Tightness of LP via Max-product Belief Propagation

Sujay Sanghavi Devavrat Shah*

Abstract

We investigate the question of tightness of linear programming (LP) relaxation for finding a maximum weight independent set (MWIS) in sparse random weighted graphs. We show that an edge-based LP relaxation is asymptotically tight for Erdos-Renyi graph $G(n, c/n)$ for $c \leq 2e$ and random regular graph $G(n, r)$ for $r \leq 4$ when node weights are i.i.d. with exponential distribution of mean 1. We establish these results, through a precise relation between the tightness of LP relaxation and convergence of the max-product belief propagation algorithm. We believe that this novel method of understanding structural properties of combinatorial problems through properties of iterative procedure such as the max-product should be of interest in its own right.

1 Introduction

The max-weight independent set (MWIS) problem is the following: given a graph with weights on the nodes, find the heaviest set of disjoint nodes. It is a canonical combinatorial optimization problem, known to be NP-hard [14] and hard to approximate [15] in the worst case. This has led to considerable interest in average-case characterizations of fundamental structural properties, as well as the hardness, of the MWIS problem. We summarize some of this work in Section 1.1.

In this paper we establish that, with high probability, the standard simple edge-based linear programming (LP) relaxation of the MWIS problem is asymptotically tight on $1 - o(1)$ fraction of the nodes, for the Erdos-Renyi graph $G(n, c/n)$ for $c \leq 2e$ and random regular graph $G(n, r)$ for $r \leq 4$, when the node weights are drawn i.i.d. with exponential distribution of mean 1. This means that problems from this ensemble are "easy" with high probability, since the LP can be solved in polynomial time [10].

We arrive at this result via an analysis of the max-product form of belief propagation. In particular, we establish the following two properties of max-product: *(a)* for any *arbitrary* problem instance, max-product succeeds[1] for a given node *only if* every LP optimum assignment is integral for that node, and *(b)* for the random weighted graphs above, max-product succeeds for almost all nodes, with high probability. To the best of our knowledge, our work represents the first instance where analysis of iterative procedures like max-product has been used as a tool to establish fundamental properties of optimization problems on graphs; usually, the analysis goes the other way. We believe that this method of analysis has the potential to shed insight into other graph-theoretic/algorithmic problems beyond the ones presented in this paper.

*Laboratory for Information and Decision Systems, MIT. Email: {sanghavi, devavrat}@mit.edu

[1]That is, the estimate of max-product converges, which may not necessarily correct.

Other motivations for our work are: *(1)* to obtain a better understanding of iterative procedures like max-product and its relation to underlying problem structure, a topic of much recent interest (for example, [33–37], and *(2)* to characterize the performance of max-product as a simple, distributed solution to the MWIS problem, in applications where such a solution is needed (e.g. wireless scheduling [11]).

We now summarize some of the most closely related work on average-case analysis of graph problems (in particular MWIS), and on the analysis of max-product. Then we summarize our main contributions, and provide an outline for the rest of the paper.

1.1 Related Work

We now summarize the most relevant existing literature in the two areas concerning this paper: average-case analysis of independent set problems, and analysis of max-product and belief propagation.

There has been much work on average-case characterizations of hard problems, see [16, 17]. For the (unweighted) independent set problem, there has been much work on the dense Erdos-Renyi graphs $G(n, \frac{1}{2})$, where it is known that the max-size independent set is of size $2\log_2 n$ almost surely. However, no algorithm is known to find efficiently an independent set of size significantly larger than $\log_2 n$, [16, 18]. Feige [19], and Feige and Krauthgamer [20] investigate the tightness of the Lovasz-Schriver hierarchy of relaxations for these graphs – relaxation upto $\log n$ level will lead to $1 - o(1)$ approximation. In constrast, here we prove tightness of LP relaxation for sparse random graphs.

There has also been substantial interest in independent set problems on sparse random graphs. Karp and Sipser [21] analyze a greedy algorithm for matchings on sparse Erdos-Renyi graphs $G(n, c/n)$; a similar analysis for independent set shows that the algorithm works for $c \leq e$ (see e.g. [8] for details). Frieze and Suen [23] investigate the success of a greedy algorithm for independent sets in random 3-regular graphs. Bollabas [22] provides an upper bound on the size of the largest independent set for regular graphs.

There is a tremendous amount of literature on the study of message-passing algorithms for inference. We will now briefly review the existing work most directly related to this paper. The main results of this paper illustrate a close relationship between the performance of the max-product algorithm, and linear programming. Such precise or semantic connections have been noticed in other contexts: in decoding of linear codes [4], weighted matching [1–3] and weighted independent set [5]. For more general inference problems, several authors [24–27] develop alternative message-passing algorithms that explicitly solve linear programming relaxations. These are related to, but not the same as, the classical max-product belief propagation that we will use in this paper.

1.2 Our Contributions and Paper Organization

In this paper we investigate the (simple, edge-based) LP relaxation of the MWIS problem for certain random ensembles. The weights on the nodes in our random ensembles are drawn from a continuous distribution, and hence the LP optimum x^* will be unique with probability 1. In general, each node i will be assigned a (possibly fractional[2]) value x_i^* at this optimum. Our main result in this paper is the following theorem.

[2]In fact, it is known [6, Theorem 64.7] that the edge-based LP is half-integral: $x_i^* = 0, 1$ or $\frac{1}{2}$

Theorem 1.1 *Let G be an Erdos-Renyi random graph $G(n, c/n)$ for $c \leq 2e$, or random regular graph $G(n, r)$ for $r \leq 4$. Suppose each node has a weight that is chosen to be i.i.d. with exponential distribution of mean 1, independent of the graph. Let x^* be the optimum of the edge-based LP relaxation of the MWIS problem (described in section 2). Let i be a node picked uniformly at random, independent of everything else*[3]*. Then, given any ε, there exists an $N(\varepsilon)$ such that $\Pr[x_i^* = 0 \text{ or } 1] \geq 1 - \varepsilon$ as long as $n \geq N(\varepsilon)$.*

The above theorem states that LP will be tight on a fraction $1-o(1)$ of the nodes, with high probability. We arrive at this result via the novel route of max-product belief propagation. Max-product is an iterative algorithm, and at every iteration produces an estimate $\hat{x}_i^t = 0, 1$ or ? (corresponding to "not in the MWIS", "in the MWIS", and "don't know" respectively) for each node i and time t. In this paper, we prove the following two results on the performance of max-product:

1. For any *arbitrary* graph with arbitrary weights, consider a node i. If there exists *any* LP optimum x^* that puts a fractional mass x_i^* on node i, then the max-product estimate will be $\hat{x}_i^t = 0$ or ? for every odd time t, and $\hat{x}_i^t = 1$ or ? for every even time t. This is Theorem 3.1.

2. For the random graph ensembles, and randomly chosen node i as in Theorem 1.1 above, given any $\varepsilon > 0$ there exists an $N(\varepsilon)$ and $t(\varepsilon)$ such that for all $n \geq N(\varepsilon)$, the max-product estimate $\hat{x}_i^t$ remains constant and equal to either 0 or 1, for all $t \geq t(\varepsilon)$. This is Theorem 4.1.

