RC24942 (W1002-002) February 1, 2010
Mathematics

IBM Research Report

Planar Three-Index Assignment Problem via Dependent Contention Resolution

Dmitriy Katz-Rogozhnikov, Maxim Sviridenko
IBM Research Division
Thomas J. Watson Research Center
P.O. Box 218
Yorktown Heights, NY 10598

Planar Three-Index Assignment Problem via Dependent Contention Resolution

Dmitriy Katz-Rogozhnikov* Maxim Sviridenko†

Abstract

In this paper we design an approximation algorithm for the planar three-dimensional assignment problem with performance guarantee 0.669. The algorithm is based on a novel rounding technique of the linear programming relaxation (the dependent contention resolution) that might be interesting in its own right and applied to other optimization problems.

1 Introduction

In the three-dimensional (or three-index) assignment problem we are given three index sets V_1, V_2 and V_3 of size n each. We are also given a three dimensional array $W = (w_{ijk} | (i,j,k) \in V_1 \times V_2 \times V_3)$. The goal is to choose a set of triples of maximum weight satisfying certain feasibility criteria. The two most popular variants are the axial three-dimensional assignment and the planar three-dimensional assignment problems [21, 22]. In the axial three-dimensional assignment problem we must choose a set of at most n triples such that each element of $V_1 \cup V_2 \cup V_3$ is in at most one triple. In the planar three-dimensional assignment problem we must choose a set of at most n^2 triples such that each element of $(V_1 \times V_2) \cup (V_2 \times V_3) \cup (V_1 \times V_3)$ is in at most one triple.

In this paper we study the planar three-dimensional assignment problem that can be naturally formulated as the following integer programming problem:

$$\max \sum_{i \in V_1, j \in V_2, k \in V_3} w_{ijk} x_{ijk}, \tag{1}$$

$$\sum_{i \in V_1} x_{ijk} \le 1, \qquad j \in V_2, k \in V_3, \tag{2}$$

$$\sum_{j \in V_2} x_{ijk} \le 1, \qquad i \in V_1, k \in V_3, \tag{3}$$

$$\sum_{k \in V_3} x_{ijk} \le 1, \qquad i \in V_1, j \in V_2, \tag{4}$$

$$x_{ijk} \in \{0, 1\}, \qquad i \in V_1, j \in V_2, k \in V_3. \tag{5}$$

We consider the linear programming relaxation where we replace constraints (5) with the constraints

$$x_{ijk} \ge 0, \qquad i \in V_1, j \in V_2, k \in V_3. \tag{6}$$

Problem Motivation and Applications: The maximum planar three-index assignment problem is a well-known optimization problem [6, 7, 21, 22]. It models some natural real-life problems, e.g. timetabling [17], practical rostering problem [14], sattelite launching [3].

*IBM T.J. Watson Research Center, dkatzrog@us.ibm.com
†IBM T.J. Watson Research Center, sviri@us.ibm.com

The planar three-dimensional assignment problem also naturally models the so-called partial Latin Square Extension (LSE) Problem which has applications in conflict-free wavelength routing in wide-area optical networks [4], statistical designs and error-correcting codes [8, 9]. In the LSE problem, given a partial Latin Square, i.e. a square matrix such that some entries are colored by $\{1, \ldots, n\}$ such that there are no color repetitions in each row and column, the goal is to complete the partial coloring in a feasible way. Such a completion is not always possible, moreover the problem of deciding if such a completion exists is NP-complete. The natural optimization variant is to color as many entries as possible in a feasible way. To model LSE using (1)-(5) one needs to identify one set of indices (say V_3) with the set of colors and the other two sets of indices with the set of indices in the partial Latin Square. We define $w_{ijk} = 1$ if the entry (i, j) in Latin Square is not defined and $w_{ijk} = 0$, otherwise.

Another popular variant of the problem is when constraints (2)-(4) are replaced with equalities. While most applications are shared for both variants of the problem, they are not equivalent. The simple example is when $|V_1| = |V_2| = |V_3| = 2$, $w_{111} = w_{222} = 1$ and all other weights in the objective function are zero. For this simple example the problem with constraints (2)-(4) has an optimal solution of value two but the problem with equalities has optimal value of one. In this paper we consider the variant with constraints (2)-(4) only.

