A Simpler Max-Product Maximum Weight Matching Algorithm and the Auction Algorithm

Mohsen Bayati
Department of EE
Stanford University
Stanford, CA 94305
Email: bayati@stanford.edu

Devavrat Shah
Departments of EECS & ESD
MIT
Cambridge, MA 02139
Email: devavrat@mit.edu

Mayank Sharma
IBM TJ Watson Research Center
1101 Kitchawan Rd, P. O. Box 218
Yorktown Heights, NY 10598
Email: mxsharma@us.ibm.com

Abstract— **The max-product "belief propagation" algorithm has received a lot of attention recently due to its spectacular success in many application areas such as iterative decoding, computer vision and combinatorial optimization. There is a lot of ongoing work investigating the theoretical properties of the algorithm. In our previous work (2005) we showed that the max-product algorithm can be used to solve the problem of finding the Maximum Weight Matching (MWM) in a weighted complete bipartite graph. However, for a graph with n nodes the max-product algorithm requires $O(n^4)$ operations to find the MWM compared to $O(n^3)$ for best known algorithms such as those proposed by Edmonds and Karp (1972) and Bertsekas (1988).**

In this paper, we simplify the max-product algorithm to reduce the number of operations required to $O(n^3)$. The simplified algorithm has very similar dynamics to the well-known auction algorithm of Bertsekas (1988). To make this connection precise, we show that the max-product and auction algorithms, when slightly modified, are equivalent. We study the correctness of this modified algorithm. There is a tantalizing similarity between this connection and a recently observed connection between the max-product and LP-based algorithms for iterative decoding by Vontobel and Koetter.

I. INTRODUCTION

Finding the MWM in a bipartite graph is an important problem in many fields e.g. combinatorial optimization and networks (see [4], [5] for references and details). In this section, we first define the problem of finding the MWM in a bipartite graph, then describe the min-sum version of the max-product algorithm and finally present previously known results.

A. MAXIMUM WEIGHT MATCHING

Consider an undirected weighted complete bipartite graph $K_{n,n} = (V_1, V_2, E)$, where $V_1 = \{\alpha_1, \ldots, \alpha_n\}$, $V_2 = \{\beta_1, \ldots, \beta_n\}$ and $(\alpha_i, \beta_j) \in E$ for $1 \le i, j \le n$. Let each edge (α_i, β_j) have weight $w_{ij} \in \mathbb{R}$. If $\pi = \{\pi(1), \ldots, \pi(n)\}$ is a permutation of $\{1, \ldots, n\}$ then the set of n edges $\{(\alpha_1, \beta_{\pi(1)}), \ldots, (\alpha_n, \beta_{\pi(n)})\}$ is called a *matching* of $K_{n,n}$. We denote both the permutation and the corresponding matching by π. The weight of the matching π, denoted by W_π, is defined as

$$W_\pi = \sum_{1 \le i \le n} w_{i\pi(i)}.$$

Then, the maximum weight matching π^* is a matching such that $\pi^* = \text{argmax}_\pi\ W_\pi$.

Next, we transform the problem of finding the MWM in a bipartite graph into one of finding a MAP assignment in a related graphical model (GM). Consider the following GM defined on $K_{n,n}$: let $X_1, \ldots, X_n, Y_1, \ldots, Y_n$ be random variables corresponding to the vertices of $K_{n,n}$ and taking values from $\{1, 2, \ldots, n\}$. Let their joint probability distribution, $p\left(\overline{X} = (x_1, \ldots, x_n); \overline{Y} = (y_1, \ldots, y_n)\right)$, be of the form:

$$p\left(\overline{X}, \overline{Y}\right) = \frac{1}{Z} \prod_{i,j} \psi_{\alpha_i\beta_j}(x_i, y_j) \prod_i \phi_{\alpha_i}(x_i)\phi_{\beta_i}(y_i), \tag{1}$$

where the pairwise compatibility functions, $\psi_{\cdot\cdot}(\cdot,\cdot)$, are defined as

$$\psi_{\alpha_i\beta_j}(r, s) = \begin{cases} 0 & r = j \text{ and } s \neq i \\ 0 & r \neq j \text{ and } s = i \\ 1 & \text{Otherwise} \end{cases}$$

the potentials at the nodes, $\phi_\cdot(\cdot)$, are defined as

$$\phi_{\alpha_i}(r) = e^{w_{ir}}, \quad \phi_{\beta_j}(r) = e^{w_{rj}}, \quad \forall\ 1 \le\ i, j, r, s\ \le\ n,$$

and Z is the normalization constant. The following claims are a direct consequence of these definitions (see [4] for their proofs).

