Advanced mean filed annealing

Integer programming

- Objective functions are not differentiate

$$E(S) = -\frac{1}{2}\sum_{\langle i,j\rangle} S_i J_{ij} S_j + \sum_i b_i S_i$$

$$S_i \in \{+1, -1\}$$

Naive MFE

① $\Pr(s_i) \propto \exp(\beta u_i s_i)$

② $\langle s_i \rangle = \tanh(\beta u_i)$

③ $$\mathcal{F}(v) = -\frac{1}{2}\sum_{\langle i,j \rangle} v_i J_{ij} v_j + \sum_i b_i v_i - \sum_{s_i} \Pr(s_i) \log \Pr(s_i)$$

Problems of integer programming

- Graph bisection
- TSP
- Sudoku solver
- Extension to mixed integer programming

Mixed integer programming

- Clustering
- Self-organization
- Function approximation
- Classification
- Set-valued mapping approximation
- One-classification

PUMA ROBOT

- Puma toolbox

Advanced Mean field annealing

- Asynchronous updating
- Naive mean field equations
- TAP equations
- Two sets of interactive MFES

- Two sets of interactive dynamics

The simulation procedure for the interactive mean field equations (28)–(31) is as follows.

Step 1) Initialize β, A, k, and annealing factor γ.

Step 2) Randomly initialize $\{m_i\}$ and $\{v_{ij}\}$ as values near zero.

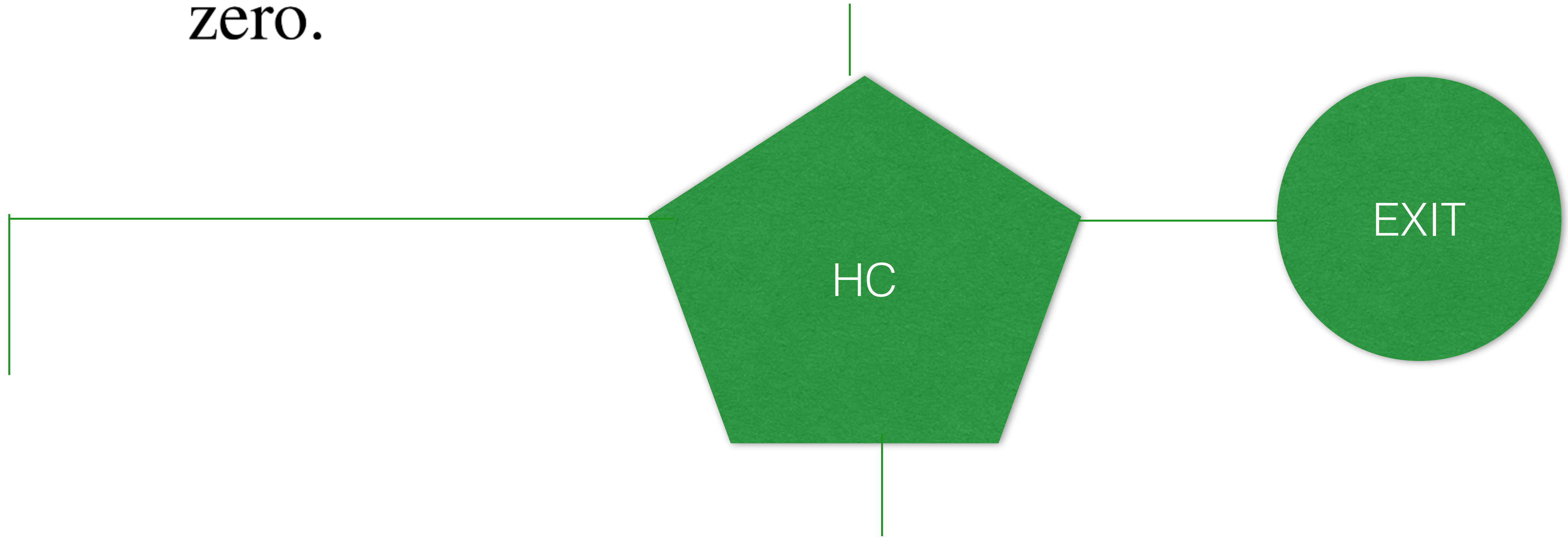
Step 3) Iteratively update $\{m_i\}$ using (28)–(29) to a stationary point.

