Belief Propagation and Linear Programming - theory and applications

Yair Weiss
School of Computer Science and Engineering
The Hebrew University of Jerusalem
www.cs.huji.ac.il/ ~yweiss

Collaborators: Talya Meltzer, Chen Yanover

Outline

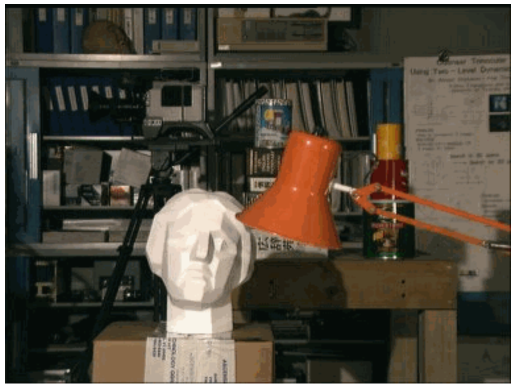
- Energy minimization in computer vision and computational biology.
- Linear Programming relaxations.
- BP and LP - closer than we thought.
- Fixing partially fractional solutions.
- Experimental results.

Pairwise energy minimization

$$E(x) = \sum_{<ij>} E_{ij}(x_i, x_j) + \sum_i E_i(x_i)$$

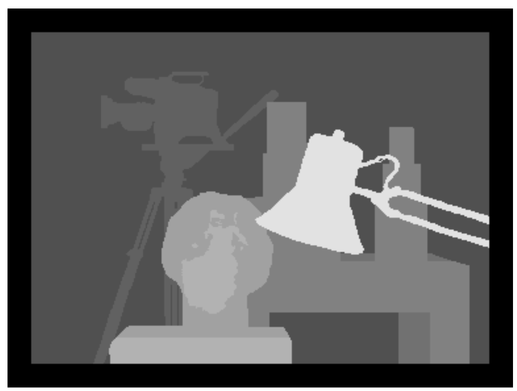
- Stereo vision.
- Side-Chain Prediction.
- Protein Design.

Stereo Vision



Left

Right



Disparity (Tsukuba University)

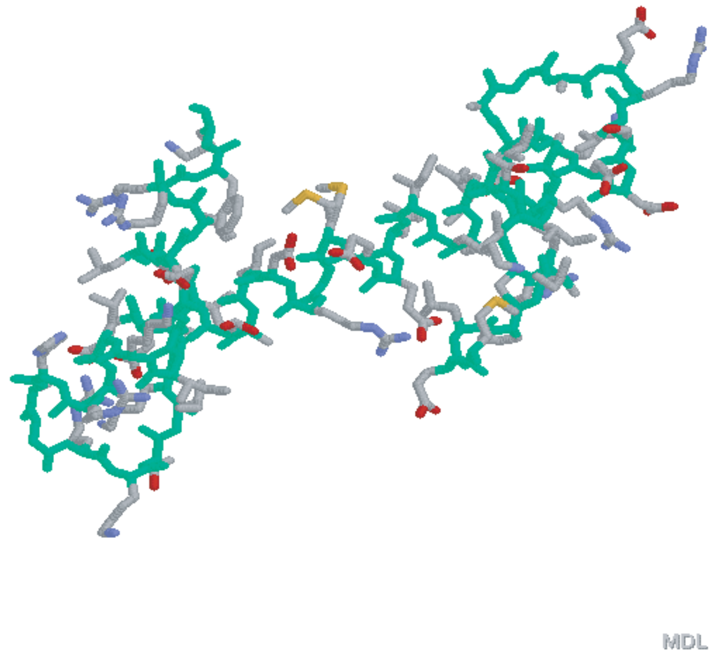
Stereo problem as discrete optimization

x is disparity image. $E(x_i, x_j)$ is compatability cost and $E(x_i)$ is local data cost.

$$x^* = \arg\min_x \sum_{i,j} E_{ij}(x_i, x_j) + \sum_i E_i(x_i)$$

Old formulation (Marr and Poggio 82) but how do we optimize?

Protein Folding

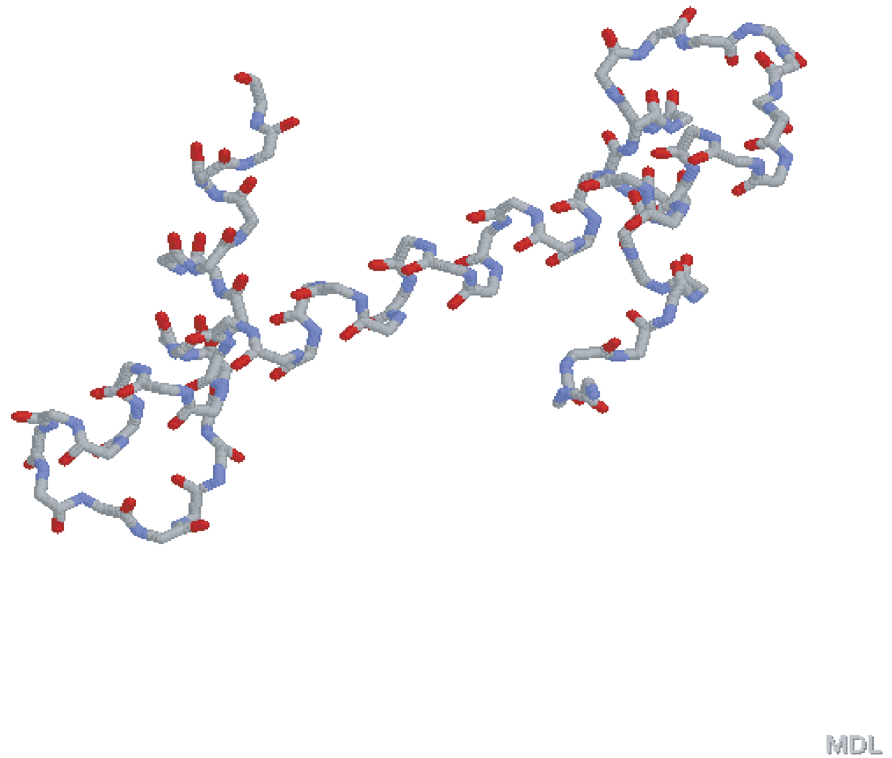


Gene sequence $\Rightarrow$ Amino-acid sequence $\Rightarrow$ 3D structure.