Note that the first result (Theorem 3.1) is non-asymptotic: it holds for finite graphs and number of iterations (i.e. it is not a "fixed point" analysis). This is crucial in avoiding an "order of limits" problem (between the number of iterations and the size of the problem) that may otherwise come up in establishing the overall result. The second result is established using ideas from the method of local weak convergence [7, 8].

In addition to the fact that they together immediately imply our main result above, we believe that each of the two theorems above are interesting in their own right. The first theorem generalizes to the case when "clique factors" are added to max-product, and clique constraints to the corresponding LP. However, in this paper we will concentrate only on the simplest edge-based case.

The theorems also shed light on the usefulness of max-product as a distributed heuristic for the MWIS problem. Consider first an arbitrary graph. In light of Theorem 3.1, we can stop max-product after a certain number of iterations and check for one-step agreement: if the estimate $\hat{x}_i^t = \hat{x}_i^{t+1} = 0$ or 1, then we know that $x_i^* = 1$ for any LP optimum. This is then also consistent with the MWIS (see Lemma 2.1 below). This means that the set of nodes for which $\hat{x}_i^t = \hat{x}_i^{t+1} = 1$ will form an independent set; we can stop max-product at any time and have a candidate independent set (although it may not be the MWIS). If we restrict our attention to the ensembles considered above, if $n \geq N(\varepsilon)$ and max-product is run for a sufficient number of iterations $t \geq t(\varepsilon)$, this set obtained from one-step agreement will be pretty large in size. The fact that the weights come from an exponential distribution means that this candidate independent set will be a very good approximation of the MWIS. This is shown in Theorem 4.2.

The rest of the paper is organized as follows. In Section 2 we lay out the groundwork and preliminaries. We also describe precisely the max-product algorithm we are considering in this paper, and state some of its other known properties. In Section 3 we state and prove Theorem 3.1. In Section 4 we state and prove Theorem 4.1, and Theorem 4.2.

[3] Or, alternatively, one can think of having an a-priori node numbering before the edges and weights are picked.

2 Preliminaries: MWIS, LP relaxation and max-product

Consider a graph $G = (V, E)$, with a set V of nodes and a set E of edges. Let $\mathcal{N}(i) = \{j \in V : (i, j) \in E\}$ be the neighbors of $i \in V$. Positive weights $w_i, i \in V$ are associated with each node. A subset of V will be represented by vector $\mathbf{x} = (x_i) \in \{0, 1\}^{|V|}$, where $x_i = 1$ means i is in the subset $x_i = 0$ means i is not in the subset. A subset $\mathbf{x}$ is called an *independent set* if no two nodes in the subset are connected by an edge: $(x_i, x_j) \neq (1, 1)$ for all $(i, j) \in E$. We are interested in finding a maximum weight independent set (MWIS) $\mathbf{x}^*$. This can be naturally posed as an integer program, denoted below by IP. The *linear programing relaxation*[4] of IP is obtained by replacing the integrality constraints $x_i \in \{0, 1\}$ with the constraints $x_i \geq 0$. We will denote the corresponding linear program by LP. The dual of LP is denoted below by DUAL.

$$\mathsf{IP}: \ \max \ \sum_{i=1}^{n} w_i x_i \qquad \text{over } x_i \in \{0,1\},$$
$$\text{s.t. } \ x_i + x_j \leq 1 \ \text{ for all } (i,j) \in E,$$

$$\mathsf{LP}: \ \max \ \sum_{i=1}^{n} w_i x_i, \qquad \text{over } x_i \geq 0,$$
$$\text{s.t. } \ x_i + x_j \leq 1 \ \text{ for all } (i,j) \in E,$$

It is well-known that LP can be solved efficiently, and if it has an integral optimal solution then this solution is an MWIS of G. If this is the case, we say that there is no *integrality gap* between LP and IP or equivalently that the LP relaxation is *tight*. We refer an interested reader to book by Schrijver [6] for many interesting properties of the LP. We note one property: partial correctness.

Lemma 2.1 ([6],Corollary 64.9a)) *LP optima are partially correct: for any graph, any LP optimum x^* and any node i, if the mass x_i^* is integral then there exists an MWIS of G for which i's membership is given by x_i^*.*

The classical max-product algorithm is a heuristic that can be used to find the MAP assignment of a probability distribution. Before, we state the (simplified) max-product algorithm applied to the MWIS problem, we state probability distribution whose MAP solution corresponds to solution of MWIS problem for completeness. Now, given an MWIS problem on $G = (V, E)$, associate a binary random variable X_i with each $i \in V$ and consider the following joint distribution: for $\mathbf{x} \in \{0, 1\}^n$,

$$p(\mathbf{x}) \ = \ \frac{1}{Z} \prod_{(i,j)\in E} \mathbf{1}_{\{x_i + x_j \leq 1\}} \prod_{i \in V} \exp(w_i x_i), \tag{1}$$

where Z is the normalization constant. In the above, $\mathbf{1}$ is the standard indicator function: $\mathbf{1}_{\text{true}} = 1$ and $\mathbf{1}_{\text{false}} = 0$. It is easy to see that $p(\mathbf{x}) = \frac{1}{Z} \exp\left(\sum_i w_i x_i\right)$ if $\mathbf{x}$ is an independent set, and $p(\mathbf{x}) = 0$ otherwise. Thus, any MAP estimate $\arg\max_{\mathbf{x}} p(\mathbf{x})$ corresponds to a maximum weight independent set of G.

Here, we present a simplified version of the max-product algorithm (obtained by taking logarithm of ratio of messages for the original algorithm) – we refer an interested reader to [5] for details on the transformation. The algorithm is iterative; in iteration t each node i sends a *message* $\{\gamma_{i\to j}^t\}$ to each neighbor $j \in \mathcal{N}(i)$ based on $\{\gamma_{k\to i}^t\}, k \in \mathcal{N}(i)$. Each node i also maintains an *estimate* of its assignment in independent set $\{x_i(\gamma^t)\}$ based on messages it received $\{\gamma_{j\to i}^t\}$, $j \in \mathcal{N}(i)$. The following describes the message and estimate updates, as well as the final output.

[4]Other (tighter) LP relaxations of IP are possible, and some of our results carry over to those relaxations as well. However, in this paper we will concentrate only on the LP relaxation presented above.

Max-product for MWIS

(o) Initially, $t = 1$ and $\gamma^1_{i\to j} = 0$ for all $(i,j) \in E$.

(i) The messages are updated as follows: for $t > 1$,

$$\gamma^t_{i\to j} = \left(w_i - \sum_{k\in\mathcal{N}(i)-j} \gamma^{t-1}_{k\to i} \right)_+, \tag{2}$$

(ii) Estimate max. wt. independent set $\mathbf{x}(\gamma^t)$ as follows:

$$x_i(\gamma^t) \;=\; \begin{cases} 1 & \text{if} \quad w_i > \sum_{k\in\mathcal{N}(i)} \gamma^t_{k\to i} \\ 0 & \text{if} \quad w_i < \sum_{k\in\mathcal{N}(i)} \gamma^t_{k\to i} \\ ? & \text{if} \quad w_i = \sum_{k\in\mathcal{N}(i)} \gamma^t_{k\to i}. \end{cases} \tag{3}$$

(iv) Update $t = t + 1$; repeat from (i)

2.1 Max-product: known properties

A popular technique for analyzing max-product is to consider its fixed points [28–30]. Here, we list relevant properties of the max-product for MWIS that are established in [5] for setting up context for the results stated in Section 3. Note that a set of messages γ^* is a fixed point of max-product if, for all $(i,j) \in E$,

$$\gamma^*_{i\to j} = \left(w_i - \sum_{k\in\mathcal{N}(i)-j} \gamma^*_{k\to i} \right)_+ \tag{4}$$

The following is a summary of results from [5].