Step 4) Iteratively update $\{v_{ij}\}$ using (30)–(31) to a stationary point.

Step 5) if both $(1/N)\sum_i m_i^2$ and $(2/(N(N+1)))\sum_{i,j\neq i} v_{ij}^2$ are larger than a threshold value, then halt, otherwise set β as (β/γ), and then, go to step 3.

Step 1) Initialize β, A, k, and annealing factor γ.

Step 2) Randomly initialize $\{m_i\}$ and $\{v_{ij}\}$ as values near zero.



Step 3) Iteratively update $\{m_i\}$ using (28)–(29) to a stationary point.

Step 4) Iteratively update $\{v_{ij}\}$ using (30)–(31) to a stationary point.

J_ij

$$u_i = \sum_{j \neq i} (T_{ij} - A) m_j + \frac{k}{\beta} \sum_{j \neq i} (v_{ij} - m_i m_j) m_j \tag{28}$$

$$m_i = \tanh(\beta u_i) \tag{29}$$

$$w_{ij} = \frac{1}{2}(T_{ij} - A) + \frac{k}{2\beta}(m_i m_j - v_{ij}) \tag{30}$$

$$v_{ij} = \tanh(\beta w_{ij}). \tag{31}$$

Sinkhorn solves sudoku

Mixed integer programming

$$F(m\,|\,v) = -\frac{\beta}{2}\sum_{i,j\neq i} J_{ij} m_i m_j + \sum_i \left(\frac{1+m_i}{2}\ln\frac{1+m_i}{2} + \frac{1-m_i}{2}\ln\frac{1-m_i}{2}\right) + \frac{k}{4}\sum_{i,j\neq i}(m_i m_j - v_{ij})^2. \tag{17}$$

J_ij^2

$m_i \equiv \langle S_i \rangle$

$\xi_{ij} \sim S_i S_j$

$v_{ij} \equiv \langle \xi_{ij} \rangle$

$$F(v\,|\,m) = -\frac{\beta}{2}\sum_{i,j\neq i} J_{ij} v_{ij} + \sum_{i,j\neq i} \left(\frac{1+v_{ij}}{2}\ln\frac{1+v_{ij}}{2} + \frac{1-v_{ij}}{2}\ln\frac{1-v_{ij}}{2}\right) + \frac{k}{4}\sum_{i,j\neq i}(m_i m_j - v_{ij})^2. \tag{21}$$