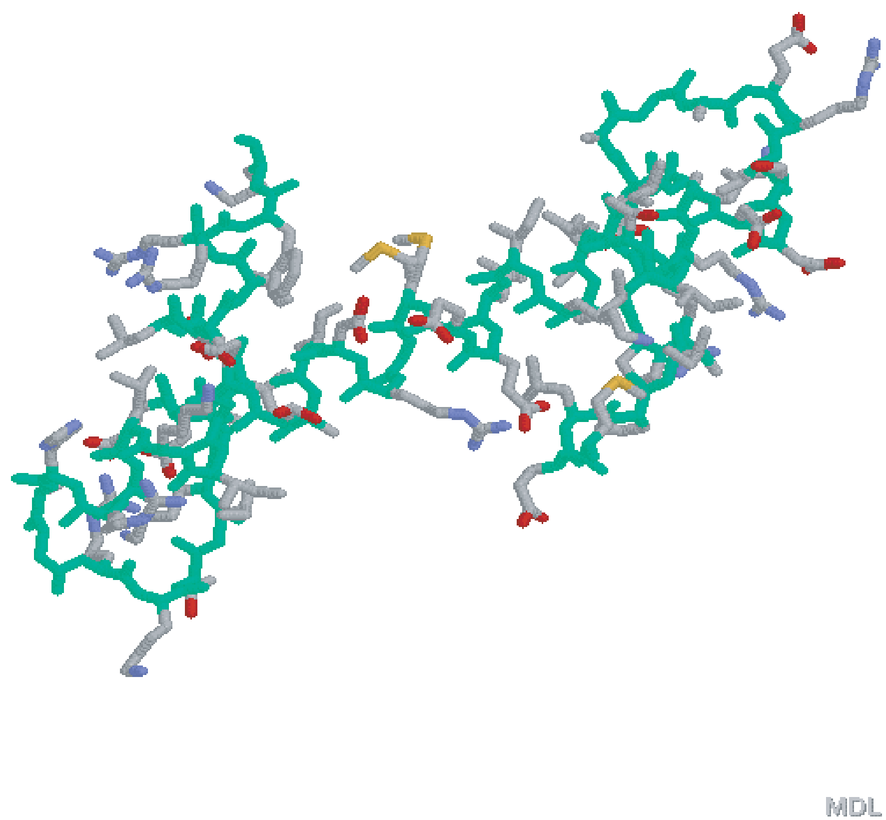
Protein folding in two stages

- Backbone
- Side Chains

Backbone



Backbone plus sidechains



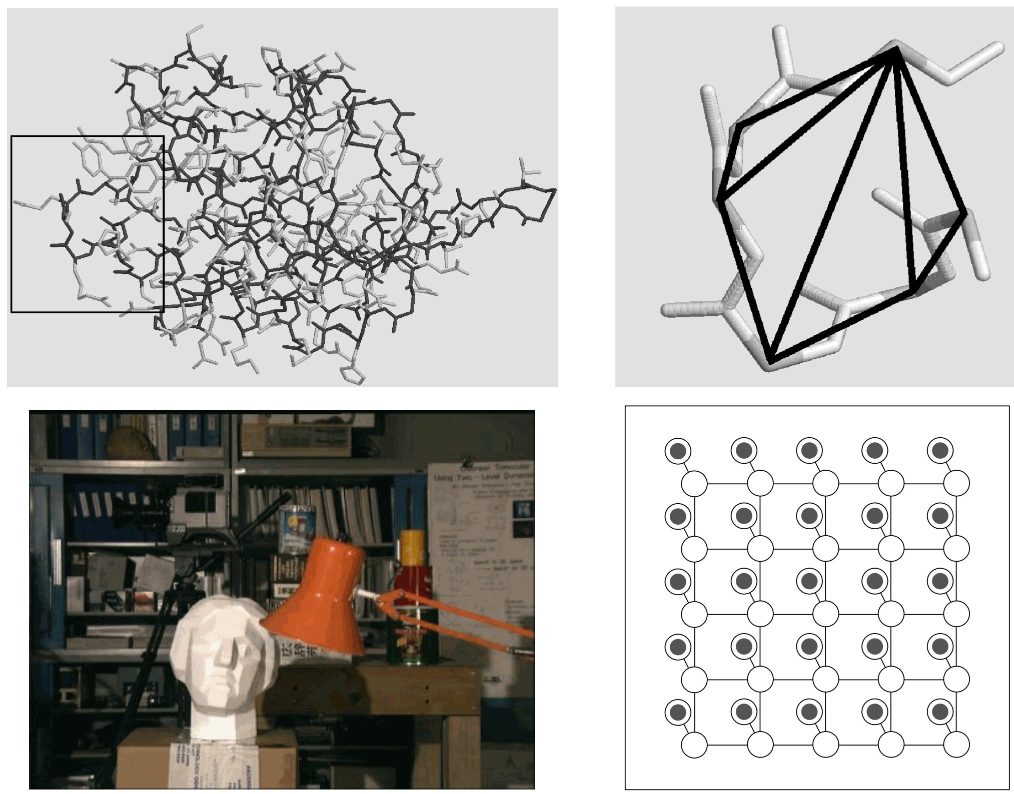
Side chain prediction as combinatorial optimization

- Search space: position of each side chain defined by 4 angles. Each angle is one of 3 possibilities. Search space is 81^n.

- Cost function: local energy and pairwise energies.

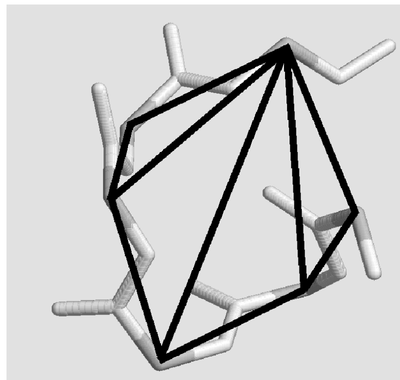
Goal: find set of angles such that energy is minimal.

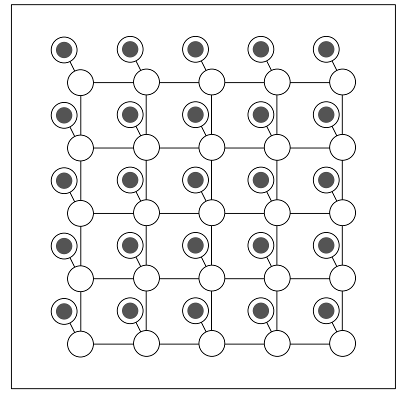
Graphical models



Variables = nodes, edges = pairwise energy term

MAP in graphical models





$$x^* = \arg\min_x \sum_{i,j} E_{ij}(x_i, x_j) + \sum_i E_i(x_i)$$

NP Hard for protein folding and stereo vision $\Rightarrow$ many approximate algorithms.

What's wrong with approximations ?

Best approximate minimizers:

- Stereo vision - unsatisfactory results.
- Side chain prediction - approx. 85%

Better minimizer or better energy functions ?

Linear Programming relaxations

$$J(x) = \sum_{i,j} E_{ij}(x_i, x_j) + \sum_i E_i(x_i)$$

$q_i(x_i), q_{ij}(x_i, x_j)$ are *indicator variables*.

$$J(\{q\}) = \sum_{i,j} \sum_{x_i, x_j} q_{ij}(x_i, x_j) E_{ij}(x_i, x_j) + \sum_i \sum_{x_i} q_i(x_i) E_i(x_i)$$

Integer Programming formulation

minimize:

$$J(\{q\}) = \sum_{i,j} \sum_{x_i,x_j} q_{ij}(x_i,x_j)E_{ij}(x_i,x_j) + \sum_i \sum_{x_i} q_i(x_i)E_i(x_i)$$

subject to:

$$\begin{aligned} q_{ij}(x_i,x_j) &\in \{0,1\} \\ \sum_{x_i,x_j} q_{ij}(x_i,x_j) &= 1 \\ \sum_{x_i} q_{ij}(x_i,x_j) &= q_j(x_j) \end{aligned}$$

Linear Programming relaxation

minimize:

$$J(\{q\}) = \sum_{i,j} \sum_{x_i,x_j} q_{ij}(x_i, x_j) E_{ij}(x_i, x_j) + \sum_i \sum_{x_i} q_i(x_i) E_i(x_i)$$

subject to:

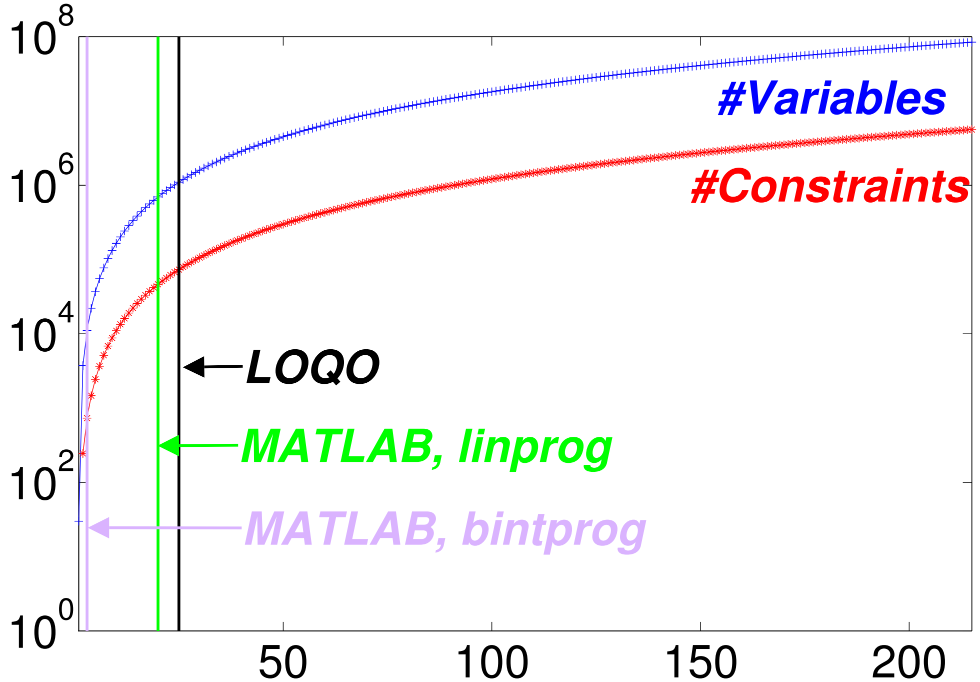
$$\begin{aligned} q_{ij}(x_i, x_j) &\in [0, 1] \\ \sum_{x_i,x_j} q_{ij}(x_i, x_j) &= 1 \\ \sum_{x_i} q_{ij}(x_i, x_j) &= q_j(x_j) \end{aligned}$$

Guarantee of optimality

If the LP solution is integer ($q_{ij}(x_i, x_j) \in \{0, 1\}$ for all i, j) then we have found the global minimum of J.

LP relaxations for vision?

State of the art LP solver can be applied to 39×39 subimage in a machine with 4G memory (34×34 with 2G).



LP using BP

Surprising connection between variants of BP and LP for certain problems (Wainwright, Jaakkola and Willsky 03, Vontobel and Koetter 06, Jung and Shah 07)

Our result

For a large family of BP max-product algorithms (including ordinary BP) and for any graphical model, there exists a BP fixed point such that BP decoding equals the LP decoding.

$$\begin{aligned} x_{BP} &= (0,1,?,0,1,?) \\ x_{LP} &= (0,1,?,0,1,?) \end{aligned}$$

Family of max-product algorithms

$$
\begin{aligned}
m^0_{\alpha i}(x_i) &= \max_{x_{\alpha \setminus i}} f(\alpha)(x_\alpha) \prod_{j \neq i} m_{j\alpha}(x_j) \\
m^0_{i,\alpha}(x_i) &= \prod_{\beta \neq \alpha} m_{\beta i}(x_i) \\
m_{\alpha i}(x_i) &\leftarrow \left(m^0_{\alpha i}(x_i)\right)^{\gamma_i} \left(m^0_{i,\alpha}(x_i)\right)^{\gamma_i - 1} \\
m_{i,\alpha}(x_i) &\leftarrow \left(m^0_{i,\alpha}(x_i)\right)^{\gamma_i} \left(m^0_{\alpha i}(x_i)\right)^{\gamma_i - 1}
\end{aligned}
$$

$\gamma_i = 1 \Rightarrow$ ordinary BP.

Proof outline

- Easy part - zero temperature sum product.

- Hard part - max product.

Easy part

Fixed points of BP correspond to stationary points of the Bethe-Kikuchi free energy (Yedidia, Freeman, Weiss 01, Kabashima Saad 98).

$$
\begin{aligned}
F \;=\; & \sum_{i,j} \sum_{x_i,x_j} b_{ij}(x_i,x_j) E_{ij}(x_i,x_j) + \sum_i \sum_{x_i} b_i(x_i) E_i(x_i) \\
& + T\left(\sum_{i,j} \sum_{x_i,x_j} b_{ij}(x_i,x_j) \ln b_{ij}(x_i,x_j) - \sum_i c_i \sum_{x_i} b_i(x_i) \ln b_i(x_i) \right)
\end{aligned}
$$

BP and LP

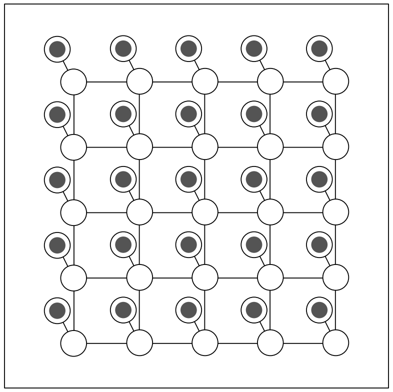
minimize:

$$
\begin{aligned}
F_\beta &= \sum_{i,j} \sum_{x_i,x_j} b_{ij}(x_i,x_j) E_{ij}(x_i,x_j) + \sum_i \sum_{x_i} b_i(x_i) E_i(x_i) \\
&\quad + T \left(\sum_{i,j} \sum_{x_i,x_j} b_{ij}(x_i,x_j) \ln b_{ij}(x_i,x_j) - \sum_i c_i \sum_{x_i} b_i(x_i) \ln b_i(x_i) \right)
\end{aligned}
$$

subject to:

$$
\begin{aligned}
b_{ij}(x_i,x_j) &\in [0,1] \\
\sum_{x_i,x_j} b_{ij}(x_i,x_j) &= 1 \\
\sum_{x_i} b_{ij}(x_i,x_j) &= b_j(x_j)
\end{aligned}
$$