Theorem 2.1 *[5] There exists at least one fixed point γ^* such that $\gamma^*_{i\to j} \in [0, w_i]$ for each $(i,j) \in E$. Given such a fixed point γ^*, let $\mathbf{x}(\gamma^*) = (x_i(\gamma^*))$ be the corresponding estimate. Define $\mathbf{y} = (y_i) \in [0,1]^n$ as follows: $y_i = \frac{1}{2}$ if $x_i(\gamma^*) =?$, and $y_i = x_i(\gamma^*)$ otherwise. Then, $\mathbf{y}$ corresponds to an extreme point of the LP for MWIS.*

Theorem 2.1 implies that the fixed point estimate of max-product for MWIS is an extreme point of LP, and hence one that maximizes *some* weight function consisting of positive node weights. Note however that this *may not* be the true weights w_i. In other words, given any MWIS problem with graph G and weights w, each max-product fixed point represents the optimum of the LP relaxation of some MWIS problem on the same graph G, but possibly with different weights $\widehat{w}$. The fact that max-product estimates optimize a different weight function means that both eventualities are possible: LP giving the correct answer but max-product failing, and vice versa. In [5], two examples are presented for each one of these situations. These examples indicate that it may not be possible to resolve the question of relative strength of the two procedures based solely on an analysis of the fixed points of max-product.

3 Max-product: Convergence and tightness of LP

The max-product is a deterministic iterative algorithm. It may have multiple fixed points. In which case, it converges (if it does) to a fixed point depending upon its starting condition. In absence of prior information, the "natural" (and popular) initialization of messages is the one described in Section 2 (i.e. $\gamma^0_{i\to j} = 0$, for all $(i,j) \in E$. In this section, we directly analyze the performance of max-product algorithm with these initial conditions. We show that the resulting estimates are very exactly characterized by optima of the *true* LP, at every time instant (not just at fixed points). This implies that, if a fixed point is reached, it will exactly reflect an optimum of LP. Our main theorem for this section is stated below.

Theorem 3.1 *Given any MWIS problem on weighted graph G, suppose max-product is started from the initial condition $\gamma = 0$. Then, for any node $i \in G$.*

1. *If there exists* any *optimum x^* of* LP *for which the mass assigned to edge i satisfies $x^*_i > 0$, then the max-product estimate $x_i(\gamma^t)$ is 1 or ? for all* odd *times t.*
2. *If there exists* any *optimum x^* of* LP *for which the mass assigned to i satisfies $x^*_i < 1$, then the max-product estimate $x_i(\gamma^t)$ is 0 or ? for all* even *times t.*

An important and direct consequence of the above result is as follows.

Corollary 3.1 *If* LP *has non-integral optima, i.e. there is i such that $x^*_i \in (0,1)$. Then, estimate $x_i(\gamma^t)$ either oscillates or converges to ?.*

The proof of this theorem relies on the computation tree interpretation of max-product estimates. We now specify this interpretation for our problem, and then prove Theorem 3.1.

3.1 Computation Tree for MWIS

The proof of Theorem 3.1 relies on the well-known computation tree interpretation [28, 32] of the loopy max-product estimates. In this section we briefly outline this interpretation. For any node i, the *computation tree* at time t, denoted by $T_i(t)$, is defined recursively as follows: $T_i(1)$ is just the node i. This is the *root* of the tree, and in this case is also its only leaf. The tree $T_i(t)$ at time t is generated from $T_i(t-1)$ by adding to each leaf of $T_i(t-1)$ a copy of each of its neighbors in G, *except for the one neighbor that is already present in $T_i(t-1)$*. Each node in T_i is a copy of a node in G, and the weights of the nodes in T_i are the same as the corresponding nodes in G. As an example, Figure 1 presents computation tree $T_a(4)$ for the node a in graph G at time $t = 4$.

Lemma 3.1 *For any node i at time t, (a) $x_i(\gamma^t) = 1$ if and only if the root of $T_i(t)$ is a member of* every *MWIS on $T_i(t)$; (b) $x_i(\gamma^t) = 0$ if and only if the root of $T_i(t)$ is not a member of* any *MWIS on $T_i(t)$; and (c) $x_i(\gamma^t) =?$ else.*

Thus the max-product estimates correspond to max-weight independent sets on the computation trees $T_i(t)$, as opposed to on the original graph G.

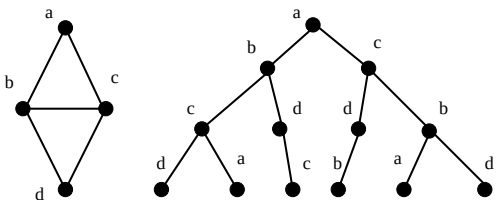

Figure 1: An example of computation tree for $t = 4$ iterations for a 4 node graph.

3.2 Proof of Theorem 3.1

We now prove Theorem 3.1. We will present the proof of part (1). The proof of part (2) follows from very similar arguments and hence we will skip it.

We will prove part 1 through contradiction. To this end, suppose part 1 is not true. That is, there exists node i, an optimum x^* of LP with $x_i^* > 0$, and an odd time t at which the estimate is $\hat{x}_i^t = 0$. For brevity, in the remainder of the proof we will use the notation $\hat{x}_i^t = x_i(\gamma^t)$ for the estimates. Let $T_i(t)$ be the corresponding computation tree. Using Lemma 3.1 this means that the root i is *not* a member of any MWIS of $T_i(t)$. Let I be some MWIS on $T_i(t)$. We now define the following set of nodes

$$I^* = \left\{ j \in T_i(t) \,:\, j \notin I, \text{ and copy of } j \text{ in } G \text{ has } x_j^* > 0 \right\}$$

In words, I^* is the set of nodes in $T_i(t)$ which are not in I, and whose copies in G are assigned strictly positive mass by the LP optimum x^*.

Note that by assumption the root $i \in I^*$ and $i \notin I$. Now, from the root, recursively build a *maximal alternating subtree* S as follows: first add root i, which is in $I^* - I$. Then add all neighbors of i that are in $I - I^*$. Then add all *their* neighbors in $I^* - I$, and so on. The building of S stops either when it hits the bottom level of the tree, or when no more nodes can be added while still maintaining the alternating structure. Note the following properties of S:

- S is the disjoint union of $(S \cap I)$ and $(S \cap I^*)$.
- For every $j \in S \cap I$, all its neighbors in I^* are included in $S \cap I^*$. Similarly for every $j \in S \cap I^*$, all its neighbors in I are included in $S \cap I$.
- Any edge (j, k) in $T_i(t)$ has at most one endpoint in $(S \cap I)$, and at most one in $(S \cap I^*)$.

We now state a lemma, which we will prove later. The proof uses the fact that t is odd.

Lemma 3.2 *The weights satisfy* $w(S \cap I) \leq w(S \cap I^*)$.

