**Problem Set 3**

**Problem 1: The Approximate MAP Problem Revisited**

1. Suppose you run max-product to convergence. Let represent the converged messages and represent the converged beliefs. Show that

Based on the formula above, we can get the result of all the messages, in order to compute the messages, we should initialize it firstly. For instance, we can initialize all the

to be 1. Then iterate for some times until convergence, we can get the max value for each , so we can compute the beliefs by the convergent max message values.

The formula for computing the belief is:

Also, we need to normalize these values for avoiding the overflow, the normalization constant:

We also need to compute the between two nodes, the formula is:

We also need to normalize the , for ,the normalization constant is:

We can see that will be the same as the if node j included, so =

So we can conclude that

2. Suppose you run max-product to convergence. Let represent the converged messages and represent the converged beliefs. Show that .

Based on the formula above, we can get the result of all the messages, in order to compute the messages, we should initialize it firstly. For instance, we can initialize all the

to be 1. Then iterate for some times until convergence, we can get the max value for each , so we can compute the beliefs by the convergent max message values.

The formula for computing the belief is:

]/[]

=[]/[] (1)

And =

Take the into , we can get the formula below:

/

[]

This formula does not include index j.

Thus is proportioned to the normalization factor:

[]/

[] (2)

Also, we need to normalize these values for avoiding the overflow, the normalization constant:

We also need to compute the between two nodes, the formula is:

=[]/

[] (3)

Based on the equation (1), (2), (3), we can get:

3. We can obtain approximate marginals for the approximate MAP problem over the local

marginal polytope by using sum-product.

If the graph is a tree, the joint probability can be factorized as below:

For the loopy BP process, if the BP converges, we can get the value about and , and also we can get the approximated partition function Z.

So, the can be expressed by and :

We can take this formula into the .

Then we can compute the limit for this function and get use the function:

Limit()

**Problem 2: Gibbs Sampling**

Use the graph like this:



The adjacent matrix is:

A = [0 1 1 1;

1 0 0 1;

1 0 0 1;

1 1 1 0];

The weight vector is:

w = [1 2 3 4];

we can get the test result of the matlab code:

%-----------the test report----------%

1 3 3 4

0.2656 0.3281 0.1406 0.2656

0.0781 0.0625 0.3750 0.4844

0.0938 0.0625 0.4219 0.4219

0.1094 0.3906 0.2813 0.2188

4 2 3 1

0.1709 0.3223 0.3965 0.1104

0.0518 0.0781 0.1621 0.7080

0.0518 0.0752 0.1680 0.7051

0.1885 0.3721 0.3271 0.1123

1 4 4 2

0.1744 0.3194 0.3240 0.1822

0.0609 0.1084 0.2460 0.5847

0.0585 0.1091 0.2453 0.5871

0.1789 0.3325 0.3191 0.1695

3 1 1 2

0.1741 0.3272 0.3243 0.1744

0.0627 0.1114 0.2387 0.5872

0.0639 0.1112 0.2392 0.5857

0.1733 0.3237 0.3266 0.1764

1 4 4 3

0.0625 0.1875 0.4063 0.3438

0.0781 0 0.3750 0.5469

0.0313 0.0313 0.4063 0.5313

0.1719 0.6563 0.1250 0.0469

2 3 4 1

0.1914 0.3076 0.2939 0.2070

0.0879 0.1504 0.2539 0.5078

0.0977 0.1602 0.2617 0.4805

0.1396 0.2861 0.3408 0.2334

1 4 4 2

0 0.0156 0 0.9844

0.1563 0.1875 0.6406 0.0156

0.1250 0.2500 0.6094 0.0156

0.2813 0.4844 0.2344 0

4 3 2 1

0.1602 0.2998 0.4424 0.0977

0.0498 0.0957 0.1309 0.7236

0.0596 0.0967 0.1348 0.7090

0.1855 0.3848 0.3193 0.1104

3 1 2 4

0.1705 0.3087 0.3237 0.1971

0.0626 0.1217 0.2349 0.5808

0.0645 0.1209 0.2329 0.5817

0.1794 0.3253 0.3349 0.1604

mbi =

0.2969 0.2568 0.1658 0.1737

0.2656 0.1104 0.1822 0.1744

0.3438 0.2070 0.1732 0.1722

0.9844 0.0977 0.1971 0.1779

>> w

**So we can conclude that**: after iteration for some times, the result of the Gibbs sampling does not have relationship with the initial vector.

**Problem 3: Maximum Likelihood**

And =0, , plug into the formula above, we can get:

is the normalization constant, so we can get the formula of the :

=\*\*+ ………x=[1, 1, 1]

=\*\*+ ………x=[1, 1, -1]

=\*\*+ ………x=[1, -1, 1]

=\*\*+ ……x=[1, -1, -1]

=\*\*+ ……x=[-1, 1, 1]

=\*\*+ ……x=[-1, 1, -1]

=\*\*+ ……x=[-1, -1, 1]

=\*\*+ x=[-1,- 1, -1]

The L(J) is:

------------------------x=[-1, -1, 1]

------------------------x=[1, -1, -1]

--------------------------x=[1, 1, 1]

- ----------------------x=[-1, -1, -1]

- -------------------------x=[1, -1, -1]

Then we get the final answer.