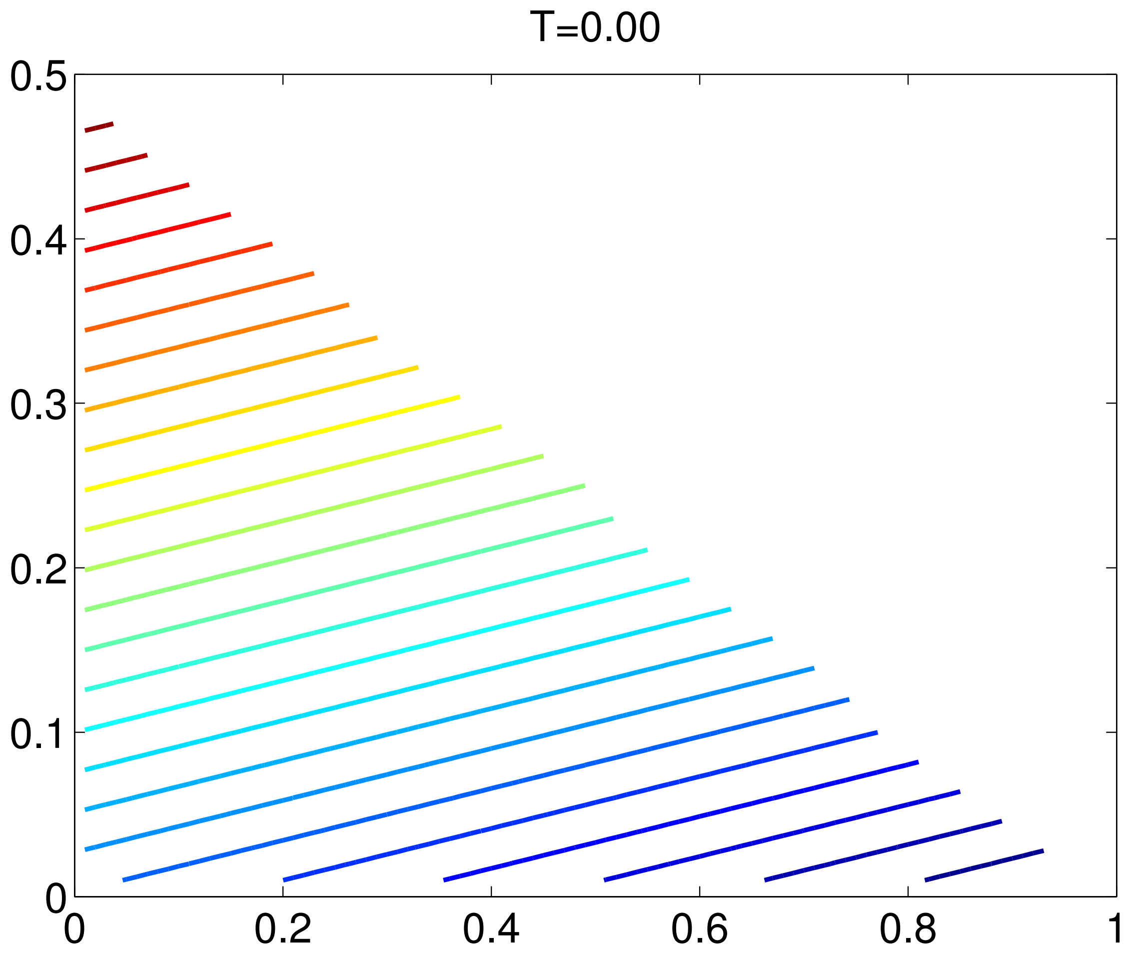
Visualizing BP and LP



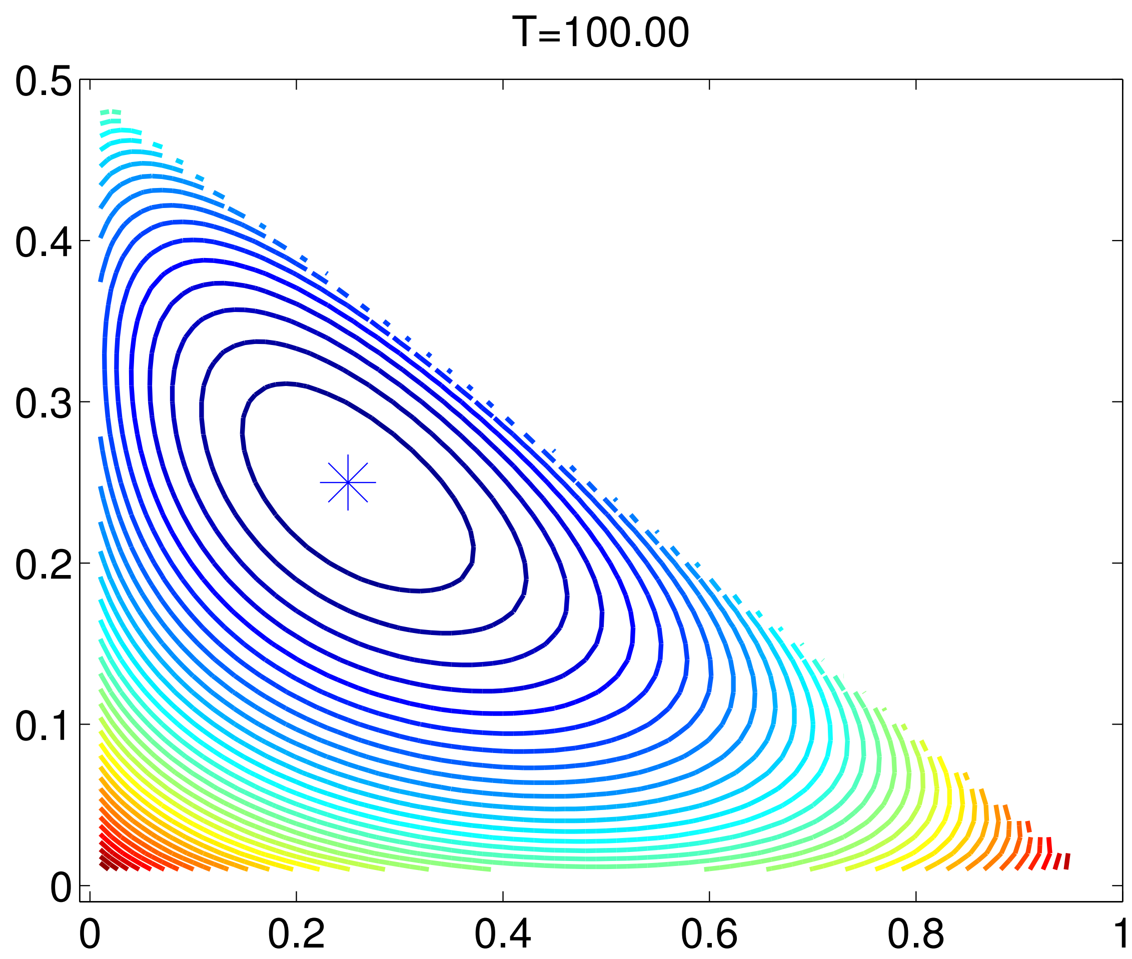
$$
\begin{aligned}
F_\beta \;=\; & \sum_{i,j}\sum_{x_i,x_j} b_{ij}(x_i,x_j)E_{ij}(x_i,x_j) + \sum_i \sum_{x_i} b_i(x_i)E_i(x_i) \\
& +T\left(\sum_{i,j}\sum_{x_i,x_j} b_{ij}(x_i,x_j)\ln b_{ij}(x_i,x_j) - \sum_i (d_i-1)\sum_{x_i} b_i(x_i)\ln b_i(x_i)\right)
\end{aligned}
$$