We now use this lemma to prove the theorem. Consider the set I' which changes I by flipping S:

$$I' = I - (S \cap I) + (S \cap I^*)$$

We first show that I' is also an independent set on $T_i(t)$. This means that we need to show that every edge (j, k) in $T_i(t)$ touches at most one node in I'. There are thus three possible scenarios for edge (j, k):

- $j, k \notin S$. In this case, membership of j, k in I' is the same as in I, which is an independent set. So (j, k) has at most one node touching I'.
- One node $j \in S \cap I$. In this case, $j \notin I'$, and hence again at most one of j, k belongs to I'.
- One node $k \in S \cap I^*$ but other node $j \notin S \cap I$. This means that $j \notin I$, because every neighbor of k in I should be included in $S \cap I$. This means that $j \notin I'$, and hence only node $k \in I'$ for edge (j, k).

Thus I' is an independent set on $T_i(t)$. Also, by Lemma 3.2, we have that

$$w(I') \ \geq \ w(I)$$

However, I is an MWIS, and hence it follows that I' is also an MWIS of $T_i(t)$. However, by construction, root $i \in I'$, which violates the fact that $\hat{x}_i(t) = 0$. The contradiction is thus established, and part (1) of the theorem is proved. ■

3.2.1 Remaining proofs

The proof of this lemma involves a perturbation argument on the LP. For each node $j \in G$, let m_j denote the number of times j appears in $S \cap I$ and n_j the number of times it appears in $S \cap I^*$. For $\varepsilon > 0$, define

$$x \ = \ x^* + \varepsilon(m - n). \tag{5}$$

We now show state a lemma that is proved immediately following this one.

Lemma 3.3 *x is a feasible point for* LP*, for small enough ε.*

Using the above, we will complete the proof of Lemma 3.2. Since x^* is an optimum of LP, it follows that $w'x \leq w'x^*$, and so $w'm \leq w'n$. However, by definition, $w'm = w(S \cap I)$ and $w'n = w(S \cap I^*)$. This finishes the proof of Lemma 3.2. ■

Proof of Lemma 3.3: Now, we complete proof of the only remaining part: Lemma 3.3. We wish to show that x as defined in (5) is a feasible point for LP, for small enough $\varepsilon > 0$. To do so we have to check node constraints $x_j \geq 0$ and edge constraints $x_j + x_k \leq 1$ for every edge $(j, k) \in G$. Consider first the node constraints. Clearly we only need to check them for any j which has a copy $j \in I^* \cap S$. If this is so, then by the definition (3.2) of I^*, $x^*_j > 0$. Thus, for any m_j and n_j, making ε small enough can ensure that $x^*_j + \varepsilon(m_j - n_j) \geq 0$.

Before we proceed to checking the edge constraints, we make two observations. Note that for any node j in the tree, $j \in S \cap I$ then

- $x^*_j < 1$, i.e. the mass x^*_j put on j by the LP optimum x^* is *strictly* less than 1. This is because of the alternating way in which the tree is constructed: a node j in the tree is included in $S \cap I$ *only if* the parent p of j is in $S \cap I^*$ (note that the root $i \in S \cap I^*$ by assumption). However, from the definition of I^*, this means that $x^*_p > 0$, i.e. the parent has positive mass at the LP optimum x^*. This means that $x^*_j < 1$, as having $x^*_j = 1$ would mean that the edge constraint $x^*_p + x^*_j \leq 1$ is violated.

- j is not a leaf of the tree. This is because S alternates between I and I^*, and starts with I^* at the root in level 1 (which is odd). Hence $S \cap I$ will occupy even levels of the tree, but the tree has odd depth (by assumption t is odd).

Now consider the edge constraints. For any edge (j,k), if the LP optimum x^* is such that the constraint is loose – i.e. if $x^*_j + x^*_k < 1$ – then making ε small enough will ensure that $x_j + x_k \leq 1$. So we only need to check the edge constraints which are tight at x^*.

For edges with $x^*_j + x^*_k = 1$, every time any copy of one of the nodes j or k is included in $S \cap I$, the other node is included in $S \cap I^*$. This is because of the following: if j is included in $S \cap I$, and k is its parent, we are done since this means $k \in S \cap I^*$. So suppose k is not the parent of j. From the above it follows that j is not a leaf of the tree, and hence k will be one of its children. Also, from above, the mass on j satisfies $x^*_j < 1$. However, by assumption $x^*_j + x^*_k = 1$, and hence the mass on k is $x^*_k > 0$. This means that the child k has to be included in $S \cap I^*$.

It is now easy to see that the edge constraints are satisfied: for every edge constraint which is tight at x^*, every time the mass on one of the endpoints is increased by ε (because of that node appearing in $S \cap I$), the mass on the other endpoint is decreased by ε (because it appears $S \cap I^*$). ■

4 Max-product for Random Weighted Graphs

In this section, we establish the correctness and convergence of max-product algorithm when underlying graph is random and node weights are chosen as per exponential distribution. Specifically, we consider two types of sparse random graphs, $G(n, c/n)$ and $G(n, r)$:

1. The $G(n, c/n)$ has n nodes. An edge is present between any node-pair i, j with probability c/n independently. Thus, on average $c(n-1)/2$ edges are present.

2. The $G(n, r)$ has n nodes. It is formed by sampling one of the r-regular n node graph uniformly at random.

In either of these two cases, we assign node weight randomly. Specifically, let W_i denote the (random) weight of node i. Then, W_i are independent and identically distributed with exponential distribution of mean 1. That is, for any $\zeta \geq 0$,

$$\Pr(W_i \geq \zeta) = \exp(-\zeta).$$

4.1 Results

Convergence of max-product. First, we establish that for $1 - o(1)$ fraction of nodes, the algorithm converges after *finitely* many iterations. Formally, we state the result as follows.

Theorem 4.1 *Consider graph $G(n, c/n)$ or $G(n, r)$ with node weights assigned independently according to exponential distribution of rate 1. Let $c \leq 2e$ and $r \leq 4$. Then, for any $\varepsilon > 0$, there exists large enough*

$N(\varepsilon)$ and $T(\varepsilon)$ such that if $n \geq N(\varepsilon)$, then following holds: for any node in $G(n, c/n)$ or $G(n, r)$, say i, the $x_i^t (= x_i(\gamma^t))$ converges to the correct value, x_i^* with probability[5] at least $1 - \varepsilon$ for $t \geq T(\varepsilon)$.

Correctness of max-product. Theorem 4.1 implies that almost all nodes converge to the correct solution. However, questions that remain: (a) how to identify these 'converged' nodes? and (b) do they have $1 - o(1)$ fraction of weight of the MWIS? Indeed, we answer both of these questions in *affirmative*.

We state the following simple stopping condition under which we will establish that all converged nodes get assigned the correct values, while all other nodes get assigned values 0. As we shall establish, thus resulting assignment is indeed an independent set with $1 - o(1)$ fraction of weight of MWIS with high probability. Now the stopping condition and its approximation property.

Stopping condition. At the end of iteration t, generate an estimate $\hat{I}^t$ using $\mathbf{x}^t$ as $\{i \in V : x_i^t = x_i^{t-1} = 1\}$.