Claim 1: For the GM as defined above, the joint density $p\left(\overline{X} = (x_1, \ldots, x_n), \overline{Y} = (y_1, \ldots, y_n)\right)$ is nonzero if and only if $\pi_\alpha(\overline{X}) = \{(\alpha_1, \beta_{x_1}), (\alpha_2, \beta_{x_2}), \ldots, (\alpha_n, \beta_{x_n})\}$ and $\pi_\beta(\overline{Y}) = \{(\alpha_{y_1}, \beta_1), (\alpha_{y_2}, \beta_2), \ldots, (\alpha_{y_n}, \beta_n)\}$ are both matchings and $\pi_\alpha(\overline{X}) = \pi_\beta(\overline{Y})$. Further, when nonzero, $p\left(\overline{X}, \overline{Y}\right)$ is equal to $\frac{1}{Z} e^{2\sum_i w_{ix_i}}$.

Claim 2: Let $(\overline{X}^*, \overline{Y}^*)$ be such that

$$(\overline{X}^*, \overline{Y}^*) = \arg\max\{p\left(\overline{X}, \overline{Y}\right)\}.$$

Then, the corresponding $\pi_\alpha(\overline{X}^*) = \pi_\beta(\overline{Y}^*)$ is the MWM.

B. MIN-SUM ALGORITHM FOR $K_{n,n}$

We present the min-sum version of the max-product algorithm for finding the MWM (see [4] for the equivalence between the standard max-product and min-sum algorithms).

1-4244-0504-1/06/$20.00 ©2006 IEEE

Define the compatibility matrix $\Psi_{\alpha_i\beta_j} = [\psi_{\alpha_i\beta_j}(r,s)] \in \mathbb{R}^{n\times n}$. Also, let

$$\Phi_{\alpha_i} = [\phi_{\alpha_i}(1), \ldots, \phi_{\alpha_i}(n)]^t, \quad \Phi_{\beta_j} = [\phi_{\beta_j}(1), \ldots, \phi_{\beta_j}(n)]^t.$$

Min-Sum Algorithm

(1) Let $M^k_{\alpha_i\to\beta_j} = [m^k_{\alpha_i\to\beta_j}(1), \ldots, m^k_{\alpha_i\to\beta_j}(n)]^t \in \mathbb{R}^{n\times 1}$ denote the messages passed from α_i to β_j in the iteration $k \geq 0$, for $1 \leq i, j \leq n$. Similarly, $M^k_{\beta_j\to\alpha_i}$ is the message vector passed from β_j to α_i in the iteration k.

(2) Initially $k = 0$ and set the messages as follows. Let

$$M^0_{\alpha_i\to\beta_j} = [m^0_{\alpha_i\to\beta_j}(1) \ldots m^0_{\alpha_i\to\beta_j}(n)]^t$$
$$M^0_{\beta_j\to\alpha_i} = [m^0_{\beta_j\to\alpha_i}(1) \ldots m^0_{\beta_j\to\alpha_i}(n)]^t$$

where

$$m^0_{\alpha_i\to\beta_j}(r) = \begin{cases} w_{ij} & \text{if } r = i \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

$$m^0_{\beta_i\to\alpha_j}(r) = \begin{cases} w_{ji} & \text{if } r = i \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

(3) For $k \geq 1$, messages in iteration k are obtained from messages of iteration $k-1$ recursively as follows: for $1 \leq i, j, r \leq n$

$$\begin{aligned} m^k_{\alpha_i\to\beta_j}(r) &= \max_{1\leq q\leq n} \psi_{\alpha_i\beta_j}(q,r) \left[w_{iq} + \sum_{\ell\neq j} m^{k-1}_{\beta_\ell\to\alpha_i}(q) \right], \\ m^k_{\beta_j\to\alpha_i}(r) &= \max_{1\leq q\leq n} \psi_{\alpha_i\beta_j}(r,q) \left[w_{qj} + \sum_{\ell\neq j} m^{k-1}_{\beta_\ell\to\alpha_i}(q) \right] \end{aligned} \tag{4}$$

(4) In iteration k, for $1 \leq i, j, r \leq n$, let

$$\begin{aligned} b^k_{\alpha_i}(r) &= \left[\sum_\ell m^k_{\beta_\ell\to\alpha_i}(r) \right] + \phi_{\alpha_i}(r), \\ b^k_{\beta_j}(r) &= \left[\sum_\ell m^k_{\alpha_\ell\to\beta_i}(r) \right] + \phi_{\beta_i}(r). \end{aligned} \tag{5}$$

(5) The estimated MWM at the end of iteration k is π^k, where $\pi^k(i) = \arg\max_{1\leq j\leq n}\{b^k_{\alpha_i}(j)\}$, for $1 \leq i \leq n$.

