim 2^N$

Variational Method in Statistical Mechanics

$$P(\boldsymbol{\sigma}) = \frac{\prod_a f_a(\boldsymbol{\sigma}_a)}{Z}, \quad Z \equiv \sum_{\boldsymbol{\sigma}} \prod_a f_a(\boldsymbol{\sigma}_a)$$

Exact Variational Principle — Kullback-Leibler '51

$$F\{b(\boldsymbol{\sigma})\} = -\sum_{\boldsymbol{\sigma}} b(\boldsymbol{\sigma}) \sum_a \ln f_a(\boldsymbol{\sigma}_a) + \sum_{\boldsymbol{\sigma}} b(\boldsymbol{\sigma}) \ln b(\boldsymbol{\sigma})$$

$$\left.\frac{\delta F}{\delta b(\boldsymbol{\sigma})}\right|_{b(\boldsymbol{\sigma})=p(\boldsymbol{\sigma})} = 0 \quad \text{under} \quad \sum_{\boldsymbol{\sigma}} b(\boldsymbol{\sigma}) = 1$$

Variational Ansatz

- Mean-Field: $p(\boldsymbol{\sigma}) \approx b(\boldsymbol{\sigma}) = \prod_i b_i(\sigma_i)$
- Belief Propagation:

$$p(\boldsymbol{\sigma}) \approx b(\boldsymbol{\sigma}) = \frac{\prod_a b_a(\boldsymbol{\sigma}_a)}{\prod_{(a,b)} b_{ab}(\sigma_{ab})} \quad \text{(exact on a tree)}$$

$$b_a(\boldsymbol{\sigma}_a) = \sum_{\boldsymbol{\sigma}\setminus\boldsymbol{\sigma}_a} b(\boldsymbol{\sigma}), \quad b_{ab}(\sigma_{ab}) = \sum_{\boldsymbol{\sigma}\setminus\sigma_{ab}} b(\boldsymbol{\sigma})$$

Variational Method in Statistical Mechanics

$$P(\boldsymbol{\sigma})=\frac{\prod_a f_a(\boldsymbol{\sigma}_a)}{Z},\quad Z\equiv\sum_{\boldsymbol{\sigma}}\prod_a f_a(\boldsymbol{\sigma}_a)$$

Exact Variational Principle — Kullback-Leibler '51

$$F\{b(\boldsymbol{\sigma})\}=-\sum_{\boldsymbol{\sigma}} b(\boldsymbol{\sigma})\sum_a \ln f_a(\boldsymbol{\sigma}_a)+\sum_{\boldsymbol{\sigma}} b(\boldsymbol{\sigma})\ln b(\boldsymbol{\sigma})$$

$$\left.\frac{\delta F}{\delta b(\boldsymbol{\sigma})}\right|_{b(\boldsymbol{\sigma})=p(\boldsymbol{\sigma})}=0 \quad \text{under} \quad \sum_{\boldsymbol{\sigma}} b(\boldsymbol{\sigma})=1$$

Variational Ansatz

- Mean-Field: $p(\boldsymbol{\sigma})\approx b(\boldsymbol{\sigma})=\prod_i b_i(\sigma_i)$
- Belief Propagation:

$$p(\boldsymbol{\sigma})\approx b(\boldsymbol{\sigma})=\frac{\prod_a b_a(\boldsymbol{\sigma}_a)}{\prod_{(a,b)} b_{ab}(\sigma_{ab})} \quad \text{(exact on a tree)}$$

$$b_a(\boldsymbol{\sigma}_a)=\sum_{\boldsymbol{\sigma}\setminus\boldsymbol{\sigma}_a} b(\boldsymbol{\sigma}),\quad b_{ab}(\sigma_{ab})=\sum_{\boldsymbol{\sigma}\setminus\sigma_{ab}} b(\boldsymbol{\sigma})$$

Bethe free energy: variational approach
(Yedidia,Freeman,Weiss '01 - inspired by Bethe '35, Peierls '36)

$$F = \underbrace{-\sum_a \sum_{\boldsymbol{\sigma}_a} b_a(\boldsymbol{\sigma}_a)\ln f_a(\boldsymbol{\sigma}_a)}_{\text{self-energy}} + \underbrace{\sum_a \sum_{\boldsymbol{\sigma}_a} b_a(\boldsymbol{\sigma}_a)\ln b_a(\boldsymbol{\sigma}_a) - \sum_{(a,c)} b_{ac}(\sigma_{ac})\ln b_{ac}(\sigma_{ac})}_{\text{configurational entropy}}$$

$$\forall\ a;\ c\in a:\quad \sum_{\boldsymbol{\sigma}_a} b_a(\boldsymbol{\sigma}_a)=1,\quad b_{ac}(\sigma_{ac})=\sum_{\boldsymbol{\sigma}_a\setminus\sigma_{ac}} b_a(\boldsymbol{\sigma}_a)$$

$\Rightarrow$ Belief-Propagation Equations: $\left.\frac{\delta F}{\delta b}\right|_{\text{constr.}}=0$

MAP≈BP=Belief-Propagation (Bethe-Pieirls): iterative ⇒ Gallager '61; MacKay '98

- Exact on a tree ▸ Derivation Sketch
- Trading optimality for reduction in complexity: $\sim 2^L \to \sim L$
- BP = solving equations on the graph:

$$\eta_{\alpha j} = h_j + \sum_{\beta\neq\alpha}^{j\in\beta} \tanh^{-1}\left(\prod_{i\neq j}^{i\in\beta}\tanh\eta_{\beta i}\right) \quad \Leftarrow \text{LDPC representation}$$

- Message Passing = iterative BP
- Convergence of MP to minimum of Bethe Free energy can be enforced

Bethe free energy: variational approach
(Yedidia,Freeman,Weiss '01 - inspired by Bethe '35, Peierls '36)

$$F = \underbrace{-\sum_a \sum_{\boldsymbol{\sigma}_a} b_a(\boldsymbol{\sigma}_a) \ln f_a(\boldsymbol{\sigma}_a)}_{\text{self-energy}} + \underbrace{\sum_a \sum_{\boldsymbol{\sigma}_a} b_a(\boldsymbol{\sigma}_a) \ln b_a(\boldsymbol{\sigma}_a) - \sum_{(a,c)} b_{ac}(\sigma_{ac}) \ln b_{ac}(\sigma_{ac})}_{\text{configurational entropy}}$$

$\forall\ a;\ c \in a: \quad \sum_{\boldsymbol{\sigma}_a} b_a(\boldsymbol{\sigma}_a) = 1, \quad b_{ac}(\sigma_{ac}) = \sum_{\boldsymbol{\sigma}_a \setminus \sigma_{ac}} b_a(\boldsymbol{\sigma}_a)$

$\Rightarrow$ Belief-Propagation Equations: $\left.\frac{\delta F}{\delta b}\right|_{\text{constr.}} = 0$

MAP≈BP=Belief-Propagation (Bethe-Pieirls): iterative ⇒ Gallager '61; MacKay '98

- Exact on a tree ▸ Derivation Sketch
- Trading optimality for reduction in complexity: $\sim 2^L \rightarrow \sim L$
- BP = solving equations on the graph:

$$\eta_{\alpha j} = h_j + \sum_{\beta \neq \alpha}^{j \in \beta} \tanh^{-1}\left(\prod_{i \neq j}^{i \in \beta} \tanh \eta_{\beta i}\right) \quad \Leftarrow \text{LDPC representation}$$

- Message Passing = iterative BP
- Convergence of MP to minimum of Bethe Free energy can be enforced

Linear Programming version of Belief Propagation

In the limit of large SNR, $\ln f_a \to \pm\infty$: BP→LP

Minimize $F \approx E = -\sum_a \sum_{\boldsymbol{\sigma}_a} b_a(\boldsymbol{\sigma}_a) \ln f_a(\boldsymbol{\sigma}_a) =$ self energy
under set of linear constraints

LP decoding of LDPC codes Feldman, Wainwright, Karger '03

- ML can be restated as an LP over a codeword polytope
- LP decoding is a "local codewords" relaxation of LP-ML
- Codeword convergence certificate
- Discrete and Nice for Analysis
- Large polytope $\{b_\alpha, b_i\} \Rightarrow$ Small polytope $\{b_i\}$

Linear Programming version of Belief Propagation

In the limit of large SNR, $\ln f_a \to \pm\infty$: BP→LP

$$\text{Minimize } F \approx E = -\sum_a \sum_{\boldsymbol{\sigma}_a} b_a(\boldsymbol{\sigma}_a) \ln f_a(\boldsymbol{\sigma}_a) = \text{self energy}$$

under set of linear constraints

LP decoding of LDPC codes Feldman, Wainwright, Karger '03

- ML can be restated as an LP over a codeword polytope
- LP decoding is a "local codewords" relaxation of LP-ML
- Codeword convergence certificate
- Discrete and Nice for Analysis
- Large polytope $\{b_\alpha, b_i\} \Rightarrow$ Small polytope $\{b_i\}$

BP does not account for Loops

Questions:

- Is BP just a heuristic in a loopy case?
- Why does it (often) work so well?
- Does exact inference allow an expression in terms of BP?
- Can one correct BP systematically?

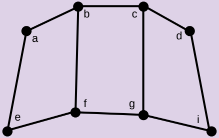
Previous Considerations:

- Rizzo, Montanari '05 - Corrections to BP approximation
- Parisi, Slanina '05 - BP as a saddle-point + corrections

Gauge Transformations

Chertkov,Chernyak '06

Local Gauge, G, Transformations



$$Z = \sum_{\boldsymbol{\sigma}} \prod_a f_a(\boldsymbol{\sigma}_a), \;\; \boldsymbol{\sigma}_a = (\sigma_{ab}, \sigma_{ac}, \cdots), \;\; \sigma_{ab} = \sigma_{ba} = \pm 1$$

$$f_a(\boldsymbol{\sigma}_a = (\sigma_{ab}, \cdots)) \to \sum_{\sigma'_{ab}} G_{ab}\left(\sigma_{ab}, \sigma'_{ab}\right) f_a(\sigma'_{ab}, \cdots)$$

$$\sum_{\sigma_{ab}} G_{ab}(\sigma_{ab}, \sigma') G_{ba}(\sigma_{ab}, \sigma'') = \delta(\sigma', \sigma'')$$

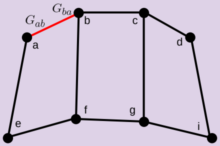
The partition function is invariant under any G-gauge!

$$Z = \sum_{\boldsymbol{\sigma}} \prod_a f_a(\boldsymbol{\sigma}_a) = \underbrace{\sum_{\boldsymbol{\sigma}} \prod_a \left(\sum_{\boldsymbol{\sigma}'_a} f_a(\boldsymbol{\sigma}'_a) \prod_{b \in a} G_{ab}(\sigma_{ab}, \sigma'_{ab}) \right)}_{\text{graphical trace}}$$

Gauge Transformations

Chertkov,Chernyak '06

Local Gauge, G, Transformations



$$Z=\sum_{\boldsymbol{\sigma}}\prod_a f_a(\boldsymbol{\sigma}_a),\ \ \boldsymbol{\sigma}_a=(\sigma_{ab},\sigma_{ac},\cdots),\ \ \sigma_{ab}=\sigma_{ba}=\pm 1$$

$$f_a(\boldsymbol{\sigma}_a=(\sigma_{ab},\cdots))\to\sum_{\sigma'_{ab}} G_{ab}\left(\sigma_{ab},\sigma'_{ab}\right) f_a(\sigma'_{ab},\cdots)$$

$$\sum_{\sigma_{ab}} G_{ab}(\sigma_{ab},\sigma')G_{ba}(\sigma_{ab},\sigma'')=\delta(\sigma',\sigma'')$$

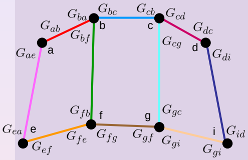
The partition function is invariant under any G-gauge!