Theorem 4.2 *Under the setup of Theorem 4.1, let algorithm stops after $t \geq T(\varepsilon)$ steps producing $\mathbf{x}^t$. Let $\hat{I}^t$ be the independent set obtained from $\mathbf{x}^t$ as per the stopping condition described above. Then, $n \geq N_1(\varepsilon)$ with large enough $N_1(\varepsilon)$, the weight of $\hat{I}^t$, $W(\hat{I}^t)$ is such that*

$$\Pr\left(\frac{|W(\hat{I}^t) - W(I^*)|}{W(I^*)} \geq \delta(\varepsilon)\right) \leq \varepsilon,$$

where $\delta(\varepsilon) = O(\varepsilon \log(1/\varepsilon)) \to 0$ as $\varepsilon \to 0$. Here, I^* is the maximum weight independent set and $W(I^*)$ is its weight.

4.1.1 Outline of proof

We now present a brief outline of the proof, and then present the details in appendix due to space constraints. Our results use the method of local weak convergence [7], and specifically the results of Gamarnik, Nowicki and Swirszcz [8]. Under the random graph models considered here, i.e. random regular or Erdos-Renyi graph, for almost all nodes of the graph their local neighborhood looks like a tree (see Lemma A.1). Now, under random selection of weights the assignment of node values under MWIS is determined by the local neighborhood for almost all nodes of the graph (see Lemma A.3) – this property is also known as the *correlation decay* property. Now, max-product produces the correct estimate for each node with respect to its computation tree. The computation tree of a node is equal to the local neighborhood as long as the local neighborhood is tree (recall Lemma 3.1). Therefore, it is likely that for almost all nodes the max-product will produce the correct estimate after finitely many iterations. In what follows, we will make this precise by overcoming important technical subtlities. The correctness of stopping condition would follow from certain "anti-monotonicity" property of the max-product estimate procedure. The good approximation property of the resulting estimate would follow from certain extremality properties of Exponential distribution (see Lemma A.4).

[5]Here, the probability distribution is induced by the choice of random graph and weights.

References

[1] M. Bayati, D. Shah and M. Sharma, "Max-product for maximum weight matching: convergence, correctness and LP duality," Accepted to appear in *IEEE Information Theory Transactions*, March 2008. Preliminary versions appeared in *IEEE ISIT*, 2005 and 2006.

[2] S. Sanghavi, D. Malioutov and A. Willsky, "Belief Propagation and LP Relaxation for Weighted Matching in General Graphs," *NIPS 2007*

[3] M. Bayati, C. Borgs, J. Chayes, and R. Zecchina "Belief-Propagation for Weighted b-Matchings on Arbitrary Graphs and its Relation to Linear Programs with Integer Solutions," at *http://arxiv.org/abs/0709.1190*

[4] P.O. Vontobel and R. Koetter,"On the relationship between linear programming decoding and min-sum algorithm decoding," in *Proc. ISITA 2004, Parma, Italy,* p. 991-996, October 10-13, 2004.

[5] S. Sanghavi, D. Shah and A. Willsky "Message-passing for max-weight independent set," in *NIPS 2007.*

[6] A. Schrijver, *Combinatorial Optimization. Polyhedra and Efficiency,* Springer-Verlag, Berlin Heidelberg 2003.

[7] D. Aldous and J. M. Steele, "The objective method: Probabilistic combinatorial optimization and local weak convergence," in *Discrete combinatorial probability,* H. Kesten, (Editor), Springer, New York, 2003, pp. 1-72.

[8] D. Gamarnik, T. Nowicki and G. Swirscsz, "Maximum Weight Independent Sets and Matchings in Sparse Random Graphs. Exact Results using the Local Weak Convergence Method," in *Random Structures and Algorithms*, Vol.28, No. 1, pp. 76-106. 2005

[9] D. Bertsekas, "Non Linear Programming", *Aetna Scientific Publications.*

[10] Grotschel, L. Lovasz and Schrijver, "Polynomial algorithms for perfect graphs," in *Annals of Discrete Mathematics,* 21 (1984) 325-356.

[11] K. Jung and D. Shah, "Low Delay Scheduling in Wireless Networks," *IEEE ISIT*, 2007.

[12] S. Janson, T. Luczak and A. Rucinski, "Random Graphs," John Wiley and Sons Inc., 2000.

[13] J. H. Spencer, "Ten Lectures in Probabilistic Method," Second Edition, SIAM, 1994.

[14] R. M. Karp, "The probabilistic analysis of some combinatorial search algorithms," in *Algorithms and complexity (Proc. Sympos.)*, pages 1-19. Academic Press, 1976.

[15] J. Hastad, "Clique is hard to approximate within $n^{1-\varepsilon}$," in *37th Annual Symposium on Foundations of Computer Science,* pages 627-636, 14-16, October 1996.

[16] A. Frieze and C. McDiarmid, "Algorithmic theory of random graphs," in *Random Structures Algorithms,* 10(1-2):5-42, 1997.

[17] A. Bogdanov and L. Trevisan, "Average-case complexity," in *Foundations and Trends in Theoretical Computer Science* Volume 2, Issue 1, October 2006.

[18] R. M. Karp "The probabilistic analysis of some combinatorial search algorithms," in *Algorithms and complexity (Proc. Sympos.)*, pages 1-19. Academic Press, 1976.

[19] U. Feige, "Randomized graph products, chromatic numbers, and the Lovasz θ-function",in *Combinatorica*, 17(1):79-90, 1997.

[20] U. Feige and R. Krauthgamer, "The probable value of the Lovasz-Schrijver relaxations for maximum independent set," in *SIAM Journal on Computing* Volume 32 , Issue 2 (2003), pages 345 - 370.

[21] R. Karp and M. Sipser, "Maximum matchings in sparse random graphs," in *22nd Annu Symp Foundations of Computer Science,* 1981, pp. 364-375

[22] B. Bollabas, "The independence radio of regular graphs," in Proc. Amer. Math. Soc. 83(2) (1981), 433-436.

[23] A. Frieze and S. Suen, "On the independence number of random cubic graphs," Random Structures Algorithms 5 (1994), 649-664.

[24] M. J. Wainwright and M. I. Jordan "Graphical models, exponential families, and variational inference," *UC Berkeley, Dept. of Statistics, Technical Report 649.* September, 2003.

[25] V. Kolmogorov and M. Wainwright, "On optimality of tree-reweighted max-product message-passing," in *Uncertainty in Artificial Intelligence,* July 2005, Edinburgh, Scotland.

[26] Y. Weiss, C. Yanover and T. Meltzer, "MAP Estimation, Linear Programming and Belief Propagation with Convex Free Energies," in *UAI 2007*

[27] A. Globerson and T. Jaakkola, "Fixing max-product: Convergent message passing algorithms for MAP LP-relaxations," in *NIPS 2007*

[28] Y. Weiss and W. Freeman, "On the optimality of solutions of the max-product belief-propagation algorithm in arbitrary graphs," in *IEEE Trans. on Information Theory,* vol. 47, no. 2, pp. 736-744, Feb. 2001.

[29] S. M. Aji, G. B. Horn, and R. J. McEliece, "On the convergence of iter- ative decoding on graphs with a single cycle," in *Proc. 1998 IEEE Int. Symp. Information Theory,* Cambridge, MA, Aug. 1998, p. 276

[30] J. Yedidia, W. Freeman, and Y. Weiss, "Constructing Free-Energy Approximations and Generalized Belief Propagation Algorithms," in *IEEE Transactions on Information Theory,* Vol. 51, Issue 7, pp. 2282-2312, July 2005

[31] D. Aldous, "The $\zeta(2)$ limit in the random assignment problem," in *Random Structures and Algorithms* 18 (2001), 381-418.

