An alternative proof for the constructive Asymmetric Lovász Local Lemma*

Ioannis Giotis[1,2], Lefteris Kirousis[1,2], Kostas I. Psaromiligkos[1], and Dimitrios M. Thilikos[1,2,3]

[1]Department of Mathematics, National Kapodistrian University of Athens, Greece and
[2]Computer Technology Institute Press “Diophantus”, Patras, Greece
[3]AlGCo project-team, CNRS, LIRMM, France

We provide an alternative constructive proof of the Asymmetric Lovász Local Lemma. Our proof uses the classic algorithmic framework of Moser and the analysis introduced by Giotis, Kirousis, Psaromiligkos, and Thilikos in “On the algorithmic Lovász Local Lemma and acyclic edge coloring”, combined with the work of Bender and Richmond on the multivariable Lagrange Inversion formula.

1 Introduction

The Lovász Local Lemma (LLL) is a powerful combinatorial tool with several applications. As a result it appeared in 1975 in a paper by Erdős and Lovász [2]. Recently, a lot of work has been focused on constructive proofs of LLL. A major step in this direction has been the constructive proof of Moser [4] and Moser and Tardos [5]. In this paper we give a constructive proof of the so called *Asymmetric Lovász Local Lemma* that is based on the original approach from [4] and the ideas from [3].

The variable setting framework. We work in a framework known as the *variable setting*, which was used in [4]. Let $X_i, i = 1, \dots, n$ be mutually independent random variables on a common probability space, taking values in the sets $D_i, i = 1, \dots, n$, respectively. Let also $E_j, j = 1, \dots, m$ be a sequence of events, each depending on a sequence of the random variables X_i. We define the *scope* e^j of event E_j to be the minimal subset of variables such that one can determine whether E_j is satisfied or not knowing only their values, i.e., event E_j depends only on the values of the variables of e^j.). With every sequence of events $\mathcal{E} = E_1, \dots E_m$ we associate a unique graph $G_{\mathcal{E}}$ called the *dependency graph* of $\mathcal{E}$ which is defined as follows: $V(G_{\mathcal{E}}) = \{1, \dots, m\}$ and for every $i, j \in \{1, \dots, m\}$, we have $\{i, j\} \in E(G_{\mathcal{E}})$ if and only if $e^i \cap e^j \neq \emptyset$. For $j = 1, \dots, m$, we define the *neighborhood* of event E_j, denoted by N_j, to be the neighborhood of the vertex j in the dependency graph, i.e., $N_j = \{i \in \{1, \dots, m\} \mid \{i, j\} \in E(G_{\mathcal{E}})\}$ (observe that $i \in N_i$)

Asymmetric Lovász Local Lemma. The main goal of this work is to present an alternative algorithmic proof of the following theorem:

Theorem 1 (Asymmetric Lovász Local Lemma)**.** *If there exist* $\chi_1, \chi_2, \dots, \chi_m \in (0, 1)$ *such that* $\forall i \in \{1, \dots, m\}$ $Pr(E_i) \leq \chi_i \prod_{j \in N_i}(1 - \chi_j)$ *then* $Pr[\overline{E_1} \wedge \overline{E_2} \wedge \cdots \wedge \overline{E_m}] > 0$*, i.e., there exists an assignment to the variables* X_i *for which none of the events* E_i *hold.*

*Research co-financed by the European Union (European Social Fund ESF) and Greek national funds through the Operational Program “Education and Lifelong Learning” of the National Strategic Reference Framework (NSRF) - Research Funding Program: ARISTEIA II.

The original proof of Theorem 1, first presented in this form in [6], was non-constructive, but was given for arbitrary events, i.e. without the assumption that the events depended on independent random variables. Below, we will give an algorithmic proof Theorem 1 within the variable framework, where each event depends on a subset of the variables.

We analyze the algorithm presented in Figure 1, which can be derived directly from the one given by Moser in [4], and which is the same that we used in [3]. Our goal is to prove that the probability of Algorithm making many steps is sub-exponential to the number of steps.

```
Algorithm:

1. Sample the variables $X_i$ and let $\alpha$ be the resulting assignment of values to them.
2. While there exist an event that occurs under the current assignment, let $E_i$ be
   the least indexed such event
      2.1 Resample($E_i$)
3. Output current assignment $\alpha$.

Resample($E_i$):

1. Resample the variables in the scope $e^i$.
2. While some $E_j \in N_i$ occurs for the current assignment $\alpha$, let $E_j$ be the least indexed such event
      2.1 Resample($E_j$)
```

Figure 1: ***Randomized sampling*** algorithm

We call *phase*, the execution period within a *root* call of Resample (line 2.1 of Algorithm in Figure 1) and we will refer to calls of Resample from within another Resample execution as *resample* calls. In these terms, our task is to bound the probability that the Algorithm makes at least n Resample calls. To do this, we first show that the number of phases in any execution is bounded, which can be derived form the following lemma:

Lemma 2. *Consider an arbitrary call of* Resample(E_i). *Let $\mathcal{E}$ be the set of events that do not occur at the beginning of this call. Then, if the call terminates, events in $\mathcal{E}$ will also not occur at the end of the call.*

The above lemma implies that the number of phases in any execution is bounded as it states that the events that do not occur at the start of a Resample call also do not occur after its end. Thus, we have the following corollary:

Corollary 3. *There are at most m phases in any execution of* Algorithm.