Previous Work: The maximum planar three-index assignment problem is known to be NP-hard [13] and even APX-hard [16]. There are exact branch and bound algorithms and tabu search type heuristics developed for the problem with equality constraints, see survey articles and books [6, 7, 21, 22]. The structure of the problem polyhedron was studied in [1, 10, 20].

The first approximation algorithm was developed for the LSE problem in [19]. This algorithm had a performance guarantee of $1/2$. That was improved to $1 - e^{-1}$ in [15]. Note that the algorithm in [15] was general enough to handle arbitrary linear objective function and as a by-product their paper implies an $(1 - e^{-1})$-approximation algorithm for the maximum planar three-index assignment problem (1)-(5). It was noted in [16] that the LSE problem can be actually represented as an instance of a maximum axial three-index assignment problem (or 3-dimensional matching) and therefore applying the best known algorithm for that problem [18] one can get a $2/3 - \varepsilon$-approximation algorithm for any $\varepsilon > 0$. The same argument also holds for the more general problem (1)-(5) with $w_{ijk} \in \{0, 1\}$. It should be noted that for the set packing type of problems there is a large difference between problems with general nonnegative weights and problems with $\{0, 1\}$-weights, e.g. the algorithm from [18] gives a $(2/3 - \varepsilon)-$approximation for the axial three-dimensional assignment problem while the best known algorithm for the problem with general weights has performance guarantee 2 [2, 5]. Analogously, while the reduction from [16] implies that the maximum planar three-index assignment problem (1)-(5) with $\{0, 1\}$-weights has an algorithm with performance guarantee $(2/3 - \varepsilon)$, the best known algorithm for the general problem has performance guarantee $1 - e^{-1}$ [15].

Contention Resolution: Randomized rounding of linear programming relaxations is a standard way to develop approximation algorithms. Quite often a natural randomized rounding leads to an infeasible solution. For example, if we have n players competing for one item and choosing that item independently with probability p_i for $i = 1, \ldots, n$ such that $\sum_{i=1}^{n} p_i = 1$ then it might happen that this item will be allocated to few players at the same time. There are few ways to resolve such conflicts [13, 12]. For example it is possible to resolve conflicts at random in such a way that each player obtains items with probability $(1 - \prod_{i=1}^{n}(1 - p_i))p_i$ [12] which leads to numerous applications in the design and analysis of randomized rounding type of algorithms.

In all applications of the contention resolution published so far the key element was that conflicts are resolved independently for each violated constraint (item). While such an approach greatly simplifies the analysis it might lead to a worse rounding scheme. In this paper we demonstrate that resolving conflicts in a coordinated way leads to an algorithm with better performance guarantee than the independent contention resolution. We believe that such an approach will appear to be useful for other randomized rounding algorithms.

Our Results: Our main result is a randomized polynomial time approximation algorithm for the planar three-dimensional assignment problem with performance guarantee lower bounded by 0.669 which is an improvement upon the best known performance guarantee of $1 - e^{-1}$ [15]. Actually, our performance guarantee even beats the performance guarantee of the local search algorithm for the unweighted case of the problem. The main tool used in the design of the algorithm and the proof of its performance guarantee is a dependent contention resolution scheme.

The rest of the paper is organized as follows. In the Section 2 we give the basic algorithm which our improvement is based on. In the Section 3 we outline the intuition and the reasoning behind our dependent contention resolution scheme, which improves the Basic algorithm. We give the formal description of our modification in Section 4 and then prove several basic Lemmas about it. We give the rest of the proof of the performance in Sections 5 and 6, and conclude with Section 7.