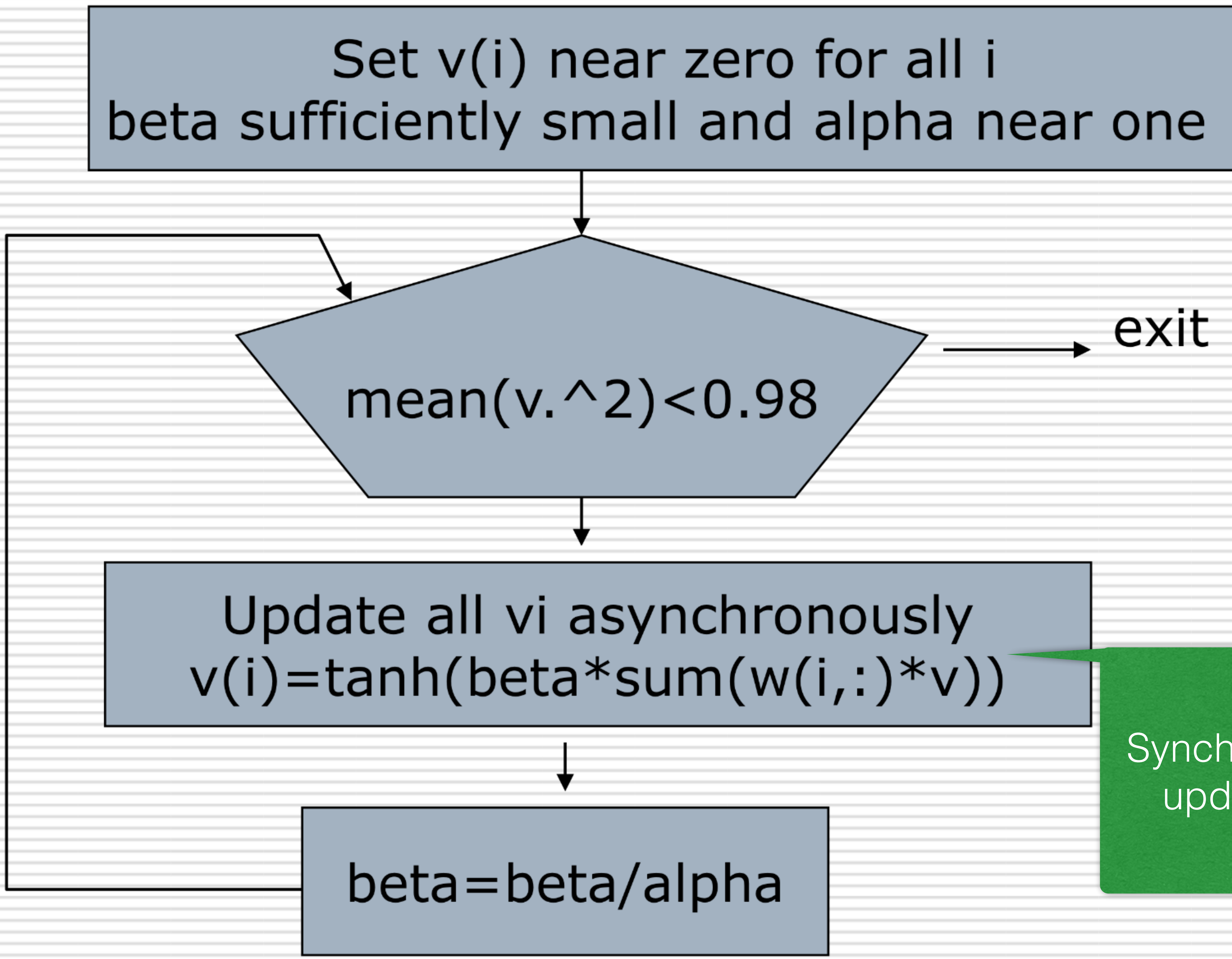
J_ij^2

① $$\Pr(\xi_{ij}) \propto \exp(\beta W_{ij} \xi_{ij})$$

② $$v_{ij} \equiv \langle \xi_{ij} \rangle = \tanh(\beta J_{ij})$$

③ $$F(v|m) = -\frac{1}{2}\sum_{\langle i,j \rangle} J_{ij} v_{ij} + \sum_{\langle i,j \rangle} \Pr(\xi_{ij}) \log \Pr(\xi_{ij})$$