[32] S. Tatikonda and M. I. Jordan, "Loopy belief propagation and Gibbs measures," in *UAI 2002.*

[33] E. Maneva and E. Mossel and M. J. Wainwright, "A New look at Survey Propogation and its Generalizations," in *Journal of the ACM*, 54(4), 2007.

[34] U. Feige and E. Mossel and D. Vilenchik, "Complete convergence of message passing algorithms for some satisfiability problems," in *Proceedings of Random*, 2006.

[35] D. Achlioptas and F. Ricci-Tersenghi, "On the solution-space geometry of random constraint satisfaction problems," in *Proceedings of STOC*, 2006.

[36] M. Mezard and T. Mora and R. Zecchina, "Clustering of Solutions in the Random Satisfiability Problem," Phys. Rev. Lett. 94 (2005), 197205.

[37] M. Bayati and D. Gamarnik and D. Katz and C. Nair and P. Tetali, " Simple deterministic approximation algorithms for counting matchings," in *Proceedings of STOC*, 2007.

A Proofs of Theorems 4.1 and 4.2

In this appendix, we present proofs of the two remaining Theorems 4.1 and 4.2. We start with some useful known properties. Then, we study certain "local convergence" properties of max-product on random graphs models considered in this paper. Building upon these, we conclude the proof of Theorem 4.1. Finally, using Theorem 4.1 and an extremal property of Exponential random variables we conclude the proof of Theorem 4.2. We note that in this proof we assume that algorithm starts at time $t = 0$ and not $t = 1$ for a peculiar notational reason. Clearly, it is of no non-trivial relevance.

A.1 Useful properties

Here, we describe useful definitions, notations and properties for proving Theorem 4.1. To this end, consider a fixed node i in graph G. Define, $V_i(t) \subset V$ as

$$V_i(t) = \{j \in V : \text{there is a path between } i \text{ and } j \text{ of length no more than } t\}.$$

Let $E_i(t) \subset E$ be set of edges incident between these vertices and $G_i(t) = (V_i(t), E_i(t))$ be the subgraph of G thus created. As defined earlier, let $T_i(t)$ be the computation tree of node i till iteration t. Then, it is straightforward that $T_i(t) = G_i(t)$ (in terms of graph structure only) if $G_i(t)$ is itself a tree.

Some notation and definitions before stating result about structural properties of $G_i(t)$ when $G = G(n, c/n)$ or $G = G(n, r)$. A Poisson tree of depth t and parameter c, denoted as $\mathcal{T}(c, t)$, is constructed as follows: starting with root, say 0, add Poisson(c) number of children to it. Recursively, for each of thus created children, add Poisson(c) number of children independently till the tree has depth t. A regular tree of depth t and parameter r, denoted as $\mathbb{T}(r, t)$, is constructed as follows: starting with root, say 0, add r number of children to it. Recursively, for each of thus created children, add $r - 1$ number of children till the tree has depth t.

Now, we state the following well-known (and very important for us) property about the local structure of $G(n, c/n)$ and $G(n, r)$ (see [12] and [13] respectively for details).

Lemma A.1 *Consider graph $G = G(n, c/n)$ or $G = G(n, r)$ with finite values of c and r. Consider a fixed node i (numbering of nodes is done prior to selection of edges). Then, as $n \to \infty$,*

(a) For $G = G(n, c/n)$, the $G_i(t)$ converges (in distribution) to the Poisson tree, $\mathcal{T}(c, t)$;

(b) For $G = G(n, r)$, the $G_i(t)$ converges (in distribution) to the regular tree, $\mathbb{T}(r, t)$.

A.2 Local convergence of max-product

Consider the max-product algorithm running at a particular node, say i. In what follows, i is always used to denote a fixed node[6]. We wish to understand the evolution its estimate x_i^t, which depends on the messages

[6]This fixed node i is chosen a priori selection of random graph structure and weights or equivalently, its selection is done uniformly at random from n nodes for the purposes of the proofs.

γ^t. Specifically, as we stated earlier, the messages γ^t can be defined recursively on the computation tree $T_i(t)$ as follows: node k generates message for its parent, say $p(k)$, using messages from its children set, say $c(k)$ and its weight W_k as

$$\gamma^t_{k\to p(k)} = \max\left(0, W_k - \sum_{\ell\in c(k)} \gamma^t_{\ell\to k}\right).$$

When, $c(k) = \emptyset$ that is k is a leaf node of $T_i(t)$, then due to 0 initial condition,

$$\gamma^t_{k\to p(k)} = \max(0, W_k) \;\; = \;\; W_k,$$

since $W_k \geq 0$ with probability 1. Call the γ^t obtained with this 0 initial conditions as $\gamma^t(\mathbf{0})$.

Now, suppose initial condition at a leaf node k, that is at depth t of the computation tree, be W_k instead of 0. Equivalently, if incoming messages to k (at the depth t) summed upto $\geq W_k$, then

$$\gamma^t_{k\to p(k)} = \max(0, W_k - W_k) \;\; = \;\; 0.$$

Call the γ^t obtained along all the edges of computation tree $T_i(t)$ with such initial condition as $\gamma^t(\mathbf{W})$: that is, all leaf nodes at level t of $T_i(t)$ have initial condition equal to their node weights, while leaf nodes[7] at level $< t$ have the usual initial condition 0.

Finally, consider starting algorithm with initial condition $L_k \in [0, W_k]$ at leaf node k at depth t in $T_i(t)$. Then, the messages from leaf nodes, say k of $T_i(t)$, is

$$0 \leq \gamma^t_{k\to p(k)} = \max(0, W_k - L_k) \leq W_k.$$

Let us denote the message obtained by starting with initial condition $\mathbf{L}$ (vector of initial condition values for leaf nodes at depth t of $T_i(t)$) as $\gamma^t(\mathbf{L})$.

The above discussion implies the following: for $t = 1$, for any j which is children of i and any starting condition $\mathbf{0} \leq \mathbf{L} \leq \mathbf{W}$ (component-wise inequality),

$$\gamma^t_{j\to i}(\mathbf{0}) \geq \gamma^t_{j\to i}(\mathbf{L}) \geq \gamma^t_{j\to i}(\mathbf{W}).$$

This non-increasing behavior of messages received at root node as initial condition is increasing holds true for all odd t inductively (can be easily verified).

Lemma A.2 *Consider an odd t, fixed node i and its computation tree $T_i(t)$. Then, for any non-negative starting condition (for leaf nodes at depth t) $\mathbf{0} \leq \mathbf{L} \leq \mathbf{W}$ (component-wise) and any children j of root node i,*

$$\gamma^t_{j\to i}(\mathbf{0}) \geq \gamma^t_{j\to i}(\mathbf{L}) \geq \gamma^t_{j\to i}(\mathbf{W}).$$

Next, we consider the estimation of node i based on its messages at time t. For convenience, we define notion of bonus at node i at time t, denote as B^t_i as follows:

$$B^t_i = W_i - \sum_{j\in\mathcal{N}(i)} \gamma^t_{j\to i}.$$

[7]Such leaf nodes can exists for Poisson tree, but will not exist for a regular tree.