2 Basic Algorithm

Our algorithm starts by solving the linear programming relaxation (1)-(4),(6) of the planar three-dimensional assignment problem. Let $(y_{ijk}), i \in V_1, j \in V_2, k \in V_3$ be an optimal solution of this relaxation. The fractional solution $(y_{ijk}), i \in V_1, j \in V_2, k \in V_3$ naturally corresponds to $|V_1|$ fractional solutions of the bipartite matching problem. More precisely, for each index $i \in V_1$ we consider the complete bipartite graph $K_{|V_2|,|V_3|}$. Then the matrix $Y_i = (y_{ijk}), j \in V_2, k \in V_3$ defines a feasible fractional solution of the bipartite matching problem in the complete bipartite graph $K_{|V_2|,|V_3|}$ (the variable y_{ijk} corresponds to the edge $(j,k) \in K_{|V_2|,|V_3|}$) and therefore it can be represented as a convex combination of partial permutation matrices or in other words the fractional matching corresponding to Y_i can be represented as a convex combination of integral matchings in $K_{|V_2|,|V_3|}$. Let M_{it} for $t = 1, \ldots, m_i$ be the set of matchings and λ_{it} be the set of coefficients in that convex combination. Obviously, $m_i \le |V_2 \times V_3|$, $\sum_{t=1}^{m_i} \lambda_{it} = 1$ and $\sum_{t|(j,k)\in M_{it}} \lambda_{it} = y_{ijk}$. Let $\Pi(M_{it})$ be a partial permutation matrix corresponding to the matching M_{it}. Then $Y_i = \sum_{t=1}^{m_i} \lambda_{it}\Pi(M_{it})$.

Our next step is a straightforward randomized rounding. For each index $i \in V_1$ choose one index $t = 1, \ldots, m_i$ (or corresponding partial permutation matrix or a matching) at random using probability distribution defined by the coefficients (λ_{it}) in the convex combination. This randomized rounding defines a solution of the planar three-dimensional assignment problem such that constraints (3) and (4) are satisfied. Unfortunately, the constraints (2) might be violated for some pairs $j \in V_2, k \in V_3$. Let $(\bar{y}_{ijk}), i \in V_1, j \in V_2, k \in V_3$ be an infeasible integral solution obtained by the above randomized rounding.

We will call a set of elements $C_{jk} = \{(i,j,k)|i \in V_1\} \subseteq V_1 \times V_2 \times V_3$ a column corresponding to the indices $j \in V_2, k \in V_3$. Note that each constraint of type (2) corresponds to one column in the array $V_1 \times V_2 \times V_3$. Our randomized rounding defines an infeasible solution $\bar{y}_{ijk}$ such that there might be some columns C_{jk} with few triples $(i,j,k) \in C_{jk}$ such that $\bar{y}_{ijk} = 1$. Our next step is to apply some contention resolution scheme, i.e. for each column C_{jk} such that there are $l_{jk} \ge 1$ indices $i \in V_1$ with $\bar{y}_{ijk} = 1$ we choose at most one index $i' \in V_1$ and define $\tilde{y}_{i'jk} = \bar{y}_{i'jk} = 1$ and all other variables in the same column are defined to be zero, i.e. $\tilde{y}_{ijk} = 0$ for $i \ne i'$. In the future sections we will refer to the infeasible integral solution $\bar{y}$ before the contention resolution and the feasible integral solution $\tilde{y}$ after the contention resolution.

There are few ways to define a contention resolution scheme. The simplest way is to choose an index $i \in V_1$ with $\bar{y}_{ijk} = 1$ at random with probability $1/l_{jk}$ for each column C_{jk} independently. After that all variables except chosen ones are defined to be zero and $\tilde{y}_{ijk} = 1$ for the chosen triples. This scheme was proposed and analyzed in [15]. It leads to an $(1 - e^{-1})$-approximation algorithm for the unweighted variant of our problem, i.e. $w_{ijk} \in \{0, 1\}$. A more sophisticated schemes were suggested in [13, 12] for different applications. Unfortunately, doing contention resolution independently at random for each column does not seem to lead to much better performance guarantees even for more sophisticated schemes. In the next section we will describe a dependent contention resolution scheme that resolves conflicts for different columns in a coordinated way. Now, assuming that we have a contention resolution scheme with certain properties we finish the description of the basic algorithm.