Let us now examine the probability distribution of the variables after a resampling caused by a call of Resample(E_i).

Lemma 4 (Randomness lemma)**.** *Let α be a random assignment sampled from the probability distribution of the variables $X_1, \ldots, X_n$ and E_i an event. Let α' be the assignment obtained from α by resampling the variables in e^i if E_i occurs for α, and let α' be α otherwise. Then, conditional that E_i occurs under α, the distribution of α' is the random distribution of assignments sampled from all variables i.e. it is the same as the distribution of α. Therefore the probability that any event E occurs under α' is equal to the probability that E occurs under α.*

Now we provide two definitions identical to the ones given in [3] which are analyzed there.

Definition 5. *A sequence of events $\mathcal{E}_1, \ldots, \mathcal{E}_k$ is called a* witness sequence *if the first k* Resample *calls (recursive or root) of* Algorithm *are applied to $\mathcal{E}_1, \ldots, \mathcal{E}_k$, respectively.*

Now let $\hat{P}_n$ be the probability that ALGORITHM performs at least n RESAMPLE calls. It follows that $\hat{P}_n = \Pr[\text{ there is some witness sequence of length } n\ ]$.

Definition 6. *A sequence of events $\mathcal{E}_1, \ldots, \mathcal{E}_k$ is called a* valid sequence *if*
- *there is a rooted forest with at most m trees labeled with the events in the sequence so that the order of the events in the sequence coincides with the preorder of the labels of the forest (the same label may appear more than once),*
- *the label of a non-root node v in the forest is a neighbor of the label of the parent of v,*
- *the indices of the labels of the successive children of any node are strictly increasing; the same is true for the indices of the successive root labels of the forest, and*
- *$\mathcal{E}_i$ occurs for the assignment $\alpha_i, i = 1, \ldots, k$, where α_1 is obtained by sampling $X_1, \ldots, X_n$ and $\alpha_{i+1}, i = 1, \ldots, k-1$ is obtained by resampling the variables in e^i.*

If we now define: $P_n = \Pr[\text{ there is some valid sequence of length } n\ ]$ and assuming $P_0 = 1$, we have clearly that $\hat{P}_n \leq P_n$. With the purpose of bounding P_n, we define $Q_{n,i}$ to be the probability that there is some valid sequence starting from E_i and of length $\geq n$. Our target from now on is to upper bound the numbers $Q_{n,i}$.

Recurrence relations. We will need the following lemma (proof is omitted):

Lemma 7. *For every n and every $i \in \{1, \ldots, m\}$, assuming that $N_i = \{i_1, \ldots i_l\}$, the numbers $Q_{n,i}$ satisfy the following equation:*

$$Q_{n,i} = Pr(E_i)\left(\sum_{n_1+\ldots+n_l=n-1} Q_{n_1,i_1} Q_{n_2,i_2} \cdots Q_{n_l,i_l}\right) \tag{1}$$

Denote $Q_i(z) = \sum_{n=1}^{\infty} Q_{n,i} \cdot z^n$. Multiplying (1) by z^n we get

$$Q_{n,i} z^n = zPr(E_i)\left(\sum_{n_1+\ldots+n_l=n-1} Q_{n_1,i_1} z^{n_1} \cdot Q_{n_2,i_2} z^{n_2} \cdots Q_{n_l,i_l} z^{n_l}\right) \tag{2}$$

and we add over all n and to get $Q_i(z) = z \cdot Pr(E_i) \prod_{j \in N_i} \big(Q_j(z) + 1\big)$. Now we have obtained a system of equations ($\overline{Q} = (Q_1, Q_2, \ldots, Q_m)$): $Q_i(z) = z f_i(\overline{Q})$ where $f_i(\overline{Q}) = \chi_i \prod_{j \in N_i} (1 - \chi_j)(Q_j + 1)$. We will use Theorem 2 from [3]. To do this, we need to make a generalization. Instead of our functions in one variable, we will generalize in m variables. This will grant us a unique solution in a more general system, and since we know that our system also does have a solution it must be the same given that $x_1 = x_2 = ... = x_m$. So we consider the same system where in each equation z is substituted by t_i and Q_i's are generating functions on $t = (t_1, ..., t_m)$. Using directly the main result of Bender and Richmond in [1] (Theorem 2), what we need to show is that the following quantity (which will be $Q_s(\mathbf{t})$) is bounded by ρ^n where $\rho < 1$ and $\sum_{i=1}^{m} n_i = n$, probably ignoring polynomial factors involving n and m since they are asymptotically irrelevant (g is the projection function): $\frac{1}{\prod n_i}[\mathbf{x^{n-1}}] \sum_T \frac{d(g, f_1^{n_1}, ..., f_m^{n_m})}{dT}$. where the sum is over all rooted trees with edges directed towards 0. We observe that since the $\frac{1}{\prod n_i}$ factor is irrelevant, and that also the number of trees is a function of m, we just need to bound from above the term of the sum for an arbitrary tree. Let T be a tree. We need to upper bound the following: $[\mathbf{x^{n-1}}] \prod_{j \in V(T)} \left\{ \left(\prod_{(i,j) \in E(T)} \frac{d}{dx_i} \right) f_j(\mathbf{x}) \right\}$. Now, we substitute the values of f_i and we group corresponding to i, meaning that we group together the factors χ_i, $1 - \chi_i$, and $x_i + 1$. First, observe that if in T vertex 0 has a child (by child of a vertex we will mean another vertex from which there exists an edge directed towards it) other than s, then the product would be zero (since the derivative of $g(\mathbf{x}) = x_s$ would be zero) and the same holds if a vertex i has a child not in its neighborhood. So we will restrict our attention to the case that T satisfies these constraints.