$$Z=\sum_{\boldsymbol{\sigma}}\prod_a f_a(\boldsymbol{\sigma}_a)=\sum_{\boldsymbol{\sigma}}\prod_a\underbrace{\left(\sum_{\boldsymbol{\sigma}'_a} f_a(\boldsymbol{\sigma}'_a)\prod_{b\in a} G_{ab}(\sigma_{ab},\sigma'_{ab})\right)}_{\text{graphical trace}}$$

Gauge Transformations

Chertkov,Chernyak '06

Local Gauge, G, Transformations



$$Z=\sum_{\boldsymbol{\sigma}}\prod_a f_a(\boldsymbol{\sigma}_a),\ \ \boldsymbol{\sigma}_a=(\sigma_{ab},\sigma_{ac},\cdots),\ \ \sigma_{ab}=\sigma_{ba}=\pm 1$$

$$f_a(\boldsymbol{\sigma}_a=(\sigma_{ab},\cdots))\to\sum_{\sigma'_{ab}} G_{ab}\left(\sigma_{ab},\sigma'_{ab}\right) f_a(\sigma'_{ab},\cdots)$$

$$\sum_{\sigma_{ab}} G_{ab}(\sigma_{ab},\sigma')G_{ba}(\sigma_{ab},\sigma'')=\delta(\sigma',\sigma'')$$

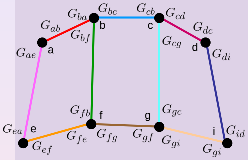
The partition function is invariant under any G-gauge!

$$Z=\sum_{\boldsymbol{\sigma}}\prod_a f_a\left(\boldsymbol{\sigma}_a\right)=\underbrace{\sum_{\boldsymbol{\sigma}}\prod_a\left(\sum_{\boldsymbol{\sigma}'_a} f_a(\boldsymbol{\sigma}'_a)\prod_{b\in a} G_{ab}(\sigma_{ab},\sigma'_{ab})\right)}_{\text{graphical trace}}$$

Gauge Transformations

Chertkov,Chernyak '06

Local Gauge, G, Transformations



$$Z = \sum_{\boldsymbol{\sigma}} \prod_a f_a(\boldsymbol{\sigma}_a), \;\; \boldsymbol{\sigma}_a = (\sigma_{ab}, \sigma_{ac}, \cdots), \;\; \sigma_{ab} = \sigma_{ba} = \pm 1$$

$$f_a(\boldsymbol{\sigma}_a = (\sigma_{ab}, \cdots)) \to \sum_{\sigma'_{ab}} G_{ab}\left(\sigma_{ab}, \sigma'_{ab}\right) f_a(\sigma'_{ab}, \cdots)$$

$$\sum_{\sigma_{ab}} G_{ab}(\sigma_{ab}, \sigma') G_{ba}(\sigma_{ab}, \sigma'') = \delta(\sigma', \sigma'')$$

The partition function is invariant under any G-gauge!

$$Z = \sum_{\boldsymbol{\sigma}} \prod_a f_a\left(\boldsymbol{\sigma}_a\right) = \underbrace{\sum_{\boldsymbol{\sigma}} \prod_a \left(\sum_{\boldsymbol{\sigma}'_a} f_a(\boldsymbol{\sigma}'_a) \prod_{b \in a} G_{ab}(\sigma_{ab}, \sigma'_{ab}) \right)}_{\text{graphical trace}}$$

Gauge Transformations: Binary Representation

$$Z=\sum_{\boldsymbol\sigma}\prod_a f_a(\boldsymbol\sigma_a)=\sum_{\boldsymbol\sigma'}\prod_a f_a(\boldsymbol\sigma_a)\prod_{bc}\frac{1+\sigma_{bc}\sigma_{cb}}{2},\quad \sigma_{bc}\neq\sigma_{cb}$$

The binary trick: $1+\sigma_{bc}\sigma_{cb}=$

$$\frac{\exp(\sigma_{bc}\eta_{bc}+\sigma_{cb}\eta_{cb})}{\cosh(\eta_{bc}+\eta_{cb})}\left(1+(\tanh(\eta_{bc}+\eta_{cb})-\sigma_{bc})(\tanh(\eta_{bc}+\eta_{cb})-\sigma_{cb})\cosh^2(\eta_{bc}+\eta_{cb})\right)$$

$$\tilde f_a(\boldsymbol\sigma_a)=f_a(\boldsymbol\sigma_a)\prod_{b\in a}\exp(\eta_{ab}\sigma_{ab})$$

$$V_{bc}(\sigma_{bc},\sigma_{cb})=1+(\tanh(\eta_{bc}+\eta_{cb})-\sigma_{bc})(\tanh(\eta_{bc}+\eta_{cb})-\sigma_{cb})\cosh^2(\eta_{bc}+\eta_{cb})$$

Graph Coloring

$$Z=\left(\prod_{bc}2\cosh(\eta_{bc}+\eta_{cb})\right)^{-1}\sum_{\boldsymbol\sigma'}\prod_a\tilde f_a(\boldsymbol\sigma_a)\prod_{bc}V_{bc}$$

$$Z=\underbrace{Z_0(\eta)}_{\substack{\text{ground state}\\ \boldsymbol\sigma=+1}}+\underbrace{\sum_{\text{all possible colorings of the graph}}Z_c(\eta)}_{\text{excited states}}$$

Gauges and BP

Fixing the gauges ⇒ BP equations!!

Two alternative ways to understand BP-gauges:

▸ BP equations

Color Principe:
no loose ends

$$Z = Z_0(\eta) + \sum_{c=colorings} Z_c(\eta)$$

$$Z_c(\eta) = \prod_{a\in C} \Psi_{a;C}(\eta)$$

Variational Principe:
ground state is η-independent

$$Z \to Z_0(\eta)$$

$$\left.\frac{\partial Z_0}{\partial \eta_{ab}}\right|_{\eta^{(bp)}} = 0$$

Related to Wainwright, Jaakkola, and Willsky '03
Reparametrization Framework

Gauges and BP

Fixing the gauges ⇒ BP equations!!

Two alternative ways to understand BP-gauges:

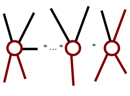
▸ BP equations

Color Principe:

no loose ends

$$Z = Z_0(\eta) + \sum_{c=colorings} Z_c(\eta)$$

$$Z_c(\eta) = \prod_{a\in C} \Psi_{a;C}(\eta)$$



Variational Principe:

ground state is η-independent

$$Z \to Z_0(\eta)$$

$$\left.\frac{\partial Z_0}{\partial \eta_{ab}}\right|_{\eta^{(bp)}} = 0$$

Related to Wainwright, Jaakkola, and Willsky '03

Reparametrization Framework

Gauges and BP

Fixing the gauges ⇒ BP equations!!

Two alternative ways to understand BP-gauges:

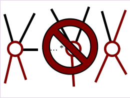
▸ BP equations

Color Principe:

no loose ends

$$Z = Z_0(\eta) + \sum_{c=colorings} Z_c(\eta)$$

$$Z_c(\eta) = \prod_{a\in C} \Psi_{a;C}(\eta)$$



Variational Principe:

ground state is η-independent

$$Z \to Z_0(\eta)$$

$$\left.\frac{\partial Z_0}{\partial \eta_{ab}}\right|_{\eta^{(bp)}} = 0$$

Related to Wainwright, Jaakkola, and Willsky '03

Reparametrization Framework

Gauges and BP

Fixing the gauges ⇒ BP equations!!

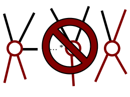
Two alternative ways to understand BP-gauges:

▸ BP equations

Color Principe:

no loose ends

$$Z = Z_0(\eta) + \sum_{c=colorings} Z_c(\eta)$$



$Z_c(\eta) = \prod_{a\in C} \Psi_{a;C}(\eta)$

Variational Principe:

ground state is η-independent

$$Z \to Z_0(\eta)$$

$$\left.\frac{\partial Z_0}{\partial \eta_{ab}}\right|_{\eta^{(bp)}} = 0$$

Related to Wainwright, Jaakkola, and Willsky '03

Reparametrization Framework

Gauges and BP

Fixing the gauges ⇒ BP equations!!

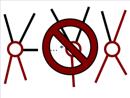
Two alternative ways to understand BP-gauges:

▸ BP equations

Color Principle:

no loose ends

$$Z = Z_0(\eta) + \sum_{c=colorings} Z_c(\eta)$$



$$Z_c(\eta) = \prod_{a\in C} \Psi_{a;C}(\eta)$$

Variational Principle:

ground state is η-independent

$$Z \to Z_0(\eta)$$

$$\left.\frac{\partial Z_0}{\partial \eta_{ab}}\right|_{\eta^{(bp)}} = 0$$

Related to Wainwright, Jaakkola, and Willsky '03

Reparametrization Framework

Loop Series:

Chertkov, Chernyak '06

Exact (!!) expression in terms of BP

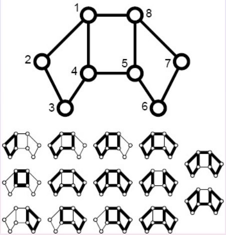
$$Z=\sum_{\boldsymbol{\sigma}_\sigma}\prod_a f_a(\boldsymbol{\sigma}_a)=Z_0\left(1+\sum_C r(C)\right)$$

$$r(C)=\frac{\prod_{a\in C}\mu_a}{\prod_{(ab)\in C}(1-m_{ab}^2)}=\prod_{a\in C}\tilde{\mu}_a$$

$C\in$ Generalized Loops $=$ Loops without loose ends

$$m_{ab}=\int d\boldsymbol{\sigma}_a b_a^{(bp)}(\boldsymbol{\sigma}_a)\sigma_{ab}$$

$$\mu_a=\int d\boldsymbol{\sigma}_a b_a^{(bp)}(\boldsymbol{\sigma}_a)\prod_{b\in a,C}(\sigma_{ab}-m_{ab})$$



- The Loop Series is finite
- All terms in the series are calculated within BP
- BP is exact on a tree
- BP is a Gauge fixing condition. Other choices of Gauges would lead to different representation.

Features of the Loop Calculus

$$Z = Z_0(1+\sum_C r_C),\ r_C = \prod_{a\in C}\tilde{\mu}_a$$

- Bethe Free Energy is related to the "ground state" term in the partition function: $F(b^*(\eta)) = -\ln Z_0(\eta)$, where $b^*_a(\boldsymbol{\sigma}_a) = \frac{f_a(\boldsymbol{\sigma}_a)\exp(\sum_{b\in a}\eta_{ab}\sigma_{ab})}{\sum_{\boldsymbol{\sigma}_a} f_a(\boldsymbol{\sigma}_a)\exp(\sum_{b\in a}\eta_{ab}\sigma_{ab})}$, $\quad b^*_{ab}(\sigma_{ab}) = \frac{\exp((\eta_{ab}+\eta_{ba})\sigma_{ab})}{2\cosh(\eta_{ab}+\eta_{ba})}$
- Extrema of $F(b)$ are related to extrema of $Z_0(\eta)$
- Loop series can be built around any extremum (minimum, maximum or saddle-point) of the Bethe Free energy
- $-1 \le r_C, \tilde{\mu}_a \le 1$
- Linear Programming limit of the Loop Calculus is well defined

Features of the Loop Calculus

$$Z = Z_0(1 + \sum_C r_C),\ r_C = \prod_{a\in C} \tilde{\mu}_a$$

- Bethe Free Energy is related to the "ground state" term in the partition function: $F(b^*(\eta)) = -\ln Z_0(\eta)$, where $b^*_a(\boldsymbol{\sigma}_a) = \frac{f_a(\boldsymbol{\sigma}_a)\exp(\sum_{b\in a}\eta_{ab}\sigma_{ab})}{\sum_{\boldsymbol{\sigma}_a} f_a(\boldsymbol{\sigma}_a)\exp(\sum_{b\in a}\eta_{ab}\sigma_{ab})}$, $\quad b^*_{ab}(\sigma_{ab}) = \frac{\exp((\eta_{ab}+\eta_{ba})\sigma_{ab})}{2\cosh(\eta_{ab}+\eta_{ba})}$
- Extrema of $F(b)$ are related to extrema of $Z_0(\eta)$
- Loop series can be built around any extremum (minimum, maximum or saddle-point) of the Bethe Free energy
- $-1 \leq r_C, \tilde{\mu}_a \leq 1$
- Linear Programming limit of the Loop Calculus is well defined