Lemma 1 *If the contention resolution scheme has the property that $Pr(\tilde{y}_{ijk} = 1) \geq (1 - e^{-1})y_{ijk}$ than the expected value of the solution obtained by the above algorithm is at least $1 - e^{-1}$ times the optimal value of the linear programming relaxation (1)-(4),(6).*

Proof. The proof is by the linearity of the expectation and the fact that $\tilde{y}$ is a feasible solution of the planar three-dimensional assignment problem. □

We say that a triple (i, j, k) conflicts with a triple (p, q, r) if at least two out of three equalities $i = p$, $j = k$, $k = r$ are satisfied, i.e. variables corresponding to those triples cannot both have value one due to constraints (2)-(4). The proof of the next lemma describes a way to boost the performance guarantee of an algorithm if a contention resolution scheme has additional properties.

Lemma 2 *Let U_{ijk} be an event corresponding to the triple $(i, j, k) \in V_1 \times V_2 \times V_3$ such that in the solution $\tilde{y}$ obtained after contention resolution we have $\tilde{y}_{i'j'k'} = 0$ for all triples (i', j', k') that conflict with the triple (i, j, k). If for our rounding algorithm combined with contention resolution scheme $Pr(U_{ijk}) \geq Q$ then there exists an approximation algorithm with performance guarantee $\frac{1-e^{-1}}{1-Q}$.*

Proof. First we describe our algorithm and then we prove its performance guarantee. Our algorithm proceeds in phases. In the beginning of each phase we already have a partial solution $\tilde{y}$ (initially, $\tilde{y} = 0$). Let $U \subseteq V_1 \times V_2 \times V_3$ be the subset of triples (i, j, k) such that we have $\tilde{y}_{i'j'k'} = 0$ for all triples (i', j', k') conflicting with (i, j, k), i.e. the event U_{ijk} happened on the previous phase of the algorithm.

Since for each triple $(i, j, k) \in U$ there are no conflicts with the existing solution we basically run our rounding algorithm again restricted to the set U. More specifically, we restrict our fractional solution to triples from U, we also restrict the matchings M_{it} to the edges corresponding to triples from U. The new phase of the algorithm defines a preliminary assignment of variables corresponding to U by choosing one matching at random for each $i \in V_1$ and after applying the contention resolution on the elements of U belonging to the same column we obtain a feasible assignment. Again after this phase there are some elements from the set U that were not included in the assignment moreover for some of them there are no conflicts with the existing assignments. We run our rounding again for this set of elements and we repeat until no such elements left.

On each phase the probability that the triple $(i, j, k) \in U$ is chosen $\geq (1 - e^{-1})y_{ijk}$ by Lemma 1. Moreover, by the conditions of this Lemma the probability that an element belongs to the set U after s iterations is at least Q^s. Therefore, the probability that $\tilde{y}_{ijk} = 1$ in the final solution is at least $(1 - e^{-1}) \sum_{s=1}^{\infty} Q^s y_{ijk} = (1 - e^{-1})y_{ijk}/(1 - Q)$. □

3 Intuition Behind Dependent Contention Resolution

On the limitations of the independent contention resolution. As we showed in Lemma 2, the quality of our solution depends on Q where $Pr(U_{ijk}) \geq Q$. Let U_i be event that for any i' we have $\tilde{y}_{i'jk} = 0$, i.e. no triple conflicting with (i, j, k) is present in the same column C_{jk}. Likewise, let U_j and U_k be events that $\tilde{y}_{ij'k} = 0$ for any j' and $\tilde{y}_{ijk'} = 0$ for any k' in other words there is no conflict with (i, j, k) in "lengthwise" row and "depthwise" row correspondingly. Clearly, $U_{ijk} = U_i \wedge U_j \wedge U_k$. It is easy to design contention resolution for which $Pr(U_i) = Pr(U_j) = Pr(U_k) \approx e^{-1}$, e.g. if we apply Feige and Vondrak [12] contention resolution independently for each column. If all variables y are small then there is no positive correlation between the three events, thus $Q \approx e^{-3} < 0.05$. Actually, the worst case happens when some variables y are large and in this case the value of Q is even smaller, thus it is not possible to obtain a significant improvement over $(1-e^{-1})$ approximation guarantee using the independent contention resolution scheme.