As per max-product algorithm, the estimation $x_i^t = 1$ if $B_i^t > 0$; $x_i^t = 0$ if $B_i^t < 0$ and ? otherwise. Let $B_i^t(\mathbf{0}), B_i^t(\mathbf{L})$ and $B_i^t(\mathbf{W})$ denote bonus values when algorithm is started with initial conditions $\mathbf{0} \leq \mathbf{L} \leq \mathbf{W}$ for leaf nodes of $T_i(t)$ respectively. From Lemma A.2 and definition of bonus, it follows that for any odd t,

$$B_i^t(\mathbf{0}) \;\; \leq \;\; B_i^t(\mathbf{L}) \;\leq\; B_i^t(\mathbf{W}). \tag{6}$$

Due to very similar reasoning, it also follows that the bonuses have anti-monotone property starting with $\mathbf{0}$ initial condition. That is, for any odd t

$$B_i^t(\mathbf{0}) \;\; \leq \;\; B_i^{t+1}(\mathbf{0}). \tag{7}$$

The following result is direct adaption of [8, Theorem 8]. It will be essential to complete our result.

Lemma A.3 *Consider a fixed $\phi \in [0, \infty)$ and initial conditions $\mathbf{0} \leq \mathbf{L} \leq \mathbf{W}$ for leaf nodes at depth t of $T_i(t)$. Let $\varepsilon > 0$ be given. Then, there exists an odd $t(\varepsilon)$ large enough such that the following holds: for $t = t(\varepsilon)$,*

(a) Let, $G_i(t)$ (and hence $T_i(t)$) be (distributed as) Poisson tree $\mathcal{T}(c, t)$ with $c \leq 2e$. Then,

$$\Pr\left(B_i^t(\mathbf{0}) \leq \phi\right) \leq \Pr\left(B_i^t(\mathbf{L}) \leq \phi\right) \leq \Pr\left(B_i^t(\mathbf{W}) \leq \phi\right) \leq \Pr\left(B_i^t(\mathbf{0}) \leq \phi\right) + \varepsilon.$$

(b) Let, $G_i(t)$ (and hence $T_i(t)$) be (distributed as) regular tree $\mathbb{T}(r, t)$ with $r \leq 4$. Then,

$$\Pr\left(B_i^t(\mathbf{0}) \leq \phi\right) \leq \Pr\left(B_i^t(\mathbf{L}) \leq \phi\right) \leq \Pr\left(B_i^t(\mathbf{W}) \leq \phi\right) \leq \Pr\left(B_i^t(\mathbf{0}) \leq \phi\right) + \varepsilon.$$

A.3 Proof of Theorem 4.1: putting things together

Now we complete the proof of Theorem 4.1 using Lemmas A.1, A.2 and A.3 along with some properties of the algorithm. We state proof for $G(n, c/n)$ with $c \leq 2e$ and $G(n, r)$ with $r \leq 4$ simultaneously.

Consider a fixed node i of G, where $G = G(n, c/n), c \leq 2e$, or $G = G(n, r), r \leq 4$. Let $\varepsilon > 0$ be given. Let E_1 be the event that the local neighborhood of depth t of node i, $G_i(t)$ is *tree*. We will assume some odd $t = t(\varepsilon)$ or $t(\varepsilon) + 1$ as required per Lemma A.3. As part of the algorithm, the bonus values at node i, $B_i^t(\mathbf{0})$ can be computed recursively starting with $\mathbf{0}$ initial condition at the leaf nodes (at depth t) of its computation tree $T_i(t)$ as described before. Now, consider the bonus value $B_i^{t+1}(\mathbf{0})$ at the *next* step. Due to the update equation of our algorithm and computation tree structure, $T_i(t) \subset T_i(t+1)$, it is easy to see that $B_i^{t+1}(\mathbf{0})$ is equal to $B_i^t(\mathbf{L})$ for some initial condition $\mathbf{L}$, $\mathbf{0} \leq \mathbf{L} \leq \mathbf{W}$, for leaf nodes at depth t of $T_i(t)$. This is because irrespective of incoming messages, the message generated along any edge is always between $[0, W]$, where W is the node weight. Similarly, we can inductively argue that $B_i^s(\mathbf{0})$ is equal to $B_i^t(\mathbf{L}_s)$ for some $0 \leq \mathbf{L}_s \leq \mathbf{W}$ for all $s \geq t$. Therefore, using Lemma A.2 and its consequence (6), for any $s \geq t$,

$$B_i^t(\mathbf{0}) \;\; \leq \;\; B_i^s(\mathbf{0}) \;\leq\; B_i^t(\mathbf{W}). \tag{8}$$

Suppose, that E_1 is true. Then, $G_i(t) = T_i(t)$ and the W_i is independent of messages γ^t based on computation tree $T_i(t)$. Therefore, $B_i^t(\mathbf{0}) \neq 0$ and $B_i^t(\mathbf{W}) \neq 0$ with probability 1 due to W_i being drawn from a continuous distribution. Therefore, by definition of x_i^s based on $B_i^s(\mathbf{0})$ and (8), it follows that

under event $E_2 = \{B_i^t(\mathbf{0}) > 0\}$, the $x_i^s = 1$ for all $s \geq t$. Consider event $E_3 = \{B_i^t(\mathbf{W}) < 0\}$. Given E_1, $E_3 = \{B_i^t(\mathbf{W}) \leq 0\}$ since $B_i^t(\mathbf{W}) \neq 0$ with probability 1. Therefore, using similar reasoning we have that $x_i^s = 0$ for all $s \geq t$. From this discussion, it follows that for $s \geq t$

$$\Pr\left(x_i^s \text{ converges}\right) \;\geq\; \Pr((E_2 \cup E_3) \cap E_1) \;=\; \Pr_{G_i(t)}((E_2 \cup E_3) \cap E_1), \tag{9}$$

where in the last equality we introduce the sub-script $G_i(t)$ as the probability of event $(E_2 \cup E_3) \cap E_1$ primarily depends only on local neighborhood of depth t, $G_i(t)$. Here, by $\Pr_{G_i(t)}$ we mean the distribution induced by the random graph $G(n, c/n)$ or $G(n, r)$ and the random node weights on the local neighborhood of node i upto depth t. Inspired by Lemma A.1, consider the distribution induced by Poisson tree $\mathcal{T}(c, t)$ or regular tree $\mathbb{T}(r, t)$ (depending upon random graph model of G) and the independent random node weights – denote this by $\mathbb{Q}(\cdot)$.