Features of the Loop Calculus

$$Z = Z_0(1+\sum_C r_C),\ r_C = \prod_{a\in C}\tilde{\mu}_a$$

- Bethe Free Energy is related to the "ground state" term in the partition function: $F(b^*(\eta)) = -\ln Z_0(\eta)$, where $b^*_a(\boldsymbol{\sigma}_a) = \frac{f_a(\boldsymbol{\sigma}_a)\exp(\sum_{b\in a}\eta_{ab}\sigma_{ab})}{\sum_{\boldsymbol{\sigma}_a} f_a(\boldsymbol{\sigma}_a)\exp(\sum_{b\in a}\eta_{ab}\sigma_{ab})}$, $\quad b^*_{ab}(\sigma_{ab}) = \frac{\exp((\eta_{ab}+\eta_{ba})\sigma_{ab})}{2\cosh(\eta_{ab}+\eta_{ba})}$
- Extrema of $F(b)$ are related to extrema of $Z_0(\eta)$
- Loop series can be built around any extremum (minimum, maximum or saddle-point) of the Bethe Free energy
- $-1 \le r_C, \tilde{\mu}_a \le 1$
- Linear Programming limit of the Loop Calculus is well defined

Features of the Loop Calculus

$$Z = Z_0(1+\sum_C r_C),\ r_C = \prod_{a\in C}\tilde{\mu}_a$$

- Bethe Free Energy is related to the "ground state" term in the partition function: $F(b^*(\eta)) = -\ln Z_0(\eta)$, where $b^*_a(\boldsymbol{\sigma}_a) = \frac{f_a(\boldsymbol{\sigma}_a)\exp(\sum_{b\in a}\eta_{ab}\sigma_{ab})}{\sum_{\boldsymbol{\sigma}_a} f_a(\boldsymbol{\sigma}_a)\exp(\sum_{b\in a}\eta_{ab}\sigma_{ab})}$, $\quad b^*_{ab}(\sigma_{ab}) = \frac{\exp((\eta_{ab}+\eta_{ba})\sigma_{ab})}{2\cosh(\eta_{ab}+\eta_{ba})}$
- Extrema of $F(b)$ are related to extrema of $Z_0(\eta)$
- Loop series can be built around any extremum (minimum, maximum or saddle-point) of the Bethe Free energy
- $-1 \leq r_C, \tilde{\mu}_a \leq 1$
- Linear Programming limit of the Loop Calculus is well defined

Features of the Loop Calculus

$$Z = Z_0(1+\sum_C r_C),\ r_C = \prod_{a\in C}\tilde{\mu}_a$$

- Bethe Free Energy is related to the "ground state" term in the partition function: $F(b^*(\eta)) = -\ln Z_0(\eta)$, where $b^*_a(\boldsymbol{\sigma}_a) = \frac{f_a(\boldsymbol{\sigma}_a)\exp(\sum_{b\in a}\eta_{ab}\sigma_{ab})}{\sum_{\boldsymbol{\sigma}_a} f_a(\boldsymbol{\sigma}_a)\exp(\sum_{b\in a}\eta_{ab}\sigma_{ab})}$, $\quad b^*_{ab}(\sigma_{ab}) = \frac{\exp((\eta_{ab}+\eta_{ba})\sigma_{ab})}{2\cosh(\eta_{ab}+\eta_{ba})}$
- Extrema of $F(b)$ are related to extrema of $Z_0(\eta)$
- Loop series can be built around any extremum (minimum, maximum or saddle-point) of the Bethe Free energy
- $-1 \le r_C, \tilde{\mu}_a \le 1$
- Linear Programming limit of the Loop Calculus is well defined

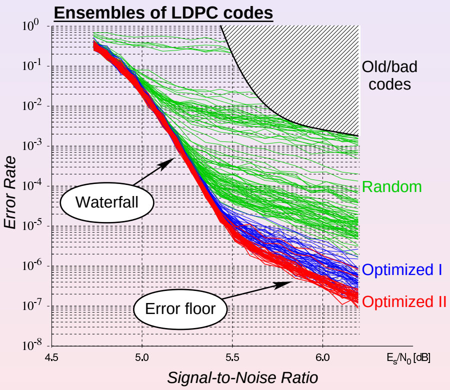
Summary (first lecture)

- BP eqs. solve Gauge fixing conditions
- BP eqs also explains no-loose-end coloring constraints
- BP minimizes gauge dependence in the ground state
- Loop series expresses partition function in terms of a sum of terms, each associated with a generalized loop of the graph
- Each term in the Loop Series depends explicitly on the BP solution

All papers are available at http://cnls.lanl.gov/∼chertkov/pub.htm

Bibliography

Error-Floor



- BER vs SNR = measure of performance
- Finite size effects
- Waterfall $\leftrightarrow$ Error-floor
- Error-floor typically emerges due to sub-optimality of decoding
- Monte-Carlo is useless at FER $\lesssim 10^{-8}$
- Need an efficient method to analyze error-floor

Pseudo-codewords and Instantons

Error-floor is caused by Pseudo-codewords:

Wiberg '96; Forney et.al'99; Frey et.al '01; Richardson '03; Vontobel, Koetter '04-'06

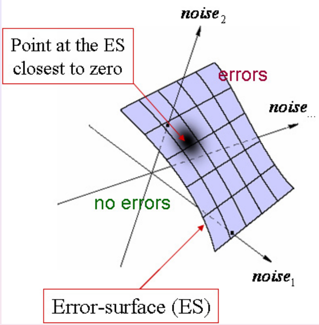
Instanton = optimal conf of the noise

$$BER = \int d(noise)\ WEIGHT(noise)$$

$$BER \sim WEIGHT\begin{pmatrix} optimal\ \ conf \\ of\ \ the\ \ noise \end{pmatrix}$$

$$\begin{matrix} optimal\ \ conf \\ of\ \ the\ \ noise \end{matrix} = \begin{matrix} Point\ \ at\ \ the\ \ ES \\ closest\ \ to\ \ "0" \end{matrix}$$

Instantons are decoded to Pseudo-Codewords



Instanton-amoeba

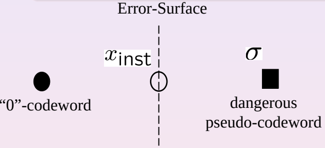
= optimization algorithm
Stepanov, et.al '04,'05
Stepanov, Chertkov '06

Pseudo-Codeword Search Algorithm.

LP decoding ($\sigma_i = 0, 1$ AWGN channel)

Minimize, $E = \sum_\alpha \sum_{\sigma_\alpha} b_\alpha(\sigma_\alpha) \sum_{i\in\alpha} \sigma_i(1-2x_i)/q_i$, under $0 \le b_i(\sigma_i), b_\alpha(\sigma_\alpha) \le 1$

$\forall\ \alpha :\ \sum_{\sigma_\alpha} b_\alpha(\sigma_\alpha) = 1$, & $\forall\ i\ \forall\ \alpha \ni i :\ b_i(\sigma_i) = \sum_{\sigma_\alpha \setminus \sigma_i} b_\alpha(\sigma_\alpha)$



Weighted Median:

$$\mathbf{x}_{\text{inst}} = \frac{\boldsymbol{\sigma}}{2}\frac{\sum_i \sigma_i}{\sum_i \sigma_i^2}, \quad d = \frac{(\sum_i \sigma_i)^2}{\sum_i \sigma_i^2}$$

$$\text{FER} \sim \exp(-d \cdot s^2/2)$$

Wiberg '96; Forney et.al '01
Vontobel, Koetter '03,'05

PCS Algorithm Chertkov, Stepanov '06

- **Start:** Initiate $\mathbf{x}^{(0)}$.
- **Step 1:** $\mathbf{x}^{(k)}$ is decoded to $\boldsymbol{\sigma}^{(k)}$.
- **Step 2:** Find $\mathbf{y}^{(k)}$ - weighted median between $\boldsymbol{\sigma}^{(k)}$, and "0"
- **Step 3:** If $\mathbf{y}^{(k)} = \mathbf{y}^{(k-1)}$, $k_* = k$ End. Otherwise go to **Step 2** with $\mathbf{x}^{(k+1)} = \mathbf{y}^{(k)} + 0$.

(155, 64, 20), AWGN test:

- Fast Convergence

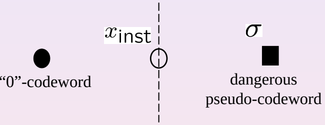
$\sim$ 200 pseudo-codewords within $16.4037 < d < 20$

Pseudo-Codeword Search Algorithm.

LP decoding $(\sigma_i = 0, 1$ AWGN channel)

Minimize, $E = \sum_\alpha \sum_{\sigma_\alpha} b_\alpha(\sigma_\alpha) \sum_{i\in\alpha} \sigma_i(1-2x_i)/q_i$, under $0 \le b_i(\sigma_i), b_\alpha(\sigma_\alpha) \le 1$

$\forall\ \alpha:\ \sum_{\sigma_\alpha} b_\alpha(\sigma_\alpha) = 1,\quad \&\quad \forall\ i\ \forall\ \alpha \ni i:\ b_i(\sigma_i) = \sum_{\sigma_\alpha\setminus\sigma_i} b_\alpha(\sigma_\alpha)$



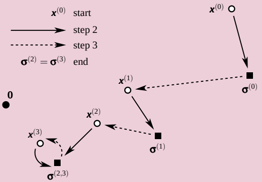
Weighted Median:

$$\mathbf{x}_{\text{inst}} = \frac{\boldsymbol{\sigma}}{2}\frac{\sum_i \sigma_i}{\sum_i \sigma_i^2},\quad d = \frac{(\sum_i \sigma_i)^2}{\sum_i \sigma_i^2}$$

$$\text{FER} \sim \exp(-d\cdot s^2/2)$$

Wiberg '96; Forney et.al '01
Vontobel, Koetter '03,'05

PCS Algorithm Chertkov, Stepanov '06



- **Start:** Initiate $\mathbf{x}^{(0)}$.
- **Step 1:** $\mathbf{x}^{(k)}$ is decoded to $\boldsymbol{\sigma}^{(k)}$.
- **Step 2:** Find $\mathbf{y}^{(k)}$ - weighted median between $\boldsymbol{\sigma}^{(k)}$, and "0"
- **Step 3:** If $\mathbf{y}^{(k)} = \mathbf{y}^{(k-1)}$, $k_* = k$ End. Otherwise go to **Step 2** with $\mathbf{x}^{(k+1)} = \mathbf{y}^{(k)} + 0$.

$(155, 64, 20)$, AWGN test:

- Fast Convergence

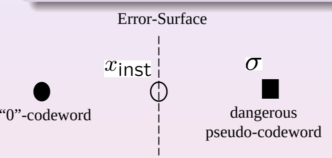
$\sim$ 200 pseudo-codewords within $16.4037 < d < 20$

Pseudo-Codeword Search Algorithm.