Our goal with dependent contention resolution is to increase Q by establishing a strong positive correlation between U_j and U_k, so if there is no conflict in "lengthwise" row with (i,j,k) (i.e. U_j occurs) then there is likely no conflict in "depthwise" row (i.e. U_k occurs). This would have the effect of increasing $Pr(U_{ijk})$, up to e^{-2} if a perfect correlation could be established (and negative correlations avoided). Of course, it is unrealistic to expect the perfect correlation between events U_j and U_k. We use the following technique to establish the positive correlation:

The key technique of dependent contention resolution. For each $i \in V_1$, choose a number u_i at random from $U[0,1]$ (uniform distribution). Then, in contention resolution stage, for each column have triples (i,j,k) with smaller u_i be more likely to "win", i.e. have $\tilde{y}_{ijk}=1$, and with larger u_i more likely to lose. This has the effect of strongly correlating U_j and U_k, since for small u_i both events are unlikely to happen, and for large u_i both are more likely to happen.

Making it work. It would be easiest to just have the element with the smallest u_i in a column win contention resolution. But while this works if all $y's$ are vanishingly small, it might violate the assumption of Lemma 1 when some $y's$ are large. So instead, we must adjust our tecnique to satisfy the assumption of Lemma 1 ($Pr(\tilde{y}_{ijk}=1) \geq (1-e^{-1})y_{ijk}$). To do this we use the following technical trick, instead of using u_i's direcly, we use $s_i(u_i, y_{ijk})$'s, and have the element with the smallest $s_i(u_i, y_{ijk})$ in a column win. We choose the (monotonically increasing in u_i) function $s_i()$ in such a way as to have contention resolution scheme satisfy assumption of Lemma 1 with equality.

Other technical tricks. We noticed that in the worst case for our analysis the events U_i and $U_j \wedge U_k$ are positively correlated. It motivated us to introduce parameter a to partly counteract the effect of such positive correlation. Part of the reason behind us introducing a, is to obtain an algorithm with performance guarantee $> 2/3$. While with $a=0$ we would still have an improvement over the best known bound for the weighted case $(1-e^{-1})$, $a=.34$ results in a (weighted case) approximation guarantee better than the previously best known bound for the usually easier unweighted case (2/3), which we felt was worth the extra effort.

4 Dependent Contention Resolution Scheme

Let $a=0.34$ be a fixed parameter. For each $i \in V_1$ we choose a number, $u_i \in [0,1]$ uniformly at random. Let $U_{jk} \subseteq C_{jk}$ be the set of triples in column C_{jk} that participate in the randomized rounding on the current phase of our algorithm (recall that this is a set of triples that do not have conflicts with the triples chosen on the previous phases). For each column C_{jk} we define a dummy element $(0,j,k)$. We define $y_{0jk} = 1 - \sum_{(i,j,k)\in U_{jk}} y_{ijk}$ and $\bar{y}_{0jk}=0$. We add this dummy element to the set U_{jk} to guarantee that $\sum_{(i,j,k)\in U_{jk}} y_{ijk} = 1$. If the dummy element $(0,j,k)$ wins the contention resolution then we define $\tilde{y}_{ijk}=0$ for all $(i,j,k)\in U_{jk}$. For each triple $(i,j,k)\in U_{jk}$ with $y_{ijk}>0$ and $\bar{y}_{ijk}=1$, i.e. for each triple chosen on the current phase of the algorithm we define

$$s_{ijk} = s_{ijk}(u_i) = \frac{-\ln(1-u_i y_{ijk})}{y_{ijk} - a y_{ijk}^2}.$$

Note that $s_{ijk}(u_i)$ is monotonically increasing for $u_i \in [0,1]$.