(6) Repeat (3)-(5) till π^k converges.

C. PREVIOUS WORK

In [4], the authors showed that the min-sum algorithm converges to the MWM whenever its unique. Define $w^* = \max_{i,j}|w_{ij}|$ and $\epsilon = W_{\pi^*} - \max_{\pi\neq\pi^*}(W_\pi)$. Then, they proved

Theorem 1: For any weighted complete bipartite graph $K_{n,n}$ with unique maximum weight matching, the min-sum algorithm converges to the MWM within $\lceil \frac{2nw^*}{\epsilon} \rceil$ iterations.

In each iteration of the min-sum algorithm every node sends n messages (numbers) to each of the n nodes in the other partition, and performs $O(n)$ operations per message computation. Thus, in each iteration a total of $O(n^3)$ messages are exchanged and $O(n^3)$ operations are performed. Hence by Theorem 1, the algorithm requires a total of $O\left(\frac{n^4 w^*}{\epsilon}\right)$ operations to find the MWM. That is, the algorithm requires $O(n^4)$ operations for fixed w^* and ϵ. However, known algorithms such as Edmond-Karp's algorithm [1] or the Auction algorithm [2] have a complexity of $O(n^3)$.

D. ORGANIZATION AND OUR RESULTS

In this paper, we present two results. In Section II, we simplify the min-sum algorithm so that the total computational cost and number of message exchanges of the resulting algorithm are of order $O(n^3)$ for fixed w^* and ϵ. Second, we relate this simplified min-sum algorithm to the auction algorithm. Specifically, we present slight modifications of the min-sum and auction algorithms which we show to be equivalent and then we establish the correctness property of this modified algorithm. This is presented in Section III. Finally, we present conclusions and directions for future research in Section IV.

II. SIMPLIFIED MIN-SUM ALGORITHM FOR $K_{n,n}$

We describe the simplified algorithm, prove its equivalence to the min-sum algorithm and analyze its computational complexity. Unlike the min-sum algorithm where messages are vectors, each α_i sends a number to β_j and vice-versa.

Simplified Min-Sum (SMS) Algorithm

(1) Let the message from α_i to β_j in iteration k be denoted as $\hat{m}^k_{\alpha_i\to\beta_j} \in \mathbb{R}$. Similarly, the messages from β_j to α_i in iteration k be denoted as $\hat{m}^k_{\beta_j\to\alpha_i} \in \mathbb{R}$.

(2) Initialize $k = 0$ and set the messages as follows:

$$\hat{m}^0_{\alpha_i\to\beta_j} = w_{ij}; \quad \hat{m}^0_{\beta_j\to\alpha_i} = w_{ij}$$

(3) For $k \geq 1$, iterate as follows:

$$\begin{aligned} \hat{m}^k_{\alpha_i\to\beta_j} &= w_{ij} - \max_{\ell\neq j} \hat{m}^{k-1}_{\beta_\ell\to\alpha_i}, \\ \hat{m}^k_{\beta_j\to\alpha_i} &= w_{ij} - \max_{\ell\neq i} \hat{m}^{k-1}_{\alpha_\ell\to\beta_j}. \end{aligned} \tag{6}$$

(4) The estimated MWM at the end of iteration k is π^k, where $\pi^k(i) = \arg\max_{1\leq j\leq n}\{\hat{m}^k_{\beta_j\to\alpha_i}\}$, for $1 \leq i \leq n$.

(5) Repeat (3)-(4) till π^k converges.

Next, we establish equivalence between the two algorithms.

Lemma 1: Modify the min-sum algorithm by adding constants A^k_{ij} and B^k_{ij} to messages $m^k_{\alpha_i\to\beta_j}(r)$ and $m^k_{\beta_j\to\alpha_i}(r)$ respectively for all $1 \leq r \leq n$ and $k \geq 0$. Then, the modified algorithm estimates the same matching π^k as the original min-sum algorithm for all k.