LP decoding ($\sigma_i = 0, 1$ AWGN channel)

Minimize, $E = \sum_\alpha \sum_{\sigma_\alpha} b_\alpha(\sigma_\alpha) \sum_{i\in\alpha} \sigma_i(1-2x_i)/q_i$, under $0 \le b_i(\sigma_i), b_\alpha(\sigma_\alpha) \le 1$

$\forall\ \alpha: \quad \sum_{\sigma_\alpha} b_\alpha(\sigma_\alpha) = 1, \qquad \& \qquad \forall\ i\ \forall\ \alpha \ni i: \quad b_i(\sigma_i) = \sum_{\sigma_\alpha\setminus\sigma_i} b_\alpha(\sigma_\alpha)$



Weighted Median:

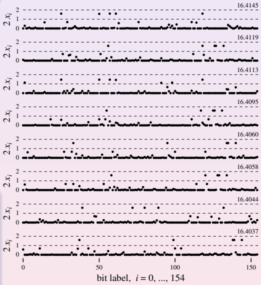
$$\mathbf{x}_{\text{inst}} = \frac{\boldsymbol{\sigma}}{2}\frac{\sum_i \sigma_i}{\sum_i \sigma_i^2}, \quad d = \frac{(\sum_i \sigma_i)^2}{\sum_i \sigma_i^2}$$

$$\text{FER} \sim \exp(-d\cdot s^2/2)$$

Wiberg '96; Forney et.al '01
Vontobel, Koetter '03,'05

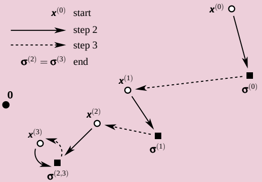
$(155, 64, 20)$, AWGN test:

- Fast Convergence



$\sim$ 200 pseudo-codewords within $16.4037 < d < 20$

PCS Algorithm Chertkov, Stepanov '06



- **Start:** Initiate $\mathbf{x}^{(0)}$.
- **Step 1:** $\mathbf{x}^{(k)}$ is decoded to $\boldsymbol{\sigma}^{(k)}$.
- **Step 2:** Find $\mathbf{y}^{(k)}$ - weighted median between $\boldsymbol{\sigma}^{(k)}$, and "0"
- **Step 3:** If $\mathbf{y}^{(k)} = \mathbf{y}^{(k-1)}$, $k_* = k$ End. Otherwise go to **Step 2** with $\mathbf{x}^{(k+1)} = \mathbf{y}^{(k)} + 0$.

Dendro-LDPC

LP complexity grows exponentially with check degree

Current solutions:

- Adaptive LP (Taghavi, Siegel '06)
- BP-style relaxation of LP (Vontobel, Koetter '06)

Dendro-trick = Graph Modification (our solution) Chertkov,Stepanov'07

- MAP solutions are identical
- Set of Pseudo-codewords are identical
- Instanton spectra are very alike, ≈

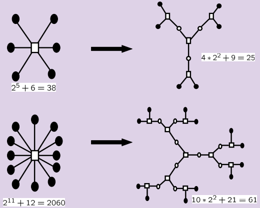
Dendro-LDPC

LP complexity grows exponentially with check degree

Current solutions:

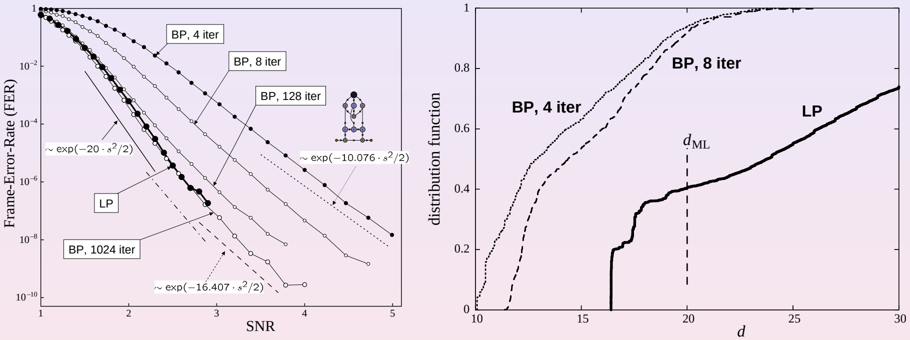
- Adaptive LP (Taghavi, Siegel '06)
- BP-style relaxation of LP (Vontobel, Koetter '06)

Dendro-trick = Graph Modification (our solution) Chertkov,Stepanov'07



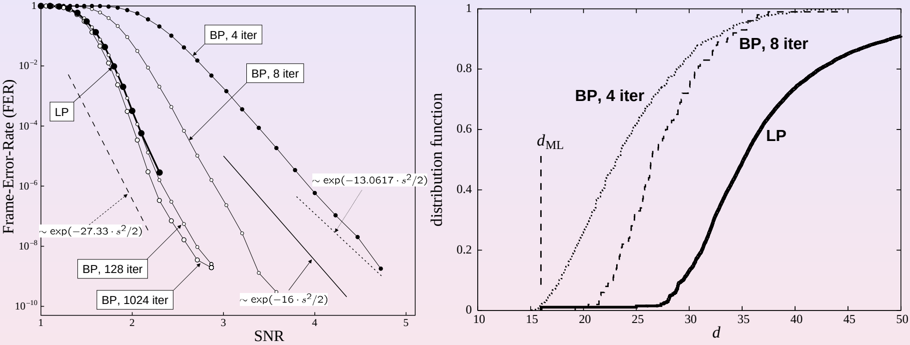
- MAP solutions are identical
- Set of Pseudo-codewords are identical
- Instanton spectra are very alike, $\approx$

FER vs SNR & pseudo-codeword spectrum: Tanner code

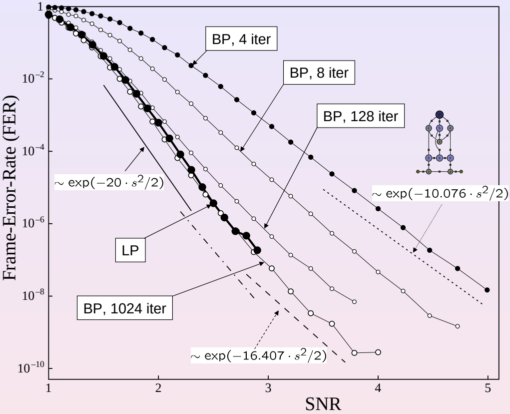


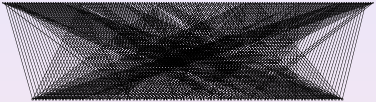
- $d_{\text{min;inst}} < d_{\text{ML}} = 20$
- Dangerous instantons are frequent

FER vs SNR & pseudo-codeword spectrum: Margulis p=7 code



- $d_{\text{min;inst;LP}} > d_{\text{ML}} = 16$
- Dangerous codewords are rare $\Rightarrow$ emergence of a steep transient asymptotic of FER vs SNR





Instanton-amoeba:
Stepanov, et.al '04,'05,'06
LP-search:
Chertkov, Stepanov '06,'07

What does Loop Calculus show for dangerous Pseudo-codewords?

Why loops?

If BP/LP fails while ML/MAP would not [pseudo-codewords]
... one needs to account for Loops

- How many loops are critical to recover from the failure?
- Will accounting for a single most important loop be sufficient?
- How long is the critical loop?
- Will it be difficult to find the critical loop?
- If there are many ...
  how are the critical loops distributed over scales?

Why loops?

If BP/LP fails while ML/MAP would not [pseudo-codewords]
... one needs to account for Loops

- How many loops are critical to recover from the failure?
- Will accounting for a single most important loop be sufficient?
- How long is the critical loop?
- Will it be difficult to find the critical loop?
- If there are many ...
  how are the critical loops distributed over scales?

Why loops?

If BP/LP fails while ML/MAP would not [pseudo-codewords]
... one needs to account for Loops

- How many loops are critical to recover from the failure?
- Will accounting for a single most important loop be sufficient?
- How long is the critical loop?
- Will it be difficult to find the critical loop?
- If there are many ...
  how are the critical loops distributed over scales?

Why loops?

If BP/LP fails while ML/MAP would not [pseudo-codewords]
... one needs to account for Loops

- How many loops are critical to recover from the failure?
- Will accounting for a single most important loop be sufficient?
- How long is the critical loop?
- Will it be difficult to find the critical loop?
- If there are many ...
  how are the critical loops distributed over scales?

Why loops?

If BP/LP fails while ML/MAP would not [pseudo-codewords]
... one needs to account for Loops

- How many loops are critical to recover from the failure?
- Will accounting for a single most important loop be sufficient?
- How long is the critical loop?
- Will it be difficult to find the critical loop?
- If there are many ...
  how are the critical loops distributed over scales?

Why loops?

If BP/LP fails while ML/MAP would not [pseudo-codewords]
... one needs to account for Loops

- How many loops are critical to recover from the failure?
- Will accounting for a single most important loop be sufficient?
- How long is the critical loop?
- Will it be difficult to find the critical loop?
- If there are many ...
  how are the critical loops distributed over scales?

Loop Calculus & Pseudo-Codeword Analysis

Chertkov,Chernyak '06

Single loop truncation

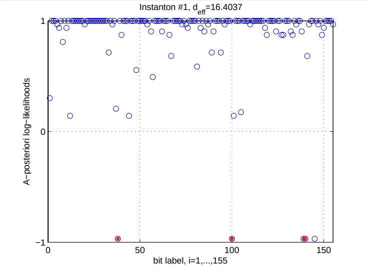
$$Z = Z_0(1+\sum_C r_C) \approx Z_0(1+r(\Gamma))$$

Synthesis of Pseudo-Codeword Search Algorithm (Chertkov, Stepanov '06) & Loop Calculus

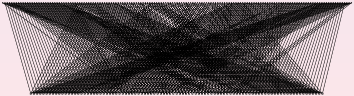
- Consider pseudo-codewords one after other
- For an individual pseudo-codeword/instanton identify a critical loop, Γ, giving major contribution to the loop series.
- Hint: look for single connected loops and use local "triad" contributions as a tester: $r(\Gamma) = \prod_{\alpha\in\Gamma} \tilde{\mu}_\alpha^{(bp)}$

Proof-of-Concept test [(155, 64, 20) code over AWGN]

- $\forall$ pseudo-codewords with $16.4037 < d < 20$ ($\sim$ 200 found) there always exists a simple single-connected critical loop(s) with $r(\Gamma) \sim 1$.
- Pseudo-codewords with the lowest d show $r(\Gamma) = 1$
- Invariant with respect to other choices of the original codeword



Bigger Set



Extended Variational Principe & Loop-Corrected BP

Bare BP Variational Principle: $\left.\frac{\partial Z_0}{\partial \eta_{ab}}\right|_{\eta^{(bp)}} = 0$

New choice of Gauges guided by the knowledge of the critical loop Γ

$$\left.\frac{\partial \exp(-\mathcal{F})}{\partial \eta_{ab}}\right|_{\eta_{\text{eff}}} = 0, \quad \mathcal{F} \equiv -\ln(Z_0 + Z_\Gamma)$$

BP-equations are modified along the critical loop Γ

$$\left.\frac{\sum_{\sigma_a}(\tanh(\eta_{ab}+\eta_{ba})-\sigma_{ab})P_a(\sigma_a)}{\sum_{\sigma_a} P_a(\sigma_a)}\right|_{\eta_{\text{eff}}} = \text{explicitly known contribution}\Big|_{\eta_{\text{eff}}} \neq 0 \quad [\text{along } \Gamma]$$

Loop-Corrected BP Algorithm

- **1.** Run bare BP algorithm. Terminate if BP succeeds (i.e. a valid code word is found).
- **2.** If BP fails find the most relevant loop Γ that corresponds to the maximal $|r_\Gamma|$. Triad search is helping.
- **3.** Solve the modified-BP equations for the given Γ. Terminate if the improved-BP succeeds.
- **4.** Return to **Step 2** with an improved Γ-loop selection.

Extended Variational Principe & Loop-Corrected BP

Bare BP Variational Principe: $\left.\frac{\partial Z_0}{\partial \eta_{ab}}\right|_{\eta^{(bp)}} = 0$

New choice of Gauges guided by the knowledge of the critical loop Γ

$$\left.\frac{\partial \exp(-\mathcal{F})}{\partial \eta_{ab}}\right|_{\eta_{\rm eff}} = 0, \quad \mathcal{F} \equiv -\ln(Z_0 + Z_\Gamma)$$

BP-equations are modified along the critical loop Γ

$$\left.\frac{\sum_{\sigma_a}(\tanh(\eta_{ab}+\eta_{ba})-\sigma_{ab})P_a(\sigma_a)}{\sum_{\sigma_a}P_a(\sigma_a)}\right|_{\eta_{\rm eff}} = \text{explicitly known contribution}\Big|_{\eta_{\rm eff}} \neq 0 \quad [\text{along } \Gamma]$$

Loop-Corrected BP Algorithm

- **1.** Run bare BP algorithm. Terminate if BP succeeds (i.e. a valid code word is found).
- **2.** If BP fails find the most relevant loop Γ that corresponds to the maximal $|r_\Gamma|$. Triad search is helping.
- **3.** Solve the modified-BP equations for the given Γ. Terminate if the improved-BP succeeds.
- **4.** Return to **Step 2** with an improved Γ-loop selection.