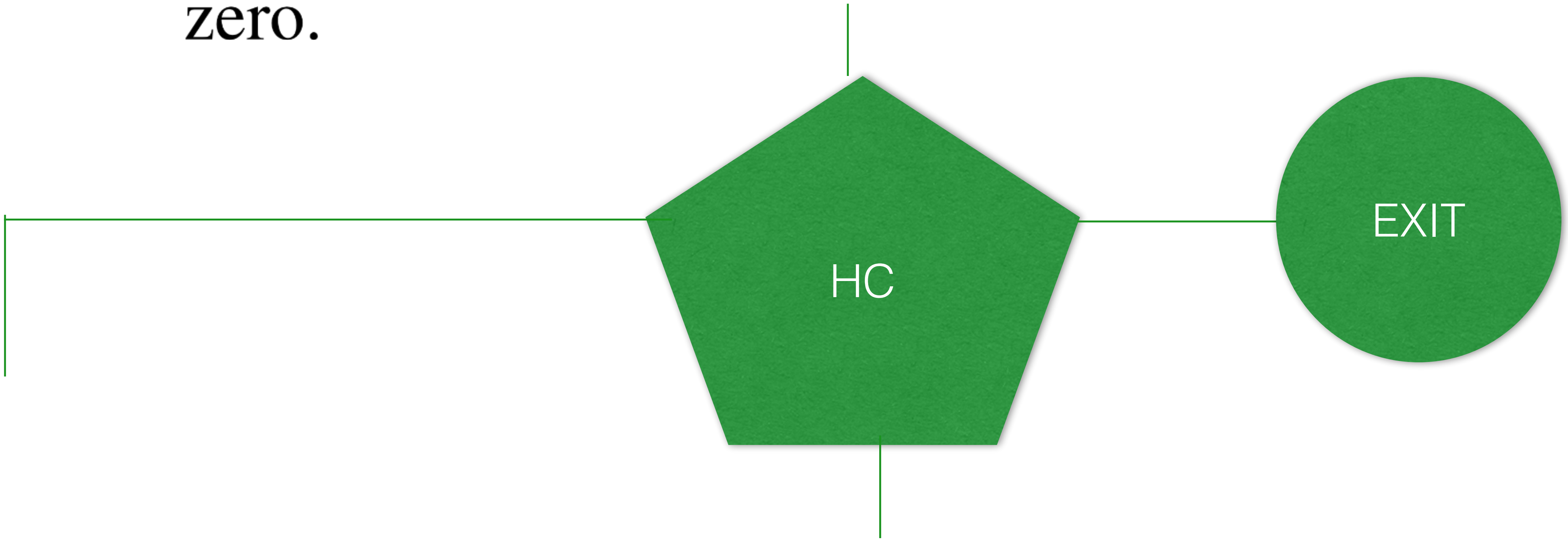
Flow chart



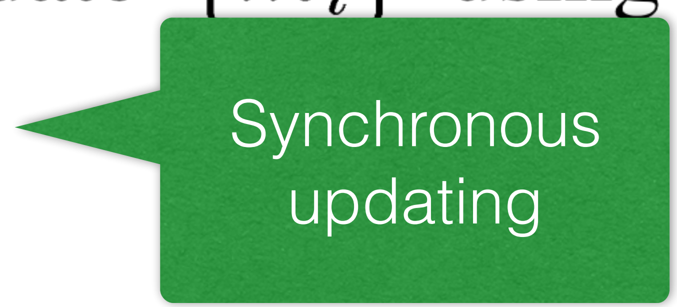
Synchronous updating

Step 1) Initialize β, A, k, and annealing factor γ.

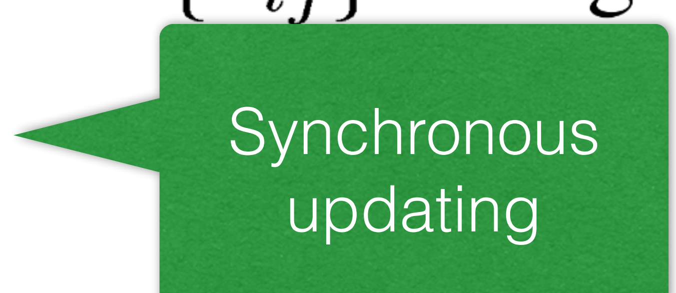
Step 2) Randomly initialize $\{m_i\}$ and $\{v_{ij}\}$ as values near zero.



Step 3) Iteratively update $\{m_i\}$ using (28)–(29) to a stationary point.



Step 4) Iteratively update $\{v_{ij}\}$ using (30)–(31) to a stationary point.



TAP

- <u>From mean field equations and TAP</u>

G_0 and G_1 Free energy of MFES

$$G_0(\mathbf{m}) = \sum_i \left\{ \frac{1+m_i}{2} \ln \frac{1+m_i}{2} + \frac{1-m_i}{2} \ln \frac{1-m_i}{2} \right\} . \quad (53)$$

The calculation of the first order term is also simple, because the first derivative of G at $\lambda = 0$ can be written as an expectation of $H[\mathbf{S}]$ with respect to a factorizing distribution with mean values $\langle S_i \rangle = m_i$. We get

$$G_1(\mathbf{m}) = -\sum_{i<j} J_{ij} m_i m_j . \quad (54)$$

A comparison of the first two terms with (12), (10) and (11) shows that we have already recovered the simple mean field approximation. One can show that the second order term in the expansion is

$$G_2(\mathbf{m}) = -\frac{1}{2} \sum_{ij} J_{ij}^2 (1-m_i)^2 (1-m_j)^2 . \quad (55)$$

$$G_2(\mathbf{m}) = -\frac{1}{2} \sum_{ij} J_{ij}^2 (1 - m_i)^2 (1 - m_j)^2 \ .$$

Strategy

- Apply the Newton method for synchronous updating of mean field annealing
    - Naive MFE ?
    - TAP equation
    - Interactive MF equations

A revision of G2

$$G_2(m) = -\frac{1}{2}\sum_{\langle i,j\rangle} J_{ij}^2 (1-m_i)^2 (1-m_j)^2$$

$$\tilde{G}_2(m) = -\frac{1}{2}\sum_{\langle i,j\rangle} \left(\tanh(\beta J_{ij}) - m_i m_j\right)^2$$