Proof: Let $\bar{m}^k_{\alpha_i\to\beta_j}(r), \bar{m}^k_{\beta_j\to\alpha_i}(r)$ be the messages of the modified algorithm for any $1 \leq i, j, r \leq n, k \geq 0$. Then the proof of the Lemma follows by (5) and establishing the following fact inductively: $\forall\ i, j$ and k,

$$\bar{m}^k_{\alpha_i\to\beta_j}(r) = m^k_{\alpha_i\to\beta_j}(r) + C^k_{ij},$$

$$\bar{m}^k_{\beta_j\to\alpha_i}(r) = m^k_{\beta_j\to\alpha_i}(r) + D^k_{ij},$$

and for all r, where C^k_{ij}, D^k_{ij} depend only on $(A^\ell_{xy}, B^\ell_{x,y}), 0 \leq \ell \leq k, 1 \leq x, y \leq n$. We omit the details of the proof of the above fact due to space constraints. ■

Lemma 2: For $k \geq 1$, the estimated matching π^k in both the min-sum and simplified min-sum algorithms are identical.

Proof: Consider the min-sum algorithm. For any $k \geq 0$, we claim that for all $1 \leq i, j \leq n$, $m^k_{\alpha_i \to \beta_j}(r), r \neq i$ are the same, i.e. for any $r_1, r_2 \neq i$, $m^k_{\alpha_i \to \beta_j}(r_1) = m^k_{\alpha_i \to \beta_j}(r_2)$. For $k = 0$, this claim holds by definition. For $k \geq 1$, consider the definition of $m^k_{\alpha_i \to \beta_j}(r), r \neq i$.

$$\begin{aligned} m^k_{\alpha_i \to \beta_j}(r) &= \max_{1 \leq q \leq n} \psi_{\alpha_i \beta_j}(q, r) \left[w_{iq} + \sum_{\ell \neq j} m^{k-1}_{\beta_\ell \alpha_i}(q) \right] \\ &= \max_{q \neq j} \left[w_{iq} + \sum_{\ell \neq j} m^{k-1}_{\beta_\ell \alpha_i}(q) \right], \qquad (7) \end{aligned}$$

where (7) follows from property of $\psi_{\alpha_i \beta_j}(\cdot, \cdot)$. Since (7) is independent of $r(\neq i)$ the desired claim is proved.

Equation (7) implies that $M^k_{\alpha_i \to \beta_j}$ has only two distinct values: $m^k_{\alpha_i \to \beta_j}(i)$ and $m^k_{\alpha_i \to \beta_j}(r)$, $r \neq i$. Now subtract $m^k_{\alpha_i \to \beta_j}(r)$, $r \neq i$ from all coordinates of $M^k_{\alpha_i \to \beta_j}$. Lemma 1 guarantees that the resulting matching π^k for all k does not change. This results in an algorithm which is equivalent to the min-sum algorithm, but where each message vector $M^k_{\alpha_i \to \beta_j}$ has all but the i^{th} coordinate equal to zero. Denote the values of these non-zero coordinates by $\tilde{m}^k_{\alpha_i \to \beta_j}$. From (4), it follows that $\tilde{m}^k_{\alpha_i \to \beta_j}$ satisfies:

$$\begin{aligned} \tilde{m}^k_{\alpha_i \to \beta_j} &= w_{ij} - \max_{\ell \neq j}(\tilde{m}^{k-1}_{\beta_\ell \to \alpha_i} + w_{i\ell}), \\ \tilde{m}^k_{\beta_j \to \alpha_i} &= w_{ij} - \max_{\ell \neq i}(\tilde{m}^{k-1}_{\alpha_\ell \to \beta_j} + w_{\ell j}). \qquad (8) \end{aligned}$$

Similarly, the new beliefs are

$$\tilde{b}^k_{\alpha_i}(r) = \tilde{m}^k_{\beta_r \to \alpha_i} + w_{ir}, \qquad \tilde{b}^k_{\beta_j}(s) = \tilde{m}^k_{\alpha_s \to \beta_j} + w_{sj}. (9)$$

To each side of (8), add w_{ij} and divide by 2. Setting $\hat{m}^k_{\alpha_i \to \beta_j} = \frac{\tilde{m}^k_{\alpha_i \to \beta_j} + w_{ij}}{2}$ gives us (6). Further, $\hat{m}^k_{\alpha_i \to \beta_j}$ as defined satisfies the same initial condition as in the simplified min-sum algorithm. Consequently for all i, j, k, we obtain

$$\hat{m}^k_{\alpha_i \to \beta_j} = \frac{\tilde{m}^k_{\alpha_i \to \beta_j} + w_{ij}}{2} = 2\tilde{b}_{\alpha_i}(j). \qquad (10)$$