Extended Variational Principle & Loop-Corrected BP

Bare BP Variational Principle: $$\left.\frac{\partial Z_0}{\partial \eta_{ab}}\right|_{\eta^{(bp)}} = 0$$

New choice of Gauges guided by the knowledge of the critical loop Γ

$$\left.\frac{\partial \exp(-\mathcal{F})}{\partial \eta_{ab}}\right|_{\eta_{\text{eff}}} = 0, \quad \mathcal{F} \equiv -\ln(Z_0 + Z_\Gamma)$$

BP-equations are modified along the critical loop Γ

$$\left.\frac{\sum_{\boldsymbol{\sigma}_a}(\tanh(\eta_{ab}+\eta_{ba})-\sigma_{ab})P_a(\boldsymbol{\sigma}_a)}{\sum_{\boldsymbol{\sigma}_a} P_a(\boldsymbol{\sigma}_a)}\right|_{\eta_{\text{eff}}} = \text{explicitly known contribution}\Big|_{\eta_{\text{eff}}} \neq 0 \quad [\text{along } \Gamma]$$

Loop-Corrected BP Algorithm

- **1.** Run bare BP algorithm. Terminate if BP succeeds (i.e. a valid code word is found).
- **2.** If BP fails find the most relevant loop Γ that corresponds to the maximal $|r_\Gamma|$. Triad search is helping.
- **3.** Solve the modified-BP equations for the given Γ. Terminate if the improved-BP succeeds.
- **4.** Return to **Step 2** with an improved Γ-loop selection.

Extended Variational Principle & Loop-Corrected BP

Bare BP Variational Principle: $\left.\frac{\partial Z_0}{\partial \eta_{ab}}\right|_{\eta^{(bp)}} = 0$

New choice of Gauges guided by the knowledge of the critical loop Γ

$$\left.\frac{\partial \exp(-\mathcal{F})}{\partial \eta_{ab}}\right|_{\eta_{\text{eff}}} = 0, \quad \mathcal{F} \equiv -\ln(Z_0 + Z_\Gamma)$$

BP-equations are modified along the critical loop Γ

$$\left.\frac{\sum_{\boldsymbol{\sigma}_a}(\tanh(\eta_{ab}+\eta_{ba})-\sigma_{ab})P_a(\boldsymbol{\sigma}_a)}{\sum_{\boldsymbol{\sigma}_a} P_a(\boldsymbol{\sigma}_a)}\right|_{\eta_{\text{eff}}} = \text{explicitly known contribution}\Big|_{\eta_{\text{eff}}} \neq 0 \quad [\text{along } \Gamma]$$

Loop-Corrected BP Algorithm

- **1.** Run bare BP algorithm. Terminate if BP succeeds (i.e. a valid code word is found).
- **2.** If BP fails find the most relevant loop Γ that corresponds to the maximal $|r_\Gamma|$. Triad search is helping.
- **3.** Solve the modified-BP equations for the given Γ. Terminate if the improved-BP succeeds.
- **4.** Return to **Step 2** with an improved Γ-loop selection.

LP-erasure = simple heuristics

- **1.** Run LP algorithm. Terminate if LP succeeds (i.e. a valid code word is found).
- **2.** If LP fails, find the most relevant loop Γ that corresponds to the maximal amplitude $r(\Gamma)$.
- **3.** Modify the log-likelihoods along the loop Γ introducing a shift towards zero, i.e. introduce a complete or partial erasure of the log-likelihoods at the bits. Run LP with modified log-likelihoods. Terminate if the modified LP succeeds.
- **4.** Return to **Step 2** with an improved selection principle for the critical loop.

$(155, 64, 20)$ Test

- **IT WORKS!**
  All troublemakers ($\sim$ 200 of them) previously found by LP-based Pseudo-Codeword-Search Algorithm method were successfully corrected by the LP-erasure algorithm.
- Method is invariant with respect the choice of the codeword (used to generate pseudo-codewords).

General Conjecture:

- Loop-erasure algorithm is capable of reducing the error-floor
- Bottleneck is in finding the critical loop
- Local adjustment of the algorithm, anywhere along the critical loop, in the spirit of the Facet Guessing (Dimakis, Wainwright '06), may be sufficient $\Rightarrow$

LP-erasure = simple heuristics

- **1.** Run LP algorithm. Terminate if LP succeeds (i.e. a valid code word is found).
- **2.** If LP fails, find the most relevant loop Γ that corresponds to the maximal amplitude $r(\Gamma)$.
- **3.** Modify the log-likelihoods along the loop Γ introducing a shift towards zero, i.e. introduce a complete or partial erasure of the log-likelihoods at the bits. Run LP with modified log-likelihoods. Terminate if the modified LP succeeds.
- **4.** Return to **Step 2** with an improved selection principle for the critical loop.

$(155, 64, 20)$ Test

- **IT WORKS!**
  All troublemakers ($\sim$ 200 of them) previously found by LP-based Pseudo-Codeword-Search Algorithm method were successfully corrected by the LP-erasure algorithm.
- Method is invariant with respect the choice of the codeword (used to generate pseudo-codewords).

General Conjecture:

- Loop-erasure algorithm is capable of reducing the error-floor
- Bottleneck is in finding the critical loop
- Local adjustment of the algorithm, anywhere along the critical loop, in the spirit of the Facet Guessing (Dimakis, Wainwright '06), may be sufficient $\Rightarrow$

LP-erasure = simple heuristics

- **1.** Run LP algorithm. Terminate if LP succeeds (i.e. a valid code word is found).
- **2.** If LP fails, find the most relevant loop Γ that corresponds to the maximal amplitude $r(\Gamma)$.
- **3.** Modify the log-likelihoods along the loop Γ introducing a shift towards zero, i.e. introduce a complete or partial erasure of the log-likelihoods at the bits. Run LP with modified log-likelihoods. Terminate if the modified LP succeeds.
- **4.** Return to **Step 2** with an improved selection principle for the critical loop.

$(155, 64, 20)$ Test

- **IT WORKS!**
  All troublemakers ($\sim$ 200 of them) previously found by LP-based Pseudo-Codeword-Search Algorithm method were successfully corrected by the LP-erasure algorithm.
- Method is invariant with respect the choice of the codeword (used to generate pseudo-codewords).

General Conjecture:

- Loop-erasure algorithm is capable of reducing the error-floor
- Bottleneck is in finding the critical loop
- Local adjustment of the algorithm, anywhere along the critical loop, in the spirit of the Facet Guessing (Dimakis, Wainwright '06), may be sufficient $\Rightarrow$

Breaking the critical loop locally

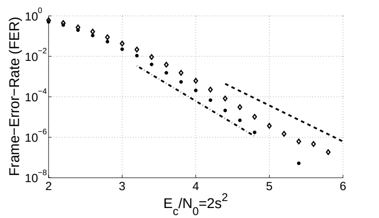
Chertkov '07

- Exhaustive Bit Guessing (simplified version of the Facet Guessing [Dimakis, Wainwright '06]) corrects all the ~ 200 dangerous pseudo-codewords !!
- Set of "successful" bits correlates strongly with the set of bits forming the critical loop

Loop Guided Guessing (LGG)

- **1.** Run the LP algorithm. Terminate if LP succeeds (i.e. a valid code word is found).
- **2.** If LP fails, find the critical loop, Γ, the one with maximal value of $|r(\Gamma)|$ in the loop series.
- **3.** Pick any bit along the critical loop at random and form two corrected LP schemes, different from the bare LP schemes by only one extra equality condition, enforcing the value of a bit to be 1 or 0 respectively.
- **4.** Run both LP-corrected schemes and choose the output which corresponds to the smallest self-energy. Terminate if the modified LP succeeds.
- **5.** Return to **Step 3** selecting another bit along the critical loop or to **Step 2** for an improved selection principle for the critical loop if the list of all the bits along the previously selected loop is exhausted.

[155, 64, 20] test of LGG



- Complexity of LGG is the same as of LP
- LGG corrects 9 out of 10 errors at $E_b/N_0 = 4.8$!!
- Error Floor is Reduced !!

Summary (second lecture)

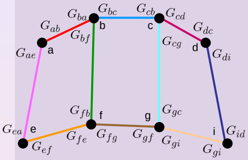
- Error floor is typically due to rare but dangerous pseudo-codewords.
- Instanton-amoeba and, especially, Pseudo-Codeword Search Algorithm offer efficient methods of the error-floor exploration.
- Loop Series for the factor functions of a dangerous pseudo-codeword can be accurately approximated by a sum of the leading BP term and a critical loop term. [Experimentally verified conjecture.]
- Loop Guided Guessing is an efficient algorithm (of the same complexity as LP) seriously outperforming the bare LP and overall reducing the error-floor. LGG "brakes" the critical loop locally at any bit of the critical loop.

All papers are available at http://cnls.lanl.gov/∼chertkov/pub.htm

▸ Bibliography

Gauge Transformations

Local Gauge, G, Transformations



$$Z = \sum_{\boldsymbol{\sigma}} \prod_a f_a(\boldsymbol{\sigma}_a), \ \ \boldsymbol{\sigma}_a = (\sigma_{ab}, \sigma_{ac}, \cdots)$$

$$\sigma_{ab} = \sigma_{ba} = 0, \cdots, q-1$$

$$f_a(\boldsymbol{\sigma}_a = (\sigma_{ab}, \cdots)) \to \sum_{\sigma'_{ab}} G_{ab}\left(\sigma_{ab}, \sigma'_{ab}\right) f_a(\sigma'_{ab}, \cdots)$$

$$\sum_{\sigma_{ab}} G_{ab}(\sigma_{ab}, \sigma') G_{ba}(\sigma_{ab}, \sigma'') = \delta(\sigma', \sigma'')$$

The partition function is invariant under any G-gauge!

$$Z = \sum_{\boldsymbol{\sigma}} \prod_a f_a\left(\boldsymbol{\sigma}_a\right) = \underbrace{\sum_{\boldsymbol{\sigma}} \prod_a \left(\sum_{\boldsymbol{\sigma}'_a} f_a(\boldsymbol{\sigma}'_a) \prod_{b \in a} G_{ab}(\sigma_{ab}, \sigma'_{ab}) \right)}_{\text{graphical trace}}$$

Belief Propagation as a Gauge Fixing Condition

$$Z=\sum_{\boldsymbol\sigma}\prod_a f_a(\boldsymbol\sigma_a)=\sum_{\boldsymbol\sigma}\prod_a\left(\sum_{\boldsymbol\sigma'_a} f_a(\boldsymbol\sigma'_a)\prod_{b\in a} G_{ab}(\sigma_{ab},\sigma'_{ab})\right)$$

$$Z=\underbrace{Z_0(G)}_{\text{ground state } \boldsymbol\sigma=\mathbf{0}}+\underbrace{\sum_{\text{all possible colorings of the graph}} Z_c(G)}_{\boldsymbol\sigma\neq\mathbf{0},\ \text{excited states}}$$

Belief Propagation Gauge

$$\underline{\forall a\ \&\ \forall b\in a:}\quad \sum_{\boldsymbol\sigma'_a} f_a(\boldsymbol\sigma')G^{(bp)}_{ab}(\sigma_{ab}\neq 0,\sigma'_{ab})\prod_{c\in a}^{c\neq b} G^{(bp)}_{ac}(0,\sigma'_{ac})=0$$

No loose colored edges at any vertex of the graph!