All χ's remain as they were since they are constants. Notice that without the existence of the derivatives, the product would be: $\prod_{i=1}^m \left\{ \chi_i^{n_i} \cdot (1-\chi_i)^{\sum_{j\in N_i} n_j} \cdot (x_i+1)^{\sum_{j\in N_i} n_j} \right\}$. In the actual product, the exponent of $x_i + 1$ will be reduced by 1 for every $\frac{d}{dx_i}$. But T is a tree, so every node i has outdegree one (except from the i of the projection function but it will become obvious later that is safe to assume that this exponent is also reduced by 1). So every exponent is reduced exactly by 1. So, ignoring n_k's (which is safe as they are bounded by n^m), we need to bound: $[\mathbf{x^{n-1}}] \prod_{i=1}^m \left\{ \chi_i^{n_i} \cdot (1-\chi_i)^{\sum_{j\in N_i} n_j} \cdot (x_i+1)^{\sum_{j\in N_i} n_j - 1} \right\}$ But this, by binomial theorem, is: $\prod_{i=1}^m \left\{ \chi_i^{n_i} \cdot (1-\chi_i)^{\sum_{j\in N_i} n_j} \cdot \binom{\sum_{j\in N_i} n_j - 1}{n_i - 1} \right\}$. Obviously, by the identity $\binom{n}{k} = \frac{n}{k}\binom{n-1}{k-1}$ and by ignoring again non-exponential factors, this is:

$$\begin{aligned}
&= \prod_{i=1}^m \left\{ \chi_i^{n_i} \cdot (1-\chi_i)^{\sum_{j\in N_i} n_j} \cdot \binom{\sum_{j\in N_i} n_j}{n_i} \right\} \\
&= \prod_{i=1}^m (1-\chi_i)^{n_i} \cdot \chi_i^{n_i} \cdot (1-\chi_i)^{\sum_{j\in N_i} n_j - n_i} \cdot \binom{\sum_{j\in N_i} n_j}{n_i} \\
&< \prod_{i=1}^m (1-\chi_i)^{n_i}
\end{aligned}$$

The last inequality is derived by expanding $1 = \big(\chi_i + (1-\chi_i)\big)^{\sum_{j\in N_i} n_j}$ and noticing that the term we bounded is just the n_i^{th} term of the sum (which is a sum of nonnegative terms since $\chi_i \in (0,1)$, so any of its terms is less than the sum itself). Now, let $M = \max\{(1-\chi_i)\}$ and observe that $M < 1$, since $1 - \chi_i < 1$ for every i. Finally, we have: $\prod_{i=1}^m (1-\chi_i)^{n_i} \leq \prod_{i=1}^m M^{n_i} = M^{\sum n_i} = M^n$.

To sum up, we proved that the probability of Algorithm making more than n steps is sub-exponential in n, so we proved Theorem 1.

References

[1] Edward A. Bender and L. Bruce Richmond. A multivariate lagrange inversion formula for asymptotic calculations. *Electr. J. Comb.*, 5, 1998.

[2] Paul Erdős and László Lovász. Problems and results on 3-chromatic hypergraphs and some related questions. *Infinite and finite sets*, 10:609–627, 1975.

[3] Ioannis Giotis, Lefteris M. Kirousis, Kostas I. Psaromiligkos, and Dimitrios M. Thilikos. On the algorithmic lovász local lemma and acyclic edge coloring. In *Proceedings of the Twelfth Workshop on Analytic Algorithmics and Combinatorics, ANALCO 2015, San Diego, CA, USA, January 4, 2015*, pages 16–25, 2015.

[4] Robin A Moser. A constructive proof of the Lovász Local Lemma. In *Proceedings of the 41st annual ACM Symposium on Theory of Computing*, pages 343–350. ACM, 2009.

[5] Robin A Moser and Gábor Tardos. A constructive proof of the general Lovász Local Lemma. *Journal of the ACM (JACM)*, 57(2):11, 2010.

[6] Joel Spencer. Asymptotic lower bounds for Ramsey functions. *Discrete Mathematics*, 20(0):69 – 76, 1977.