Similarly, for all i, j, k we also have

$$\hat{m}^k_{\beta_j \to \alpha_i} = \frac{\tilde{m}^k_{\beta_j \to \alpha_i} + w_{ij}}{2} = 2\tilde{b}_{\beta_j}(i). \qquad (11)$$

Equations (10)$-$(11) and the above discussion prove that the modified min-sum and SMS algorithms produce the same matching π^k for all k, thus proving Lemma 2. ■

A. Complexity of Simplified Min-Sum

Lemma 2 and Theorem 1 imply that the SMS algorithm converges after $O(nw^*/\epsilon)$ iterations. Also, the SMS algorithm exchanges a total of $O(n^2)$ messages per iteration. Next, we will show that each iteration requires $O(n^2)$ simple operations. This will establish that for fixed w^* and ϵ, the algorithm performs $O(n^3)$ operations and message exchanges.

To complete the complexity analysis, we describe an algorithm to compute messages $\hat{m}^k_{\alpha_1 \to \beta_j}, 1 \leq j \leq n$ using received messages $\hat{m}^{k-1}_{\beta_j \to \alpha_1}, 1 \leq j \leq n$ that requires a total of $O(n)$ operations. By symmetry, the same algorithm will be used at $\alpha_i, 1 \leq i \leq n$, and $\beta_j, 1 \leq j \leq n$. Thus, the total number of operations per iteration will be $O(n^2)$. Define

$$\begin{aligned} i_1 &= \text{argmax}_{1 \leq j \leq n} \hat{m}^{k-1}_{\beta_j \to \alpha_1}, \\ i_2 &= \text{argmax}_{1 \leq j \leq n, j \neq i_1} \hat{m}^{k-1}_{\beta_j \to \alpha_1}, \\ \text{Max}_1 = \hat{m}^{k-1}_{\beta_{i_1} \to \alpha_1}, & \qquad \text{Max}_2 = \hat{m}^{k-1}_{\beta_{i_2} \to \alpha_1}. \qquad (12) \end{aligned}$$

Then, from (6) we obtain

$$\begin{aligned} \hat{m}^k_{\alpha_1 \to \beta_{i_1}} &= w_{1 i_1} - \text{Max}_2, \\ \hat{m}^k_{\alpha_1 \to \beta_j} &= w_{1j} - \text{Max}_1, \quad \text{for } j \neq i_1. \qquad (13) \end{aligned}$$

From (12) and (13), it is easy to see that computing all messages $\hat{m}^k_{\alpha_1 \to \beta_j}, 1 \leq j \leq n$ takes $O(n)$ operations. Summarizing the discussion of the above section, we obtain the following.

Theorem 2: The simplified min-sum algorithm finds the MWM in $O\left(\frac{nw^*}{\epsilon}\right)$ iterations with a total of $O\left(\frac{n^3 w^*}{\epsilon}\right)$ operations and message exchanges.

III. AUCTION AND MIN-SUM

In this section, we will first recall the auction algorithm [2] and then describe its relation to the min-sum algorithm.

A. AUCTION ALGORITHM FOR MWM

The Auction algorithm finds the MWM via an "auction": all α_i become buyers and all β_j become objects. Let p_j denote the price of β_j and w_{ij} be the cost of α_i buying β_j. The net benefit of an assignment or matching π is defined as

$$\sum_{i=1}^{n} \left(w_{i\pi(i)} - p_{\pi(i)} \right).$$

The goal is to find π^* that maximizes this net benefit. It is clear that for any set of prices $p_1, \ldots, p_n$, the MWM maximizes the net benefit. The auction algorithm is an iterative method for finding the optimal prices and an assignment that maximizes the net benefit (and is therefore the MWM).

Auction Algorithm.

- Initialize the assignment $S = \emptyset$, the set of unassigned buyers $I = \{\alpha_1, \ldots, \alpha_n\}$, and prices $p_j = 0$ for all j.
- The algorithm runs in two phases, which are repeated until S is a complete matching.
- *Phase 1: Bidding* For all $\alpha_i \in I$,
  (1) Find benefit maximizing β_j. Let,

$$j_i = \text{argmax}_j \{w_{ij} - p_j\}, \; v_i = \max_j \{w_{ij} - p_j\}, \quad (14)$$

$$\text{and } u_i = \max_{j \neq j_i} \{w_{ij} - p_j\}. \qquad (15)$$

(2) Compute the ”bid” of buyer α_i, denoted by $b_{\alpha_i\to\beta_j}$ as follows: given a fixed positive constant δ,

$$b_{\alpha_i\to\beta_{j_i}} = w_{ij_i} - u_i + \delta.$$

- *Phase 2: Assignment.* For each object β_j,

(3) Let $P(j)$ be the set of buyers from which β_j received a bid. If $P(j) \neq \emptyset$, increase p_j to the highest bid,

$$p_j = \max_{\alpha_i \in P(j)} b_{\alpha_i\to\beta_j}.$$

(4) Remove the maximum bidder α_{i_j} from I and add (α_{i_j}, β_j) to S. If $(\alpha_k, \beta_j) \in S,\ k \neq i_j$, then put α_k back in I.