Belief Propagation Equations: $\forall a$ & $\forall b \in a$:

$$\left\{\begin{array}{l} \sum_{\sigma'_a} f_a(\sigma') G^{(bp)}_{ab}(\sigma_{ab} \neq 0, \sigma'_{ab}) \prod_{c\in a}^{c\neq b} G^{(bp)}_{ac}(0, \sigma'_{ac}) = 0 \\ \sum_{\sigma_{ab}} G_{ab}(\sigma_{ab}, \sigma') G_{ba}(\sigma_{ab}, \sigma'') = \delta(\sigma', \sigma'') \end{array}\right. \Rightarrow \left\{\begin{array}{l} G^{(bp)}_{ba}(0, \sigma'_{ab}) = \rho_a^{-1} \sum_{\sigma'_a \setminus \sigma'_{ab}} f_a(\sigma') \prod_{c\in a}^{c\neq b} G^{(bp)}_{ac}(0, \sigma'_{ac}) \\ \rho_a = \sum_{\sigma'_a} f_a(\sigma') \prod_{c\in a} G^{(bp)}_{ac}(0, \sigma'_{ac}) \end{array}\right.$$

Belief Propagation in terms of Messages

$$\epsilon_{ab}(\sigma) = G_{ab}(0, \sigma) = \frac{\exp\left(\eta_{ab}(\sigma)\right)}{\sum_\sigma \exp\left(\eta_{ab}(\sigma) + \eta_{ba}(\sigma)\right)}$$

$$\frac{\exp\left(\eta^{(bp)}_{ab}(\sigma_{ab})\right)}{\sum_{\sigma_{ab}} \exp\left(\eta^{(bp)}_{ab}(\sigma_{ab}) + \eta^{(bp)}_{ba}(\sigma_{ab})\right)} = \frac{\sum_{\sigma_a \setminus \sigma_{ab}} f_a(\sigma_a) \exp\left(\sum_{b\in a} \eta^{(bp)}_{ab} \sigma_{ab}\right)}{\sum_{\sigma_a} f_a(\sigma_a) \exp\left(\sum_{b\in a} \eta^{(bp)}_{ab}(\sigma_{ab})\right)}$$

LDPC case, binary alphabet: $\eta^{bp}_{\alpha j} = h_j + \sum_{\beta\neq\alpha}^{j\in\beta} \tanh^{-1}\left(\prod_{i\neq j}^{i\in\beta} \tanh \eta^{bp}_{\beta i}\right)$

- h_j is log-likelihood on bit j
- Message Passing = Iterative BP
- Message Passing is not guaranteed to converge

Loop Series. Binary Alphabet.

Exact (!!) expression in terms of BP

$$Z = \sum_{\boldsymbol{\sigma}_\sigma} \prod_a f_a(\boldsymbol{\sigma}_a) = Z_0 \left(1 + \sum_C r(C)\right)$$

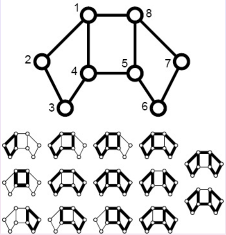
$$r(C) = \frac{\prod\limits_{a \in C} \mu_a}{\prod\limits_{(ab) \in C} (1 - m_{ab}^2)} = \prod_{a \in C} \tilde{\mu}_a$$

$C \in$ Generalized Loops = Loops without loose ends

$$m_{ab} = \int d\boldsymbol{\sigma}_a b_a^{(bp)}(\boldsymbol{\sigma}_a)(1 - 2\sigma_{ab})$$

$$\mu_a = \int d\boldsymbol{\sigma}_a b_a^{(bp)}(\boldsymbol{\sigma}_a) \prod_{b \in a, C} (1 - 2\sigma_{ab} - m_{ab})$$

$$b_a^{(bp)}(\boldsymbol{\sigma}_a) = \frac{f_a(\boldsymbol{\sigma}_a) \exp\left(\sum_{b \in a} \eta_{ab}^{(bp)}(\sigma_{ab})\right)}{\sum_{\boldsymbol{\sigma}_a} f_a(\boldsymbol{\sigma}_a) \exp\left(\sum_{b \in a} \eta_{ab}^{(bp)}(\sigma_{ab})\right)}$$

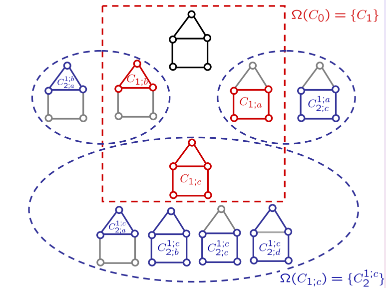


- The Loop Series is finite
- All terms in the series are calculated within BP
- BP is exact on a tree
- BP is a Gauge fixing condition. Other choices of Gauges would lead to different representation.

Loop Tower for q-ary alphabet

$$Z_{C_0} = \sum_{\boldsymbol{\sigma}_{C_0}} \bar{p}(G|\boldsymbol{\sigma}_{C_0}) = Z_{0;C_0} + \sum_{C_1 \in \Omega(C_0)} Z_{C_1}$$

$$Z_{C_1} = \sum_{\boldsymbol{\sigma}_{C_1}} \bar{p}(G^{(bp)}|\boldsymbol{\sigma}_{C_1})$$



- $\sigma_{ab;C_1} = 1, \cdots, q-1$ = not fixed at $q > 2$
- Z_{C_1} is a partition function over reduced alphabet
- Freedom in selection of colored/excited gauges at $q > 2$; $\{G^{(bp)}_{ab;C_0}(\sigma_{ab}, \sigma'_{ab}); (ab) \in C_0\}$

Loop Tower =
Embedded set of Loop Series over sequentially reduced alphabets

$$j = 1, \cdots, q-2: \quad Z_{C_j} = Z_{0;C_j} + \sum_{C_{j+1} \in \Omega(C_j)} Z_{C_{j+1}}$$

Variational Principe & Bethe Free Energy

$$Z=\underbrace{Z_0(G)}_{\sigma=0}+\sum_{\sigma\neq 0} Z_c(G),\quad Z_0(G)\Rightarrow \underbrace{Z_0(\epsilon),\quad \epsilon_{ab}(\sigma_{ab})=G_{ab}(0,\sigma_{ab})}_{\text{depends only on the ground state gauges}}$$

Variational formulation of Belief Propagation

$$\left.\frac{\partial Z_0(\epsilon)}{\partial \epsilon_{ab}(\sigma_{ab})}\right|^{(bp)}=0\quad\Leftrightarrow\quad \text{Belief Propagation Equations}$$

$\mathcal{F}_0(\epsilon)=-\ln Z_0(\epsilon)$ is directly related to the Bethe Free Energy of *Yedidia, Freeman, Weiss '01* Bethe Free Energy

Dilute Gas of Loops: $Z = Z_0(1+\sum_C r_C) \approx Z_0 \cdot \prod_{C_{sc}=\text{single connected}} (1+r_{sc})$

Applies to

- Lattice problems in high spatial dimensions
- Large Erdös-Renyi problems (random graphs with controlled connectivity degree)

- The approximation allows an easy multi-scale re-summation
- In the para-magnetic phase and $\mathbf{h}=0$: the only solution of BP is a trivial one $\boldsymbol{\eta}=0$, $Z_0 \to 1$, and the Loop Series is reduced to the high-temperature expansion [Domb, Fisher, et al '58-'90]

Ising model in the factor graph terms

$$Z=\sum_{\boldsymbol{\sigma}} \prod_{\alpha=(i,j)\in X} \exp\left(J_{ij}\sigma_i\sigma_j\right) = \sum_{\boldsymbol{\sigma}} \prod_{a\in\{i\}\cup\{\alpha\}} f_a(\boldsymbol{\sigma}_a)$$

$$f_i(\boldsymbol{\sigma}_i)=\begin{cases} \exp(h_i\sigma_i), & \sigma_{i\alpha}=\sigma_{i\beta}=\sigma_i \quad \forall\alpha,\beta\ni i \\ 0, & \text{otherwise;} \end{cases}$$

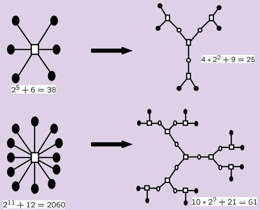
$$f_\alpha\left(\boldsymbol{\sigma}_\alpha=(\sigma_{\alpha i},\sigma_{\alpha j})\right)=\exp\left(J_{ij}\sigma_{\alpha i}\sigma_{\alpha j}\right)$$

Loop Series trivially pass the common "loop" tests (from Rizzo, Montanari '05)

- Evaluation of the critical temperature in the constant exchange, zero field Ising model
- Leading $1/N$ corrections to the Free Energy of the Viana-Bray model in the vicinity of the critical point (glass transition)

Dendro Trick

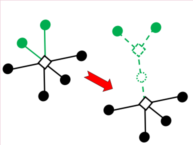
Dendro-trick = Graph Modification   Chertkov,Stepanov'07



- MAP solutions are identical
- Set of Pseudo-codewords are identical
- Instanton spectra are very alike, $\approx$

$$Z=\sum_{\boldsymbol{\sigma}}\left(\prod_{\alpha}\delta\left(\prod_{i\in\alpha}\sigma_i,+1\right)\right)\left(\exp\left(\sum_i h_i\sigma_i\right)\right)$$

$$\delta\left(\prod_{i\in\alpha}\sigma_i,+1\right)=\sum_{\bar\sigma=\pm1}\delta\left(\bar\sigma\prod_{i\in\alpha}^{i\neq j,k}\sigma_i,+1\right)\delta\left(\bar\sigma\sigma_j\sigma_k,+1\right)$$



Self-avoiding Tree — Weitz '06

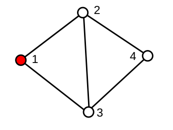
Bipartite. Binary.

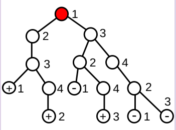
$$\mathcal{P}(\boldsymbol{\sigma}) = Z^{-1} \prod_{(i,j)} f(\sigma_i, \sigma_j)$$

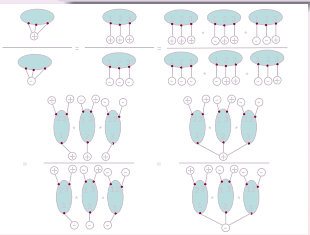
$$p_i(\sigma_i) = \sum_{\boldsymbol{\sigma}\setminus\sigma_i} \mathcal{P}(\boldsymbol{\sigma})$$

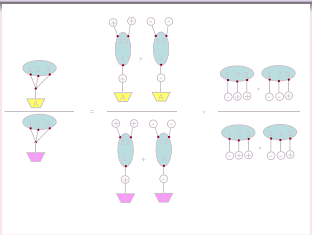
$$\frac{p_i(+)}{p_i(-)} = \frac{\sum_{\boldsymbol{\sigma}\setminus\sigma_i}\left(\prod_{(k,j)} f(\sigma_k,\sigma_j)\right)\Big|_{\sigma_i=+}}{\sum_{\boldsymbol{\sigma}\setminus\sigma_i}\left(\prod_{(k,j)} f(\sigma_k,\sigma_j)\right)\Big|_{\sigma_i=-}}$$

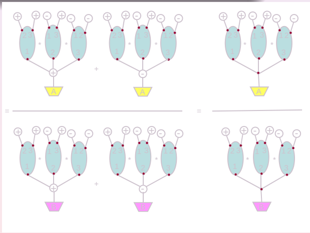
$\Rightarrow$











One can carry the transformations over ... or stop at an intermediate step

Self-avoiding Tree

Weitz '06

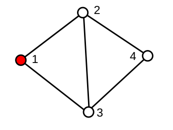
Bipartite. Binary.

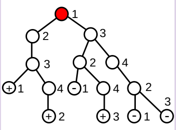
$$\mathcal{P}(\boldsymbol{\sigma}) = Z^{-1} \prod_{(i,j)} f(\sigma_i, \sigma_j)$$

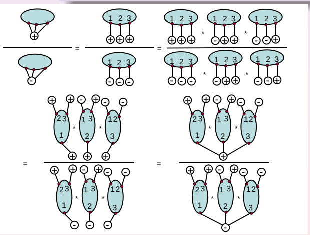
$$p_i(\sigma_i) = \sum_{\boldsymbol{\sigma}\setminus\sigma_i} \mathcal{P}(\boldsymbol{\sigma})$$

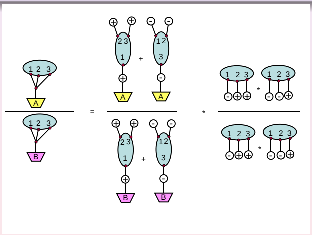
$$\frac{p_i(+)}{p_i(-)} = \frac{\sum_{\boldsymbol{\sigma}\setminus\sigma_i}\left(\prod_{(k,j)} f(\sigma_k,\sigma_j)\right)\Big|_{\sigma_i=+}}{\sum_{\boldsymbol{\sigma}\setminus\sigma_i}\left(\prod_{(k,j)} f(\sigma_k,\sigma_j)\right)\Big|_{\sigma_i=-}}$$

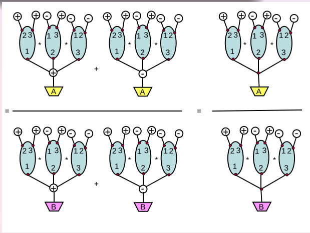
$\Rightarrow$











One can carry the transformations over ... or stop at an intermediate step

Self-avoiding Tree

Weitz '06

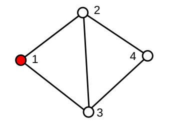
Bipartite. Binary.