$$
b_{ij}(x_i,x_j) = \begin{pmatrix} a & b \\ b & 1-(a+2b) \end{pmatrix}
$$

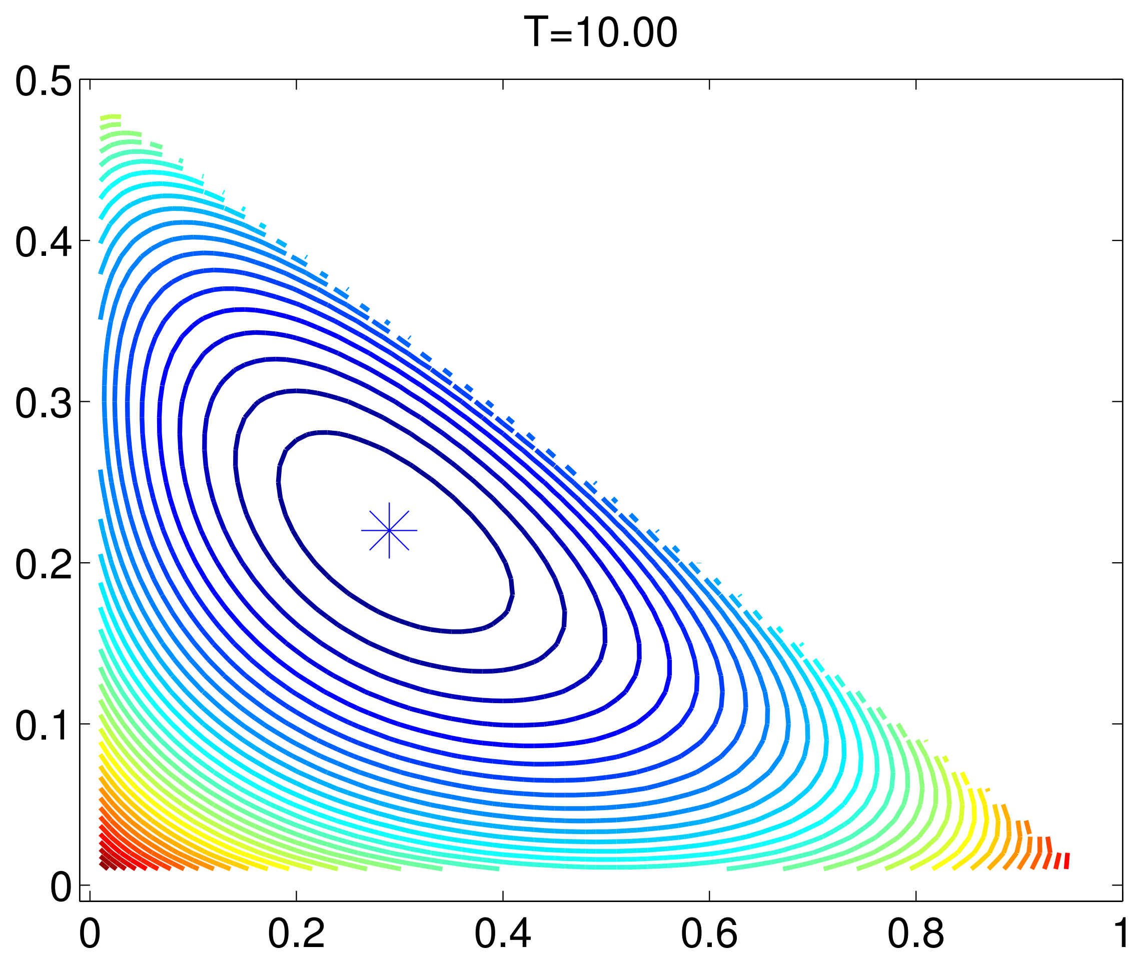
Visualizing BP and LP



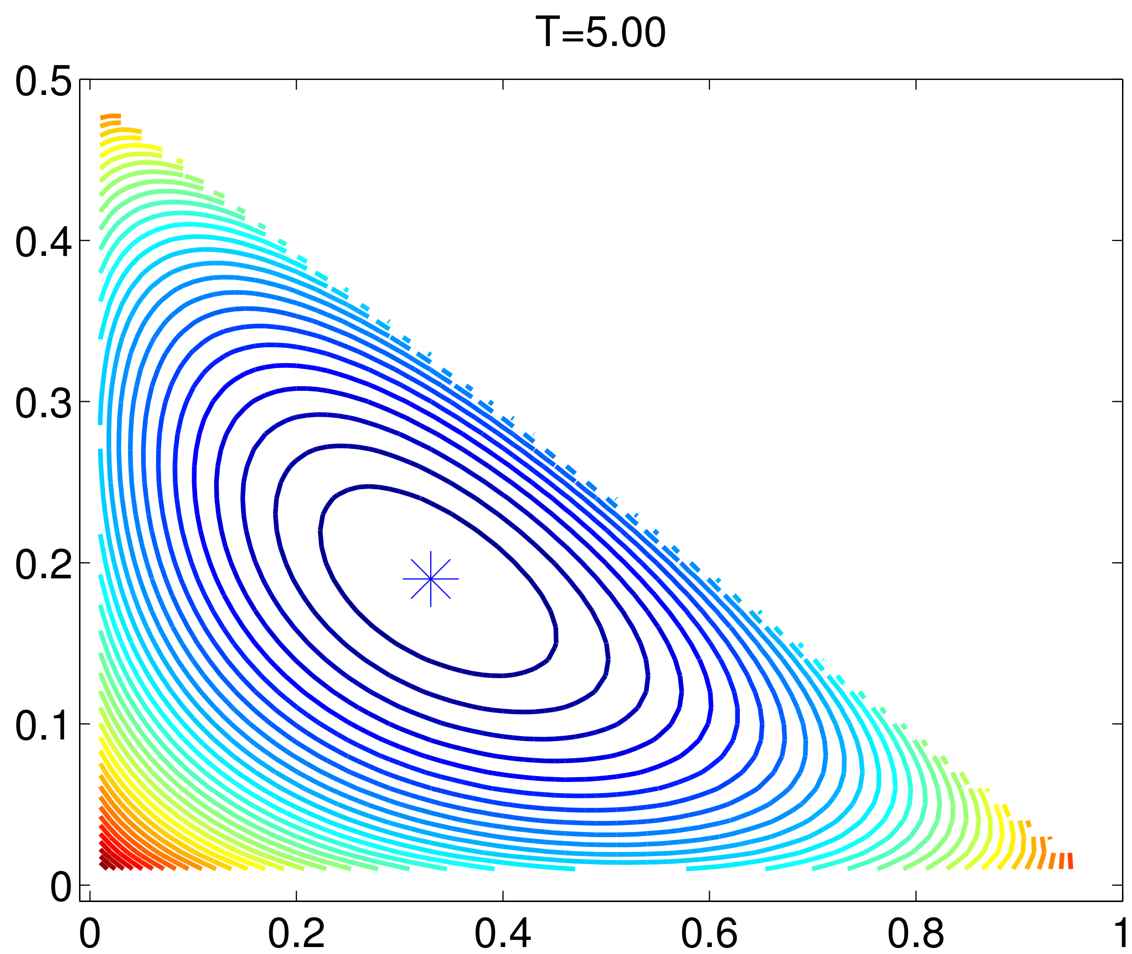
Visualizing BP and LP



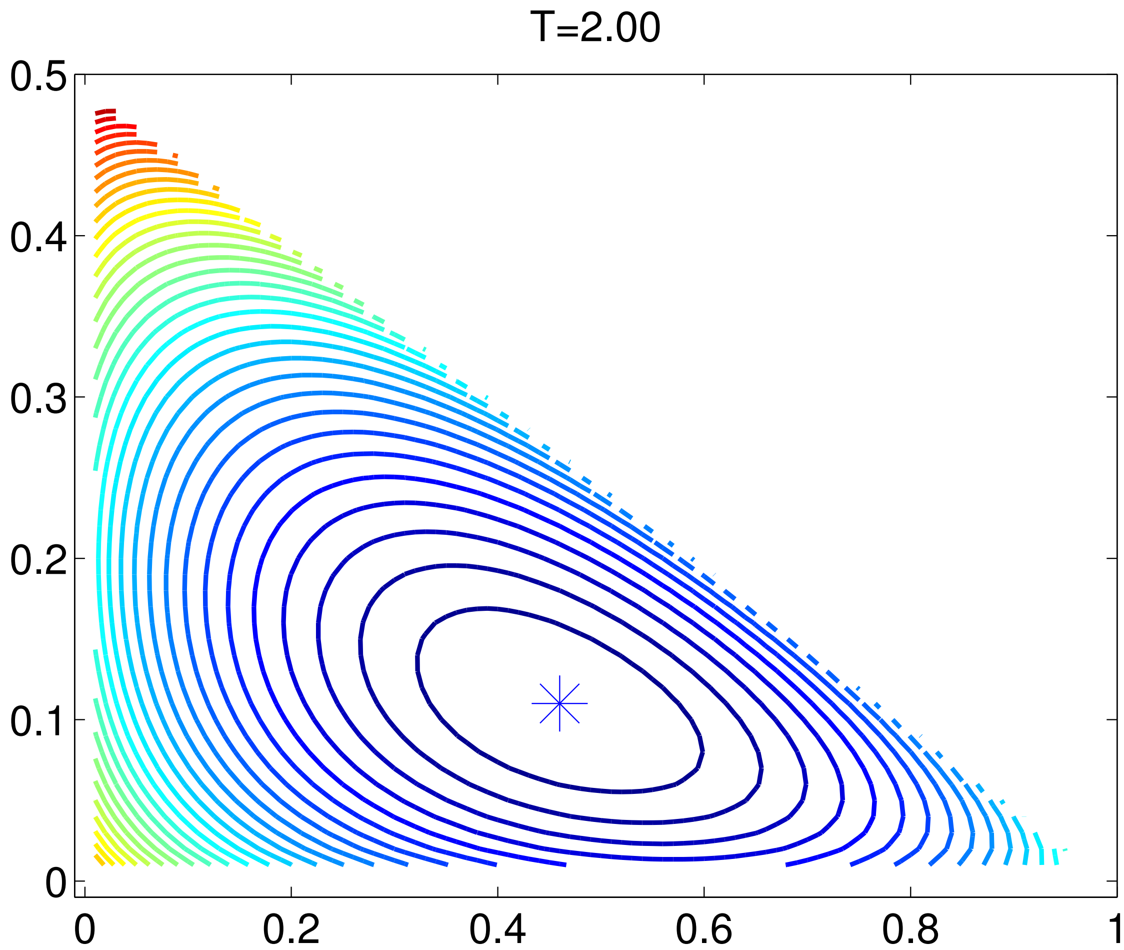
Visualizing BP and LP



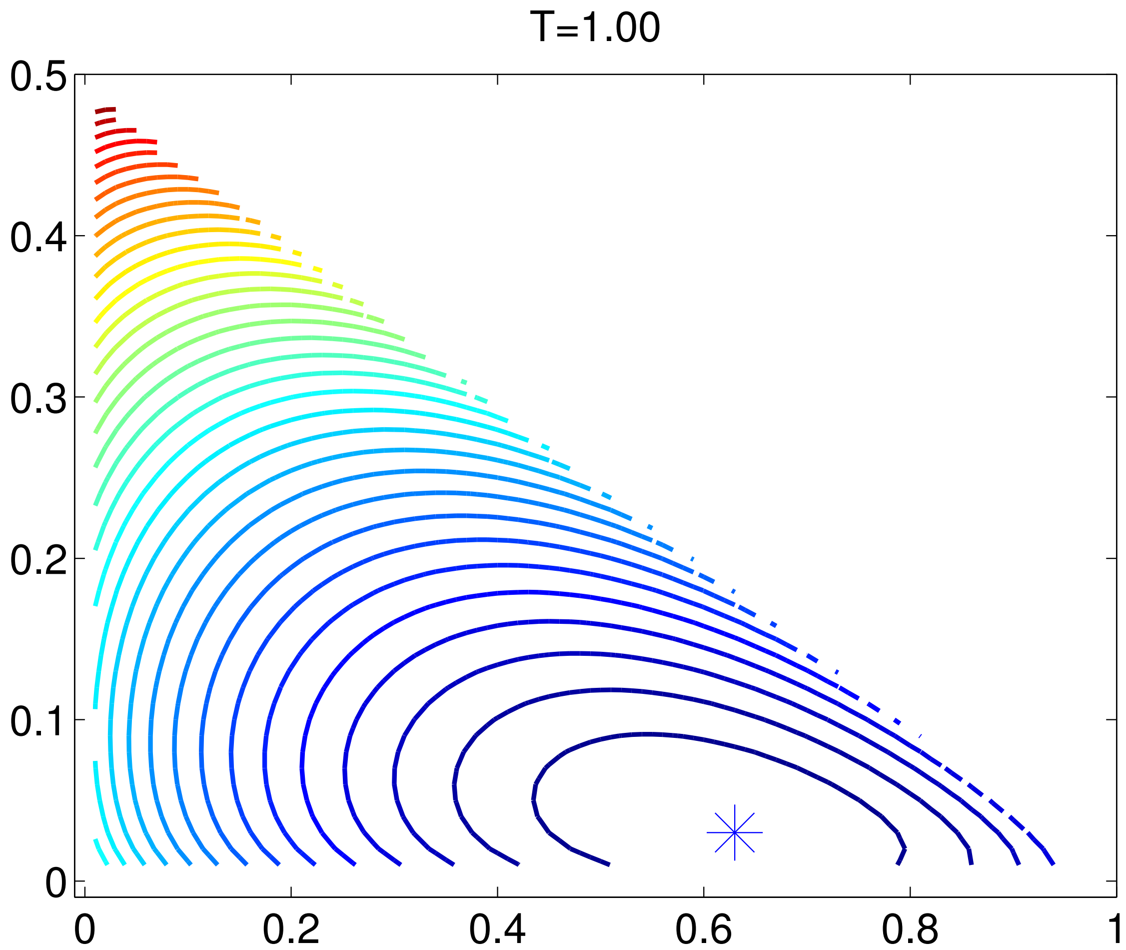
Visualizing BP and LP



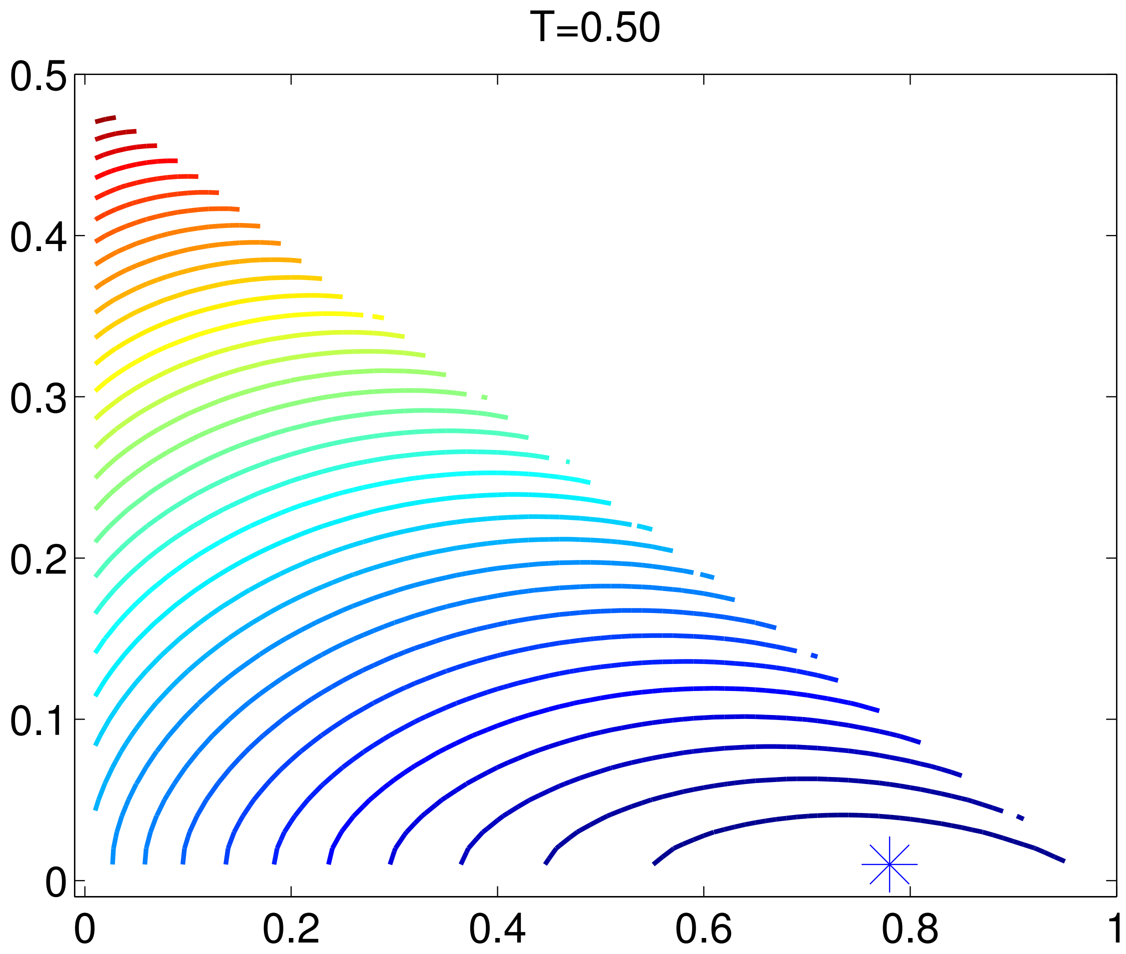
Visualizing BP and LP



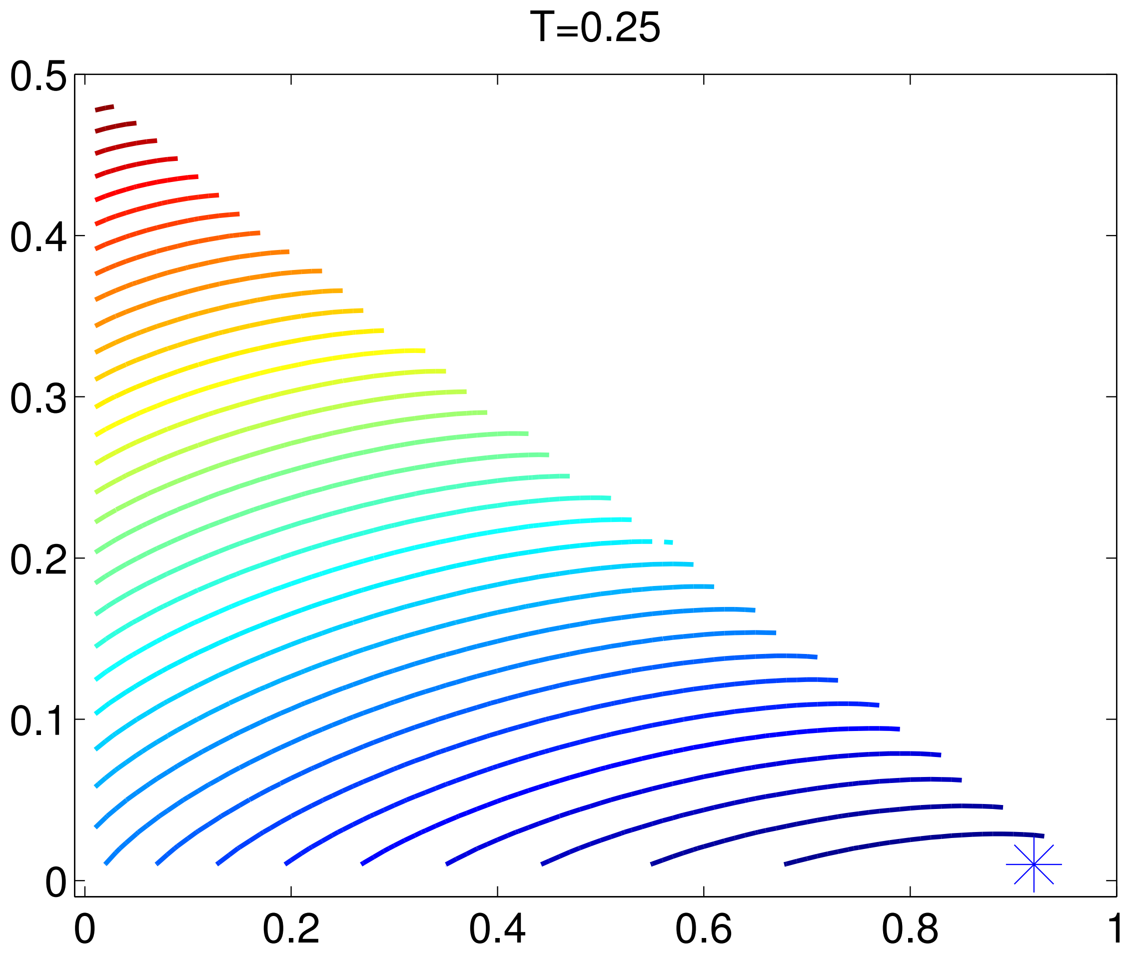