Theorem 3 ([3]): If $0 < \delta < \epsilon/n$, then the assignment S converges to the MWM in $O(nw^*/\epsilon)$ iterations with running time $O(n^3 w^*/\epsilon)$ (where ϵ and w^* are as defined earlier).

B. *CONNECTING MIN-SUM AND AUCTION*

The similarity between equations (12)-(13) and (14)-(15) suggests a connection between the min-sum and auction algorithms. Next, we describe modifications to the min-sum and auction algorithms, called *min-sum auction I* and *min-sum auction II*, respectively. We will show that these versions are equivalent and derive some of their key properties. Here we consider the auction algorithm with $\delta = 0$ and deal with the case $\delta > 0$ in the next section.

Min-Sum Auction I

(1) Each α_i sends a number to β_j and vice-versa. Let the messages in iteration k be denoted as $\tilde{m}^k_{\alpha_i\to\beta_j},\ \tilde{m}^k_{\beta_j\to\alpha_i} \in \mathbb{R}$.

(2) Initialize $k = 0$ and set $\tilde{m}^0_{\beta_j\to\alpha_i} = 0$.

(3) For $k \geq 1$, update messages as follows:

$$\begin{aligned} \tilde{m}^k_{\alpha_i\to\beta_j} &= w_{ij} - \max_{\ell\neq j}\{w_{i\ell} - \tilde{m}^{k-1}_{\beta_\ell\to\alpha_i}\}, \\ \tilde{m}^k_{\beta_j\to\alpha_i} &= \max_{\ell=1}^{n} \tilde{m}^k_{\alpha_\ell\to\beta_j}, \end{aligned} \tag{16}$$

(4) The estimated MWM at the end of iteration k is the set of edges

$$\pi^k = \{(\alpha_{i_j}, \beta_j)|\ i_j = \arg\max_{1\leq\ell\leq n}\{\tilde{m}^k_{\alpha_\ell\to\beta_j}\}\ 1 \leq j \leq n,$$
$$\text{and } \tilde{m}^k_{\alpha_{i_j}\to\beta_j} \geq \tilde{m}^{k-1}_{\beta_j\to\alpha_i}\}$$

(5) Repeat (3)-(4) till π^k is a complete matching.

Min-Sum Auction II.

- Initialize the assignment $S = \emptyset$ and prices $p_j = 0$ for all j.
- The algorithm runs in two phases, which are repeated until S is a complete matching.
- *Phase 1: Bidding.* For all α_i,

(1) Find β_j that maximizes the benefit. Let,

$$j_i = \text{argmax}_j\{w_{ij} - p_j\},\ v_i = \max_j\{w_{ij} - p_j\}, \tag{17}$$

$$\text{and } u_i = \max_{j\neq j_i}\{w_{ij} - p_j\}. \tag{18}$$

(2) Compute the ”bid” of buyer α_i, denoted by $b_{\alpha_i\to\beta_j}$:

$$b_{\alpha_i\to\beta_{j_i}} = w_{ij_i} - u_i, \text{ and } b_{\alpha_i\to\beta_j} = w_{ij} - v_i,\ j \neq j_i.$$

- *Phase 2: Assignment.* For each object β_j,

(3) Set price p_j to the highest bid, $p_j = \max_{\alpha_i} b_{\alpha_i\to\beta_j}$.

(4) Reset $S = \emptyset$. Then, for each j add the pair (α_{i_j}, β_j) to S if $b_{\alpha_{j_i}\to\beta_j} \geq p_j$, where α_{i_j} is a buyer attaining the maximum in step (3).

Theorem 4: The algorithms min-sum auction I and II are equivalent.

Proof: Let $b^k_{\alpha_i\to\beta_j}$ and p^k_j denote the bids and prices at the end of iteration k in algorithm min-sum auction II. Now, identify $b^k_{\alpha_i\to\beta_j}$ with $\tilde{m}^k_{\alpha_i\to\beta_j}$ and p^k_j with $\tilde{m}^k_{\beta_j\to\alpha_i}$. Then it is immediate that min-sum auction II becomes identical to min-sum auction I. This completes the proof of Theorem 4. ■

Next we will prove that if the min-sum auction algorithm terminates (we omit reference to I or II), it finds the correct maximum weight matching. As we will see, the proof uses standard arguments (see [2] for example).