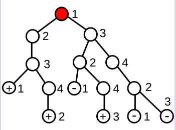
$$\mathcal{P}(\boldsymbol{\sigma}) = Z^{-1} \prod_{(i,j)} f(\sigma_i, \sigma_j)$$

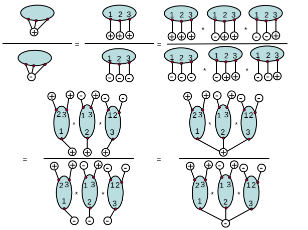
$$p_i(\sigma_i) = \sum_{\boldsymbol{\sigma}\setminus\sigma_i} \mathcal{P}(\boldsymbol{\sigma})$$

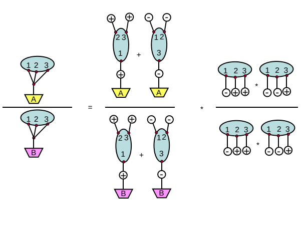
$$\frac{p_i(+)}{p_i(-)} = \frac{\sum_{\boldsymbol{\sigma}\setminus\sigma_i}\left(\prod_{(k,j)} f(\sigma_k,\sigma_j)\right)\Big|_{\sigma_i=+}}{\sum_{\boldsymbol{\sigma}\setminus\sigma_i}\left(\prod_{(k,j)} f(\sigma_k,\sigma_j)\right)\Big|_{\sigma_i=-}}$$

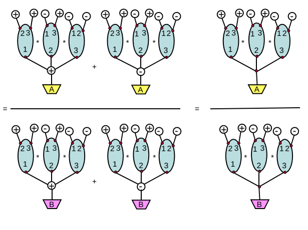
$\Rightarrow$











One can carry the transformations over ... or stop at an intermediate step

Complementarity of Loop Calculus & Graphical Transformations

Speculations

- Loop Calculus is built on Gauge Transformations. Gauge Transformations do not change the graph but reparametrize factor functions.
- Graphical Transformations keep factor functions but modify the graph.
- Loop Calculus & Graphical Transformations are complementary.
- It may be advantageous to build efficient optimality achieving algorithms on the combination of the two: the Loop Calculus and the Graphical Transformations.

Complementarity of Loop Calculus & Graphical Transformations

Speculations

- Loop Calculus is built on Gauge Transformations. Gauge Transformations do not change the graph but reparametrize factor functions.
- Graphical Transformations keep factor functions but modify the graph.
- Loop Calculus & Graphical Transformations are complementary.
- It may be advantageous to build efficient optimality achieving algorithms on the combination of the two: the Loop Calculus and the Graphical Transformations.

Complementarity of Loop Calculus & Graphical Transformations

Speculations

- Loop Calculus is built on Gauge Transformations. Gauge Transformations do not change the graph but reparametrize factor functions.
- Graphical Transformations keep factor functions but modify the graph.
- Loop Calculus & Graphical Transformations are complementary.
- It may be advantageous to build efficient optimality achieving algorithms on the combination of the two: the Loop Calculus and the Graphical Transformations.

Complementarity of Loop Calculus & Graphical Transformations

Speculations

- Loop Calculus is built on Gauge Transformations. Gauge Transformations do not change the graph but reparametrize factor functions.
- Graphical Transformations keep factor functions but modify the graph.
- Loop Calculus & Graphical Transformations are complementary.
- It may be advantageous to build efficient optimality achieving algorithms on the combination of the two: the Loop Calculus and the Graphical Transformations.

Summary (third lecture)

- Loop Calculus allows straightforward generalization to a general q-ary case.
- Loop Tower generalizes Loop Series for the q-ary case. It is a sequential construction with a new freedom in selecting the "excited" gauges at $q > 2$.
- BP equations are conditions on the "ground state" gauges. Bethe Free Energy and effective functional of the "ground state" term are in direct relation.
- Statistical Physics near second order phase transition represents a situation where many loops, of different sizes, become equally important.
- Graphical Transformations complements Loop Calculus

All papers are available at http://cnls.lanl.gov/∼chertkov/pub.htm

Bibliography

Results

- BP is better then just a heuristic in the loopy case ... BP is the special Gauge condition eliminating all contributions but loops.
- Exact Marginal probability allows explicit Loop Series expression in terms of a solution of the Belief Propagation equations.
- Truncation and/or Re-summation of the Loop Series provide hierarchy of systematically improvable approximations/algorithms. Standard BP/LP is a first member in the hierarchy.
- Finding a critical loop, or a small number of critical loops, is algorithmically sufficient for reducing effect of the decoding sub-optimality in the error-floor domain.
- Loop Calculus allows generalization to q-ary alphabet.

Results

- BP is better then just a heuristic in the loopy case ... BP is the special Gauge condition eliminating all contributions but loops.
- Exact Marginal probability allows explicit Loop Series expression in terms of a solution of the Belief Propagation equations.
- Truncation and/or Re-summation of the Loop Series provide hierarchy of systematically improvable approximations/algorithms. Standard BP/LP is a first member in the hierarchy.
- Finding a critical loop, or a small number of critical loops, is algorithmically sufficient for reducing effect of the decoding sub-optimality in the error-floor domain.
- Loop Calculus allows generalization to q-ary alphabet.

Results

- BP is better then just a heuristic in the loopy case ... BP is the special Gauge condition eliminating all contributions but loops.
- Exact Marginal probability allows explicit Loop Series expression in terms of a solution of the Belief Propagation equations.
- Truncation and/or Re-summation of the Loop Series provide hierarchy of systematically improvable approximations/algorithms. Standard BP/LP is a first member in the hierarchy.
- Finding a critical loop, or a small number of critical loops, is algorithmically sufficient for reducing effect of the decoding sub-optimality in the error-floor domain.
- Loop Calculus allows generalization to q-ary alphabet.

Results

- BP is better then just a heuristic in the loopy case ... BP is the special Gauge condition eliminating all contributions but loops.
- Exact Marginal probability allows explicit Loop Series expression in terms of a solution of the Belief Propagation equations.
- Truncation and/or Re-summation of the Loop Series provide hierarchy of systematically improvable approximations/algorithms. Standard BP/LP is a first member in the hierarchy.
- Finding a critical loop, or a small number of critical loops, is algorithmically sufficient for reducing effect of the decoding sub-optimality in the error-floor domain.
- Loop Calculus allows generalization to q-ary alphabet.

Results

- BP is better then just a heuristic in the loopy case ... BP is the special Gauge condition eliminating all contributions but loops.
- Exact Marginal probability allows explicit Loop Series expression in terms of a solution of the Belief Propagation equations.
- Truncation and/or Re-summation of the Loop Series provide hierarchy of systematically improvable approximations/algorithms. Standard BP/LP is a first member in the hierarchy.
- Finding a critical loop, or a small number of critical loops, is algorithmically sufficient for reducing effect of the decoding sub-optimality in the error-floor domain.
- Loop Calculus allows generalization to q-ary alphabet.

Path Forward – Future Chalenges

- Better Algorithms: Loop Series Truncation/Resummation
- Synthesis of Graphical Transformations and Loop Series. Graphical decoding of dense codes?
- Further generalizations. Continuous alphabets. Quantum spins. Quantum Error-correction and Information Theory.
- Loop calculus based analysis of graph ensembles, e.g. understanding and improving the cavity method [Mézard, Parisi '85-'03]
- Extending the list of Loop Calculus Applications, e.g. SAT and cryptography
- Non-BP gauges, e.g. for stat problems on regular and irregular lattices. Loop calculus for "near easy" problems on dense graphs.
- Relation to graph ζ-functions [Koetter, Li, Vontobel, Walker '05]
- . . .

Path Forward – Future Chalenges

- Better Algorithms: Loop Series Truncation/Resummation
- Synthesis of Graphical Transformations and Loop Series. Graphical decoding of dense codes?
- Further generalizations. Continuous alphabets. Quantum spins. Quantum Error-correction and Information Theory.
- Loop calculus based analysis of graph ensembles, e.g. understanding and improving the cavity method [Mézard, Parisi '85-'03]
- Extending the list of Loop Calculus Applications, e.g. SAT and cryptography
- Non-BP gauges, e.g. for stat problems on regular and irregular lattices. Loop calculus for "near easy" problems on dense graphs.
- Relation to graph ζ-functions [Koetter, Li, Vontobel, Walker '05]
- ...

Path Forward – Future Chalenges

- Better Algorithms: Loop Series Truncation/Resummation
- Synthesis of Graphical Transformations and Loop Series. Graphical decoding of dense codes?
- Further generalizations. Continuous alphabets. Quantum spins. Quantum Error-correction and Information Theory.
- Loop calculus based analysis of graph ensembles, e.g. understanding and improving the cavity method [Mézard, Parisi '85-'03]
- Extending the list of Loop Calculus Applications, e.g. SAT and cryptography
- Non-BP gauges, e.g. for stat problems on regular and irregular lattices. Loop calculus for "near easy" problems on dense graphs.
- Relation to graph ζ-functions [Koetter, Li, Vontobel, Walker '05]
- . . .

Path Forward – Future Chalenges

- Better Algorithms: Loop Series Truncation/Resummation
- Synthesis of Graphical Transformations and Loop Series. Graphical decoding of dense codes?
- Further generalizations. Continuous alphabets. Quantum spins. Quantum Error-correction and Information Theory.
- Loop calculus based analysis of graph ensembles, e.g. understanding and improving the cavity method [Mézard, Parisi '85-'03]
- Extending the list of Loop Calculus Applications, e.g. SAT and cryptography
- Non-BP gauges, e.g. for stat problems on regular and irregular lattices. Loop calculus for "near easy" problems on dense graphs.
- Relation to graph ζ-functions [Koetter, Li, Vontobel, Walker '05]
- ...

Path Forward – Future Chalenges

- Better Algorithms: Loop Series Truncation/Resummation
- Synthesis of Graphical Transformations and Loop Series. Graphical decoding of dense codes?
- Further generalizations. Continuous alphabets. Quantum spins. Quantum Error-correction and Information Theory.
- Loop calculus based analysis of graph ensembles, e.g. understanding and improving the cavity method [Mézard, Parisi '85-'03]
- Extending the list of Loop Calculus Applications, e.g. SAT and cryptography
- Non-BP gauges, e.g. for stat problems on regular and irregular lattices. Loop calculus for "near easy" problems on dense graphs.
- Relation to graph ζ-functions [Koetter, Li, Vontobel, Walker '05]
- ...

Path Forward – Future Chalenges

- Better Algorithms: Loop Series Truncation/Resummation
- Synthesis of Graphical Transformations and Loop Series. Graphical decoding of dense codes?
- Further generalizations. Continuous alphabets. Quantum spins. Quantum Error-correction and Information Theory.
- Loop calculus based analysis of graph ensembles, e.g. understanding and improving the cavity method [Mézard, Parisi '85-'03]
- Extending the list of Loop Calculus Applications, e.g. SAT and cryptography
- Non-BP gauges, e.g. for stat problems on regular and irregular lattices. Loop calculus for "near easy" problems on dense graphs.
- Relation to graph ζ-functions [Koetter, Li, Vontobel, Walker '05]
- ...