Now, the sequence (dependent on number of nodes n in G) of distributions $\Pr_{G_i(t)}(\cdot)$ converges to $\mathbb{Q}(\cdot)$ as per Lemma A.1. This convergence is defined over appropriate topology of local weak convergence (see, [31] for example). Now, it can be checked that $(E_2 \cup E_3) \cap E_1$ is an open set. Therefore, by Portmanteau theorem it follows that

$$\liminf_{n\to\infty} \Pr_{G_i(t)}((E_2 \cup E_3) \cap E_1) \;\geq\; \mathbb{Q}((E_2 \cup E_3) \cap E_1). \tag{10}$$

Equivalently, for selection of $N(\varepsilon)$ large enough we have that when $n \geq N(\varepsilon)$

$$\Pr_{G_i(t)}((E_2 \cup E_3) \cap E_1) \;\geq\; \mathbb{Q}((E_2 \cup E_3) \cap E_1) - \varepsilon. \tag{11}$$

Therefore, in order to establish that x_i^s converges for $s \geq t$ with enough probability it is sufficient to show that

$$\mathbb{Q}((E_2 \cup E_3) \cap E_1) \geq 1 - \varepsilon.$$

Now consider the following.

$$\begin{aligned}
\mathbb{Q}((E_2 \cup E_3) \cap E_1) &\overset{(a)}{=} \mathbb{Q}(E_2 \cup E_3) \\
&= \mathbb{Q}(E_2) + \mathbb{Q}(E_3 \cap E_2^c) \\
&\overset{(b)}{=} \mathbb{Q}(E_2) + \mathbb{Q}(E_3) \\
&\overset{(c)}{=} \mathbb{Q}\left(B_i^t(\mathbf{0}) > 0\right) + \mathbb{Q}\left(B_i^t(\mathbf{W}) \leq 0\right) \\
&= 1 - \mathbb{Q}\left(B_i^t(\mathbf{0}) \leq 0\right) + \mathbb{Q}\left(B_i^t(\mathbf{W}) \leq 0\right) \\
&\geq 1 - \varepsilon,
\end{aligned} \tag{12}$$

where (a) follows from fact that under $\mathbb{Q}$ the E_1 is always true; (b) follows from – $E_3 = \{B_i^t(\mathbf{W}) < 0\} \subset \{B_i^t(\mathbf{0}) \leq 0\} = E_2^c$; (c) follows from definition and (12) follows from Lemma A.3 (for either $\mathcal{T}(c, t)$ or $\mathbb{T}(r, t)$). Thus, (9), (10) and (12) imply that x_i^s converges with probability at least $1 - 2\varepsilon$ for $n \geq N(\varepsilon)$ and $s \geq t = t(\varepsilon)$.

To prove the correctness of x_i^t upon convergence, note the following: by Theorem 3.1 and it follows that if x_i^t has converged to 0 or 1 then the corresponding LP optimum solution must have those assignment for node i. By Lemma 2.1 it follows that these are the assignment from the MWIS. Therefore, if $x_i(\gamma^s)$ converges (or in fact is equal to 0, or 1, for two subsequent time steps) then Theorem 3.1 states that $x_i(\gamma^s) = x_i^*$, the value of the unique LP optimum for i and Lemma 2.1 then implies that this estimate is actually correct for G. Therefore, we have proved that the converged values is the correct value. This completes the proof of Theorem 4.1 by re-selection $\varepsilon = \varepsilon/2$ and selection appropriate values for $N(\varepsilon)$ and $T(\varepsilon)$.

A.4 Proof of Theorem 4.2

We need to establish that (a) the stopping condition induces an independent set, $\hat{I}^t$ and (b) the $\hat{I}^t$ has high enough weight. To this end, recall that if $x_i^{t-1} = x_i^t = 1$ then by Theorem 3.1 it must be that the LP optimum assignment has $x_i^* = 1$ and by Lemma 2.1 it is indeed equal to the assignment as per the MWIS. Since $\hat{I}^t$ contains only such nodes, it follows that $\hat{I}^t \subset I^*$ and hence a valid independent set.

Now, by Theorem 4.1 it follows that for graph with $n \geq N(\varepsilon)$ and number of iterations $t \geq T(\varepsilon)$, at least $1-\varepsilon$ fraction of nodes find their right assignment with probability at least $1-\varepsilon$. Therefore, it follows that $|I^* \backslash \hat{I}^t| \leq \varepsilon n$ with probability at least $1-\varepsilon$. Now, consider the following *extremal* property of Exponential random variables.

Lemma A.4 *Consider n i.i.d. random variables $X_1, \ldots, X_n$ with Exponential distribution of mean 1. Let the ordered sequence be $X_{\pi(1)} \geq \cdots \geq X_{\pi(n)}$. Then, for any $\varepsilon \in (0, e^{-10})$*

$$\Pr\left(\frac{1}{n}\sum_{i=1}^{\varepsilon n} X_{\pi(i)} \geq 2\varepsilon(1+\ln(1/\varepsilon))\right) \leq \exp\left(-n\varepsilon\ln(1/\varepsilon)\right).$$

Proof: The proof follows by an application of Cramer's Theorem for Exponential random variables. Specifically, given N i.i.d. random variables $Y_1, \ldots, Y_N$ with mean 1 and Exponential distribution, for any $L \geq 10$ it follows that

$$\begin{aligned}\Pr\left(\frac{1}{N}\sum_{k=1}^{N} Y_k \geq L\right) &\leq \exp\left(-N(L-1-\log L)\right) \\ &\leq \exp\left(-NL/2\right). \end{aligned} \tag{13}$$

Therefore, by application of (13) it follows that for any collection of εn of $X_1, \ldots, X_n$, their summation is no larger than $n\varepsilon L$ with probability at least $1 - \exp(-n\varepsilon L/2)$ for $L \geq 10$.

Now, given n random variables there are $\binom{n}{\varepsilon n}$ distinct ways to select εn indices. Therefore, by an application of union bound, above discussion and Striling's approximation it follows that

$$\begin{aligned}\Pr\left(\frac{1}{n}\sum_{i=1}^{\varepsilon n} X_{\pi(i)} \geq \varepsilon L\right) &\leq \binom{n}{\varepsilon n}\exp\left(-\frac{n\varepsilon L}{2}\right) \\ &\leq \exp\left(n\varepsilon\ln(1/\varepsilon) + n\varepsilon - n\varepsilon L/2\right). \end{aligned} \tag{14}$$

Select $L = L(\varepsilon) = \max\{2(1+\ln(1/\varepsilon)), 10\}$. Then, we obtain that

$$\Pr\left(\frac{1}{n}\sum_{i=1}^{\varepsilon n} X_{\pi(i)} \geq \varepsilon L(\varepsilon)\right) \leq \exp\left(-n\varepsilon\ln(1/\varepsilon)\right). \tag{15}$$

Note that for $\varepsilon \leq e^{-10}$, $L(\varepsilon) = 2(1+\ln(1/\varepsilon)) \geq 10$. This completes the proof of Lemma A.4. ■

From Lemma A.4, it follows that for n large enough (say larger than $N_1(\varepsilon)$) the the net weight of nodes in $|I^* \backslash \hat{I}^t|$ is at most $\gamma(\varepsilon)n$ with probability at least $1-\varepsilon$ for $\gamma(\varepsilon) = O(\varepsilon\ln(1/\varepsilon))$ as $\varepsilon \to 0$. Clearly, $\gamma(\varepsilon) \to 0$ as $\varepsilon \to 0$. As established in [8], the weight of the maximum weight independent set is $\Theta(n)$ with

probability $1 - o(1)$ under the setup of our interest. It follows (using union bound) that for $N_1(\varepsilon)$ large enough, for $n \geq N_1(\varepsilon)$,

$$P\left(\frac{|W(\hat{I}^t) - W(\mathcal{I}^*)|}{W(\mathcal{I}^*)} \geq \delta(\varepsilon)\right) \quad \leq \quad 2\varepsilon, \tag{16}$$

with $\delta(\varepsilon) = O(\varepsilon \log(1/\varepsilon))$. This completes the proof of Theorem 4.2.