Theorem 5: Let σ be the termination matching of the min-sum auction I (or II). Then it is the MWM, i.e. $\sigma = \pi^*$.

Proof: The proof follows by establishing that at termination, the messages of min-sum auction form the optimal solution for the dual of the MWM problem and σ is the corresponding optimal solution to the primal, i.e. MWM. To do so, we first state the dual of the MWM problem

$$\begin{aligned} \min \quad & \sum_{i=1}^{n} r_i + \sum_{j=1}^{n} p_j \\ \text{subject to} \quad & r_i + p_j \ \geq\ w_{ij}. \end{aligned} \tag{19}$$

Let (r^*, p^*) be the optimal solution to the above stated dual problem and let π^* solve the primal MWM problem. Then, the standard complimentary slackness conditions are:

$$r^*_i + p^*_{\pi^*(i)} \ =\ w_{i\pi^*(i)}. \tag{20}$$

Thus, (r^*, p^*, π^*) are the optimal dual-primal solution for the MWM problem if and only if (a) π^* is a matching, (b) (r^*, p^*) satisfy (19), and (c) the triple satisfies (20). To complete the proof we will prove the existence of r^*, p^* such that (r^*, p^*, σ) satisfy (a), (b) and (c).

To this end, first note that σ is a matching by the termination condition of the algorithm; thus, condition (a)is satisfied. We'll consider the min-sum auction II algorithm for the purpose of the proof. Suppose the algorithm terminates at some iteration k. Let p^{k-1}_j and p^k_j be the prices of β_j in iterations $k-1$ and k respectively. Since all β_js are matched at the termination, from step (4) of the min-sum auction II, we obtain

$$p^k_j \ \geq\ p^{k-1}_j,\ \forall j. \tag{21}$$

At termination (iteration k), α_i is matched with $\beta_{\sigma(i)}$ or β_j is matched with $\alpha_{\sigma^{-1}(j)}$. By the definition of the min-sum auction II algorithm,

$$p_j^k \quad = \quad w_{\sigma^{-1}(j)j} - \max_{\ell \neq j} \left[w_{\sigma^{-1}(j)\ell} - p_\ell^{k-1} \right]. \qquad (22)$$

From (21) and (22), we obtain that

$$w_{\sigma^{-1}(j)j} - p_j^k \quad \geq \quad \max_{\ell \neq j} \left[w_{\sigma^{-1}(j)\ell} - p_\ell^k \right]. \qquad (23)$$

Define, $r_i^* = w_{i\sigma(i)} - p_{\sigma(i)}^k$ and $p_j^* = p_j^k$. Then, from (23) (r^*, p^*) satisfy the dual feasibility, that is (19). Further, by definition they satisfy the complimentary slackness condition (20). Thus, the triple (r^*, p^*, σ) satisfies (a), (b) and (c) as required. Hence, the algorithm min-sum auction II produces the MWM, i.e. $\sigma = \pi^*$. ■

The min-sum auction II algorithm looks very similar to the auction algorithm and inherits some of its properties. However, it also inherits some properties of the min-sum algorithm. This causes it to behave differently from the auction algorithm. The proof of convergence of auction algorithm relies on two properties of the auctioning mechanism: (a) the prices are always non-decreasing and (b) the number of unmatched nodes always decreases. By design, (a) and (b) can be shown to hold for the auction algorithm. However, it is not clear if (a) and (b) are true for min-sum auction. In what follows, we state (without proof) the result that prices are eventually non-decreasing in the min-sum auction algorithm; however it seems difficult to establish a statement similar to (b) for the min-sum algorithm as of now.

Theorem 6: If π^* is unique then in the min-sum auction II algorithm, prices eventually increase. That is, $\forall k \in \mathbb{Z}^+;\ \exists\, T > k\ s.t.\ \forall t \geq T;\ p_j^t > p_j^k,\ 1 \leq j \leq n$

Proof: Due to space limitations we cannot state the proof of Theorem (6) here. Its proof is essentially based on (i) the equivalence between the min-sum auction algorithms I and II, and (ii) arguments very similar to the ones used in the proof of Lemma 2 [5], where we relate prices with the computation tree. ■

Our simulations suggests that in the absence of the condition "$\tilde{m}_{\alpha_{i_j} \to \beta_j}^k \geq \tilde{m}_{\beta_j \to \alpha_i}^{k-1}$" from step (4) of min-sum auction I, the algorithm always terminates and finds the MWM as long as it is unique. This along with Theorem 6 leads us to the following conjecture.