Path Forward – Future Chalenges

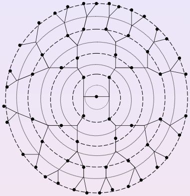
- Better Algorithms: Loop Series Truncation/Resummation
- Synthesis of Graphical Transformations and Loop Series. Graphical decoding of dense codes?
- Further generalizations. Continuous alphabets. Quantum spins. Quantum Error-correction and Information Theory.
- Loop calculus based analysis of graph ensembles, e.g. understanding and improving the cavity method [Mézard, Parisi '85-'03]
- Extending the list of Loop Calculus Applications, e.g. SAT and cryptography
- Non-BP gauges, e.g. for stat problems on regular and irregular lattices. Loop calculus for "near easy" problems on dense graphs.
- Relation to graph ζ-functions [Koetter, Li, Vontobel, Walker '05]
- ...

Path Forward – Future Chalenges

- Better Algorithms: Loop Series Truncation/Resummation
- Synthesis of Graphical Transformations and Loop Series. Graphical decoding of dense codes?
- Further generalizations. Continuous alphabets. Quantum spins. Quantum Error-correction and Information Theory.
- Loop calculus based analysis of graph ensembles, e.g. understanding and improving the cavity method [Mézard, Parisi '85-'03]
- Extending the list of Loop Calculus Applications, e.g. SAT and cryptography
- Non-BP gauges, e.g. for stat problems on regular and irregular lattices. Loop calculus for "near easy" problems on dense graphs.
- Relation to graph ζ-functions [Koetter, Li, Vontobel, Walker '05]
- ...

Thank You !!

BP is Exact on a Tree (LDPC)



$$Z(\mathbf{h}) = \sum_{\boldsymbol{\sigma}} \prod_{\alpha=1}^{M} \delta\left(\prod_{i\in\alpha} \sigma_i, 1\right) \exp\left(\sum_{i=1}^{N} h_i\sigma_i\right)$$

h_i is a log-likelihood at a bit (outcome of the channel)

$$Z^{\pm}_{j\alpha}(\mathbf{h}^{>}) \equiv \sum_{\boldsymbol{\sigma}^{>}}^{\sigma_j=\pm 1} \prod_{\beta>} \delta\left(\prod_{i\in\beta} \sigma_i, 1\right) \exp\left(\sum_{i>} h_i\sigma_i\right)$$

$$Z^{\pm}_{j\alpha} = \exp(\pm h_j) \prod_{\beta\neq\alpha}^{j\in\beta} \frac{1}{2}\left(\prod_{i\neq j}^{i\in\beta}(Z^{+}_{i\beta}+Z^{-}_{i\beta}) \pm \prod_{i\neq j}^{i\in\beta}(Z^{+}_{i\beta}-Z^{-}_{i\beta})\right)$$

$$\eta_{j\alpha} \equiv \frac{1}{2}\ln\left(\frac{Z^{+}_{j\alpha}}{Z^{-}_{j\alpha}}\right), \quad \eta_{j\alpha} = h_j + \sum_{\beta\neq\alpha}^{j\in\beta} \tanh^{-1}\left(\prod_{i\neq j}^{i\in\beta} \tanh \eta_{i\beta}\right)$$

◂ Bethe Free Energy

Gauges and BP equations

Partition function in the colored representation

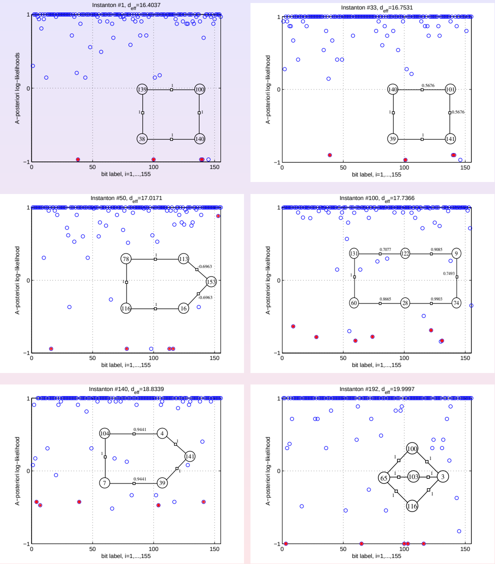
$$Z=(\prod_{bc} 2\cosh(\eta_{bc}+\eta_{cb}))^{-1}\sum_{\boldsymbol{\sigma}'}\prod_a \tilde{f}_a \prod_{bc} V_{bc},\quad \tilde{f}_a(\boldsymbol{\sigma}_a;\boldsymbol{\eta}_a)=f_a(\boldsymbol{\sigma}_a)\prod_{b\in a}\exp(\eta_{ab}\sigma_{ab})$$

$$V_{bc}(\sigma_{bc},\sigma_{cb})=1+(\tanh(\eta_{bc}+\eta_{cb})-\sigma_{bc})(\tanh(\eta_{bc}+\eta_{cb})-\sigma_{cb})\cosh^2(\eta_{bc}+\eta_{cb})$$

Fixing the gauges ⇒ BP equations!!

$$\sum_{\boldsymbol{\sigma}_a}\left(\tanh(\eta_{ab}^{(bp)}+\eta_{ba}^{(bp)})-\sigma_{ab}\right)\tilde{f}_a(\boldsymbol{\sigma}_a;\boldsymbol{\eta}_a)=0 \quad\Rightarrow\quad \underbrace{\eta_{\alpha j}^{bp}=h_j+\sum_{\beta\neq\alpha}^{j\in\beta}\tanh^{-1}\Big(\prod_{i\neq j}^{i\in\beta}\tanh\eta_{\beta i}^{bp}\Big)}_{\text{LDPC case}}$$

◂ Gauges and BP


Instanton #1, d_eff=16.4037
A−posteriori log−likelihoods
bit label, i=1,...,155
139 100 38 140
Instanton #33, d_eff=16.7531
A−posteriori log−likelihood
bit label, i=1,...,155
140 101 39 141 0.5676
Instanton #50, d_eff=17.0171
A−posteriori log−likelihood
bit label, i=1,...,155
78 113 153 116 16 -0.6963
Instanton #100, d_eff=17.7366
A−posteriori log−likelihood
bit label, i=1,...,155
131 122 9 60 28 74 0.7077 0.9085 0.7493 0.8665 0.9903
Instanton #140, d_eff=18.8339
A−posteriori log−likelihood
bit label, i=1,...,155
104 4 141 7 39 0.9441
Instanton #192, d_eff=19.9997
A−posteriori log−likelihood
bit label, i=1,...,155
100 65 103 3 116

Back

Relation to the Bethe Free Energy approach

in the spirit of Yedidia, Freeman, Weiss '01

Minimize: $\Phi_B = \sum_a \sum_{\boldsymbol{\sigma}_a} b_a(\boldsymbol{\sigma}_a) \ln\left(\frac{b_a(\boldsymbol{\sigma}_a)}{f_a(\boldsymbol{\sigma}_a)}\right) - \sum_{(ab)} \sum_{\sigma_{ab}} b_{ab}(\sigma_{ab}) \ln b_{ab}(\sigma_{ab})$

under the conditions: $\forall a$ & $\forall c \in a$

$$0 \leq b_a(\boldsymbol{\sigma}_a), b_{ac}(\sigma_{ac}) \leq 1$$
$$\sum_{\boldsymbol{\sigma}_a} b_a(\boldsymbol{\sigma}_a) = 1$$
$$b_{ac}(\sigma_{ac}) = \sum_{\boldsymbol{\sigma}_a \setminus \sigma_{ac}} b_a(\boldsymbol{\sigma}_a)$$

- $\mathcal{L}_B = \Phi_B + \sum_{(ab)} [\sum_{\sigma_{ab}} \ln(\epsilon_{ab}(\sigma_{ab}))(b_{ab}(\sigma_{ab}) - \sum_{\boldsymbol{\sigma}_a \setminus \sigma_{ab}} b_a(\boldsymbol{\sigma}_a)) + \sum_{\sigma_{ba}} \ln(\epsilon_{ba}(\sigma_{ba}))(b_{ab}(\sigma_{ba}) - \sum_{\boldsymbol{\sigma}_b \setminus \sigma_{ba}} b_b(\boldsymbol{\sigma}_b))]$
- Finding extremum of the Bethe Lagrangian with respect to beliefs, b_{ab} and b_a and expressing the result in terms of ϵ: $\mathcal{L}_B(b, \epsilon) \Rightarrow \mathcal{F}_B(\epsilon)$
- $\mathcal{F}_B(\epsilon)\big|_{\{\forall(a,b):\ \sum_{\sigma_{ab}} \epsilon_{ab}(\sigma_{ab})\epsilon_{ba}(\sigma_{ab})=1\}} = \mathcal{F}_0(\epsilon) = -\ln(Z(\epsilon))$

Variational approach

Bibliography (A)

Loop Calculus, Series, Tower

- V. Chernyak and M. Chertkov, "Loop Calculus and Belief Propagation for q-ary Alphabet: Loop Tower," *Proceedings of IEEE ISIT 2007*, June 2007, Nice, arXiv:cs.IT/0701086.
- M. Chertkov and V. Chernyak, "Loop series for discrete statistical models on graphs," JSTAT/2006/P06009, arXiv:cond-mat/0603189.
- M. Chertkov and V. Chernyak, "Loop Calculus in Statistical Physics and Information Science," *Phys. Rev. E*, **73**, 065102(R) (2006), arXiv:cond-mat/0601487.

Reducing the Error Floor

- M. Chertkov, "Reducing the Error Floor", invited talk at the *Information Theory Workshop '07 on "Frontiers in Coding"*, September 2-6, 2007.
- M. Chertkov and V. Chernyak, "Loop Calculus Helps to Improve Belief Propagation and Linear Programming Decodings of Low-Density-Parity-Check Codes," invited talk at 44^{th} *Allerton Conference*, September 27-29, 2006, Allerton, IL, arXiv:cs.IT/0609154.

All papers are available at http://cnls.lanl.gov/∼chertkov/pub.htm

Bibliography (B)

Instantons & Pseudo-Codewords. Analyzing the Error-Floor.

- M. Chertkov and M. Stepanov, "Searching for low weight pseudo-codewords," invited talk at the *2007 Information Theory and Application Workshop, proceedings*, ITA CALIT2, UCSD, `arXiv:cs.IT/0702024`.
- M. Chertkov and M. Stepanov, "Pseudo-codeword Landscape," *Proceedings of IEEE ISIT 2007*, June 2007, Nice, `arXiv:cs.IT/0701084`.
- M. Chertkov and M. Stepanov, "An Efficient Pseudo-Codeword-Search Algorithm for Linear Programming Decoding of LDPC Codes," to appear in *IEEE Transactions on Information Theory*, `arXiv:cs.IT/0601113`.
- M. Stepanov and M. Chertkov, "Instanton analysis of Low-Density-Parity-Check codes in the error-floor regime," *Proceeding of IEEE ISIT 2006*, July 2006 Seattle, `arXiv:cs.IT/0601070`.
- M. Stepanov and M. Chertkov, "The error-floor of LDPC codes in the Laplacian channel," Proceedings of 43rd Allerton Conference (September 28-30, 2005, Allerton, IL), `arXiv:cs.IT/0507031`.
- M. Stepanov, V. Chernyak, M. Chertkov and B. Vasic, "Diagnosis of weakness in error correction: a physics approach to error floor analysis," Phys. Rev. Lett. **95**, 228701 (2005),`cond-mat/0506037`.
- V. Chernyak, M. Chertkov, M. Stepanov and B. Vasic, "Error correction on a tree: An instanton approach" , Phys. Rev. Lett. **93**, 198702-1 (2004).

Other subjects related to LDPC+ decoding

- J. A. Anguita, M. Chertkov, B. Vasic and M. A. Neifeld, "Bethe-Free-Energy Based Decoding of Low-Density Parity-Check Codes on Partial Response Channels," submitted to *IEEE Journal of Selected Areas in Communications*.
- M. Stepanov and M. Chertkov, "Improving convergence of belief propagation decoding," *Proceedings of 44th Allerton Conference*, September 27-29, 2006, Allerton, IL, `arXiv:cs.IT/0607112`.