Visualizing BP and LP



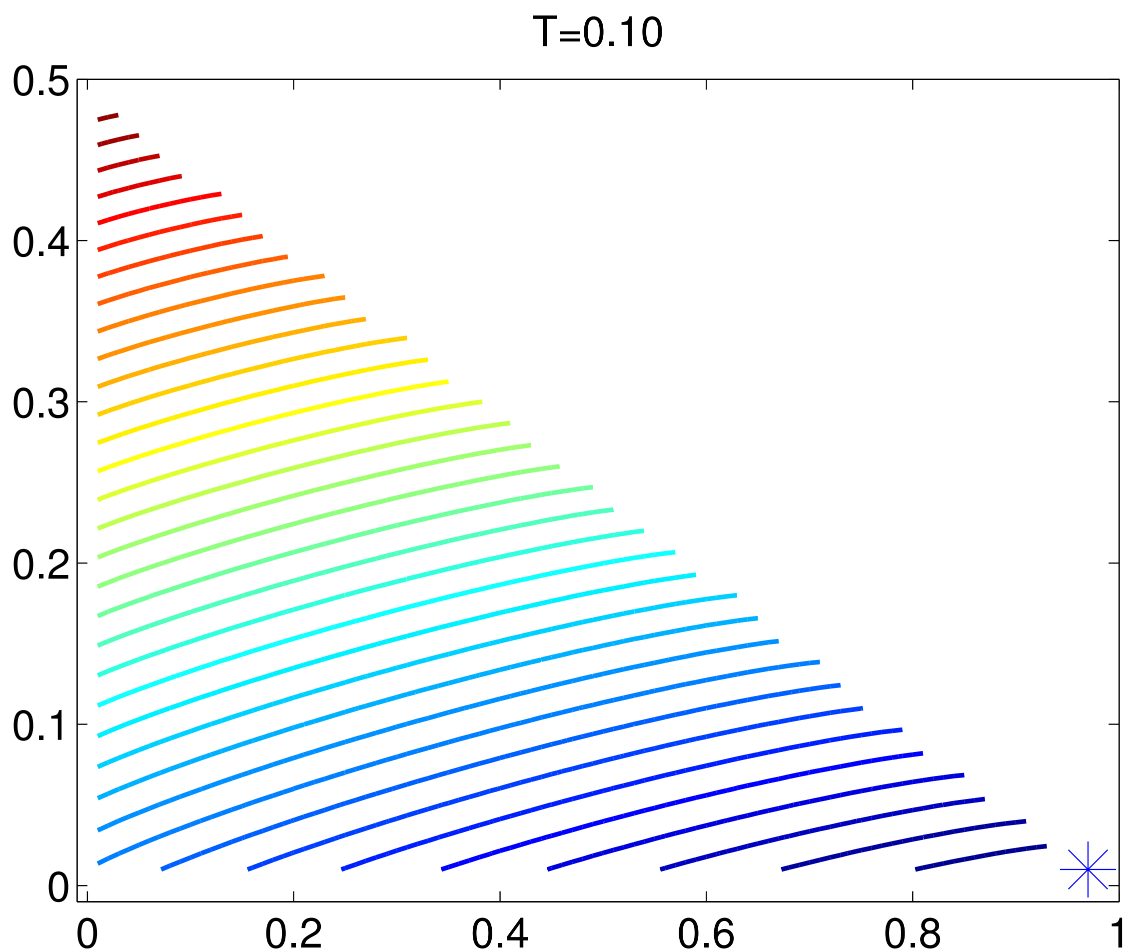
Visualizing BP and LP



Visualizing BP and LP



Visualizing BP and LP



Hard part:

- Intuition - zero temperature sum product = max-product.
- This is wrong (Kolmogorov 03). But zero temperature sum product decoding = max-product decoding.

Theoretical implications:

If BP fixed point unique then BP decoding=LP decoding.

- No ties $\Rightarrow$ BP decoding=LP decoding=MAP decoding.
- Integrality gap $\Rightarrow$ BP decoding=LP decoding $\neq$ MAP decoding.

When is BP fixed point unique?

- Ordinary BP on trees and single cycles.
- Entropy is convex combination of tree entropies (Wainwright et al. 03)
- Entropy is convex (Meltzer et al.05)
- Generalized Dobrushin conditions (Tatikonda and Jordan 02, Heskes 05, Weitz 06)

Practical implications

If you have BP code, add two lines and you have LP code.