Conjecture 1: If π^* is unique then the min-sum auction I terminates in a finite number of iterations if condition "$\tilde{m}_{\alpha_{i_j} \to \beta_j}^k \geq \tilde{m}_{\beta_j \to \alpha_i}^{k-1}$" is removed from step (4).

C. RELATION TO δ-RELAXATION

In the previous section, we established a relation between the min-sum and auction (with $\delta = 0$) algorithms. In [2], [3] the author extends the auction algorithm to obtain guaranteed convergence in a finite number of iterations via a δ-relaxation for some $\delta > 0$. At termination the δ-relaxed algorithm produces a triple (r^*, p^*, π^*) such that (a1) π^* is a matching, (b1) (r^*, p^*) satisfy (19) and (c1) the following modified complimentary slackness conditions are satisfied:

$$r_i^* + p_{\pi^*(i)}^* \quad \leq \quad w_{i\pi^*(i)} + \delta. \qquad (24)$$

The conditions (c1) are referred to as δ-CS conditions in [2]. This modification is reflected in the description of the auction algorithm where we have added δ to each bid in step (2). We established the relation between min-sum and auction for $\delta = 0$ in the previous section. Here we make a note that for every $\delta > 0$, the similar relation holds. To see this, we consider min-sum auction I and II where the bid computation is modified as follows: modify step (3) of min-sum auction I as $\tilde{m}_{\alpha_i \to \beta_j}^k = w_{ij} - \max_{\ell \neq j} \{ w_{i\ell} - \tilde{m}_{\beta_\ell \to \alpha_i}^{k-1} \} + \delta$, and modify step (2) of min-sum auction II as

$$b_{\alpha_i \to \beta_{j_i}} = w_{ij_i} - u_i + \delta, \text{ and } b_{\alpha_i \to \beta_j} = w_{ij} - v_i + \delta,\ j \neq j_i.$$

For these modified algorithms, we obtain the following result using arguments very similar to the ones used in Theorem 5.

Theorem 7: For $\delta > 0$, let σ be the matching obtained from the modified min-sum auction algorithm I (or II). Then, $w_\sigma \geq w_{\pi^*} - n\delta$ (i.e. σ is within $n\delta$ of the MWM).

IV. DISCUSSION AND CONCLUSION

We consider the question of finding the MWM in a weighted complete bipartite graph using the max-product belief propagation algorithm. In previous work, we had established the convergence property of the max-product (min-sum) algorithm. However, the complexity of the algorithm scaled as $O(n^4)$ for a graph with n nodes. In this paper, we first presented a simplification of this algorithm which only requires $O(n^3)$ operations and thus matching the running time of the best known algorithms for finding the MWM. The dynamics of the simplified min-sum (SMS) algorithm are very similar to that of the Auction algorithm. Motivated by this, we presented modifications of the min-sum and auction algorithms that are equivalent. We established correctness of termination points of this modification. Moreover, we applied the δ-relaxation method to this modified algorithm and studied its correctness.

The similarities mentioned above are very similar to the recently observed connection between the max-product algorithm and the LP-based dual algorithm for iterative decoding [6]. This suggests the possibility of there being a connection between the max-product and dual Linear programming algorithms for a more general class of problems on graphs.

REFERENCES

[1] J. Edmonds and R. Karp, "Theoretical Improvements in Algorithmic Efficiency for Network Flow Problems," *Jour. of the ACM*, Vol. 19, pp 248-264, 1972.

[2] D. P. Bertsekas , "The Auction Algorithm: A Distributed Relaxation Method for the Assignment Problem," Annals of Operations Research, Vol. 14, pp. 105-123, 1988.

[3] D. P. Bertsekas, "Auction Algorithms for Network Flow Problems: A Tutorial Introduction," Computational Optimization and Applications, Vol. 1, pp. 7-66, 1992

[4] M. Bayati, D. Shah and M. Sharma, "Maximum Weight Matching via Max-Product Belief Propagation," *Proc. IEEE Int. Symp. Information Theory*, 2005.

[5] M. Bayati, D. Shah and M. Sharma, "Maximum Weight Matching via Max-Product Belief Propagation," (longer version) *submitted*, available at http://www.stanford.edu/∼bayati/papers/mpmwm.ps, 2005

[6] P. O. Vontobel and R. Koetter, personal communication