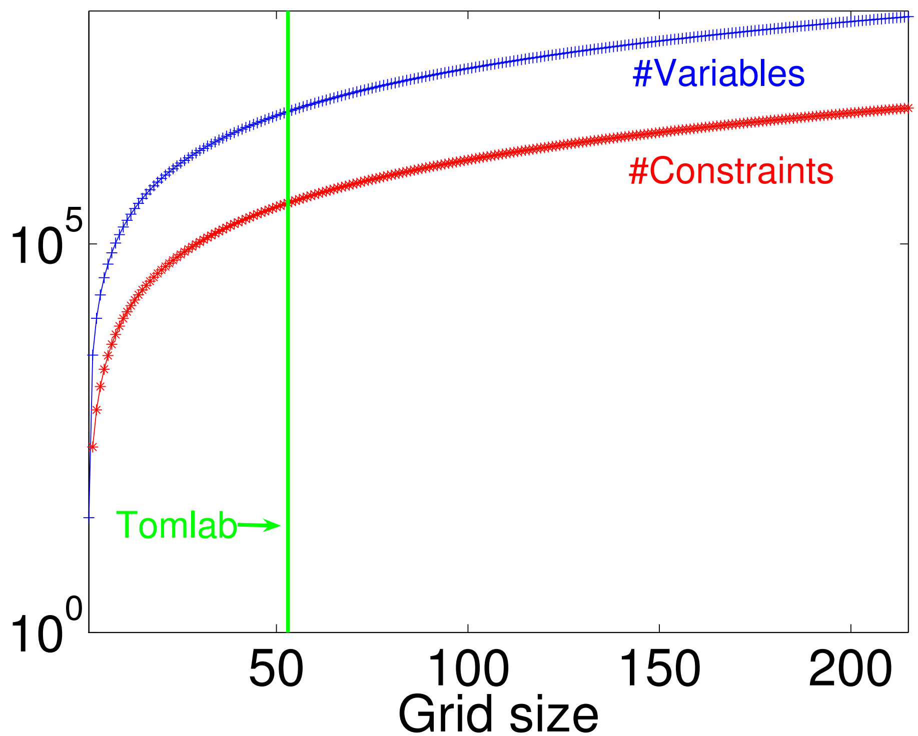
- convergence?
- convergence rate ?

Comparisons

Compare run-times of:

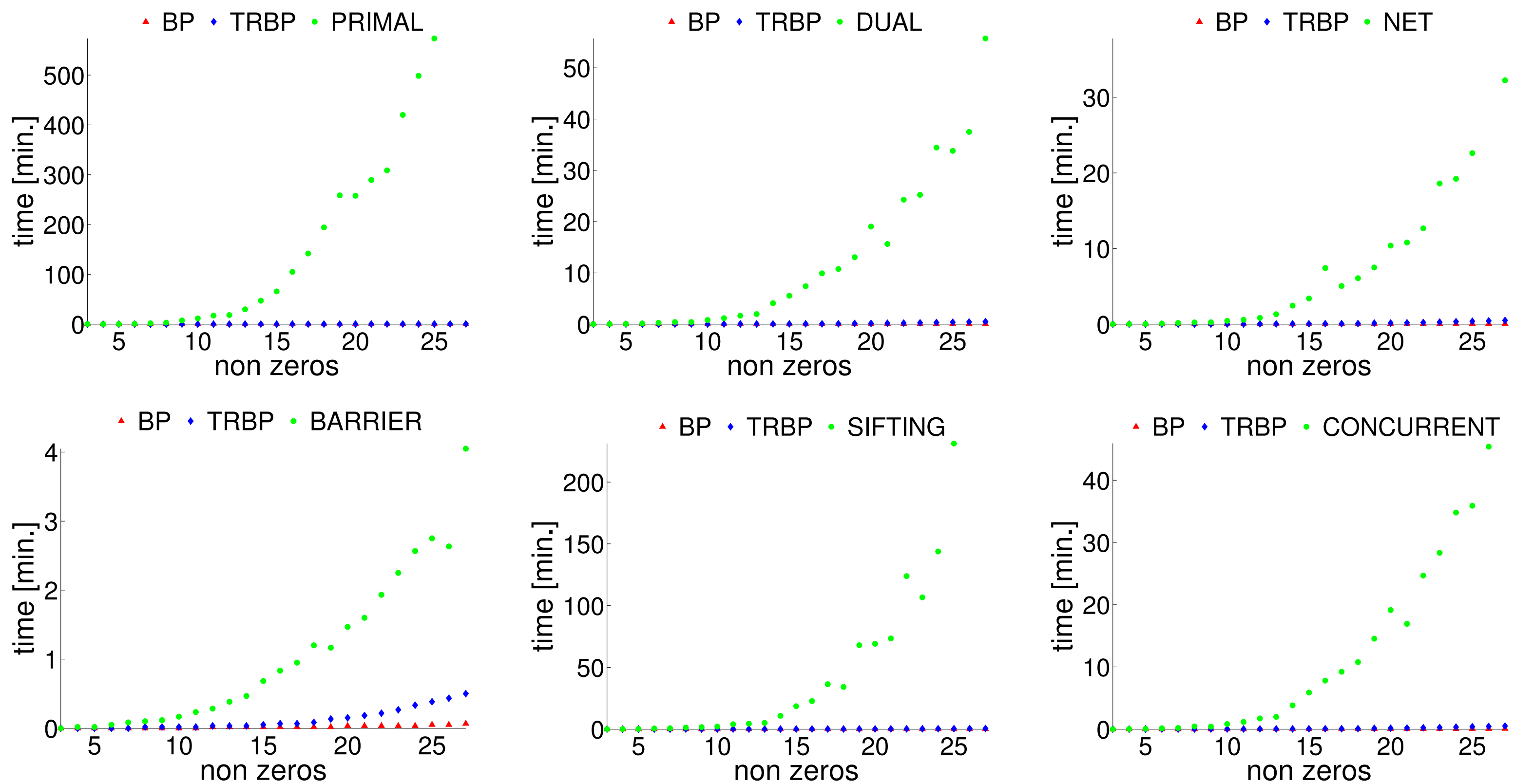
- Generic BP code.
- CPLEX 9.0. Perhaps the most powerful commercial LP solver.

Comparisons - stereo



BP can solve full sized images.

Run times - stereo



Global Optimum

$$
\begin{aligned}
x_{BP} &= (0, 1, ?, 0, 1, ?) \\
x_{LP} &= (0, 1, ?, 0, 1, ?)
\end{aligned}
$$

In any reasonably difficlt problem the LP/BP decoding will have fractional values. But often, we can prove that the non-fractional values are *correct*.

Correctness conditions

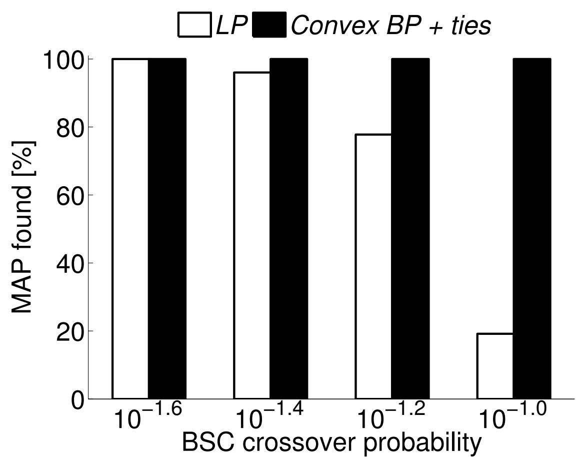
$x_{BP} = (0, 1, ?, 0, 1, ?)$

- When fractional subgraph is a tree (Meltzer et al. 05)
- When beliefs on boundary of fractional subgraph are uniform (Wainwright and Kolmogorov 05, Meltzer et al. 05)
- When MAP of fractional subgraph does not contradict BP beliefs (Meltzer et al. 05, Weiss et al. 07)

NP hardness

In about 90% of benchmark problems, we can find global optimum in about 10 minutes.

MAP decoding of $n = 204$ LDPC



Does global optimum improve performance ?

Using *global optimizers:*

- Stereo vision - unsatisfactory results.
- Side chain prediction - approx. $85 + \epsilon\%$

Better energy functions are needed !

Conclusions

- Energy minimization in computer vision and computional biology. Typically NP hard.

- Standard LP solvers cannot handle LPs arising from our applications.

- BP decoding = LP decoding for large class of problems.

- Convex BP = simple LP solver.

- Globally optimal solutions can be obtained in minutes.