For each triple (i,j,k), let R_{ijk} be an exponentially distributed random variable with intensity $y_{ijk} - a y_{ijk}^2$, i.e. $Pr(R_{ijk} > r) = e^{-(y_{ijk} - a y_{ijk}^2) r}$. For a dummy element $(0,j,k) \in U_{jk}$ we define $s_{0jk} = R_{0jk}$. For each $y_{ijk}>0$, $\bar{y}_{ijk}=0$ and $i \geq 1$ we define

$$s_{ijk} = R_{ijk} + \frac{-\ln(1-y_{ijk})}{y_{ijk} - a y_{ijk}^2}.$$

Let $b_{jk} = a\sum_{(i,j,k)\in U_{jk}} y_{ijk}^2$ (including the dummy element $(0,j,k)$). For each triple $(i,j,k) \in V_1 \times V_2 \times V_3$, let

$$z_{ijk} = \frac{-\ln\left(1 - \frac{(1-b_{jk})(1-e^{-1})}{1-ay_{ijk}}\right)}{1-b_{jk}}.$$

Contention resolution. Let (i',j,k) be a triple such that $s_{i'jk} = \min_{i\in U_{jk}} s_{ijk}$. The triple (i',j,k) is our candidate to win the contention resolution for column C_{jk}. If $s_{i'jk} \le z_{i'jk}$ and $\bar{y}_{i'jk} = 1$ then we define $\tilde{y}_{i'jk} = 1$ that is we declare the triple (i',j,k) to be the winner of the contention resolution scheme. Of course in this case we must define $\tilde{y}_{ijk} = 0$ for all other elements $i \in U_{jk}$. Finally if there is no such triple (i',j,k) then we define $\tilde{y}_{ijk} = 0$ for all triples $(i,j,k) \in U_{jk}$.

Let $u_{ijk}(s) = (1 - e^{-(y_{ijk}-ay_{ijk}^2)s})/y_{ijk}$ be the inverse function of $s_{ijk}(u)$. We now prove some properties of our contention resolution scheme.

Lemma 3 *The random variable s_{ijk} has an exponential distribution with intensity $y_{ijk} - ay_{ijk}^2$, i.e. $P(s_{ijk} > s) = e^{-s(y_{ijk}-ay_{ijk}^2)}$*

Proof. This statement is true for the dummy triple $(0,j,k)$ by the definition. Fix any nondummy triple $(i,j,k) \in U_{jk}$, let $s' = \frac{-\ln(1-y_{ijk})}{y_{ijk}-ay_{ijk}^2}$. We consider two cases. If $s \ge s'$ then $Pr(s_{ijk} \ge s') = Pr(R_{ijk} \ge s - s') \cdot (1 - y_{ijk})$ since in the case when $\bar{y}_{ijk} = 1$ we are guaranteed that $s_{ijk} \le s'$. Therefore,

$$Pr(s_{ijk} \ge s') = e^{-(y_{ijk}-ay_{ijk}^2)(s-s')} \cdot (1 - y_{ijk}) =$$
$$e^{-(y_{ijk}-ay_{ijk}^2)s} \cdot e^{(y_{ijk}-ay_{ijk}^2)s'} \cdot (1 - y_{ijk}) = e^{-(y_{ijk}-ay_{ijk}^2)s},$$

where the last equality follows from the definition of s'.

If $s < s'$ then using the definition of the inverse function $u_{ijk}(s)$ we obtain

$$Pr(s_{ijk} \ge s) = y_{ijk} \cdot Pr(u_i > u_{ijk}(s)) + (1 - y_{ijk}) =$$
$$y_{ijk}(1 - u_{ijk}(s)) + (1 - y_{ijk}) = 1 - y_{ijk}u_{ijk}(s) = e^{-(y_{ijk}-ay_{ijk}^2)s}.$$

□

To estimate the probability that a triple is chosen during one phase of our algorithm we will need the following property of the exponential distributions:

Lemma 4 *We are given a collection of independent exponential random variables $X_1, \ldots, X_n$ with rate parameters (or intensities) $\lambda_1, \ldots, \lambda_n$, i.e. $Pr(X_i > s) = e^{-\lambda_i s}$. Let $Z \ge 0$ be a real number then for any index $i' = 1, \ldots, n$:*

$$Pr(X_{i'} = \min_{i=1,\ldots,n} X_i | \min_{i=1,\ldots,n} X_i < Z) = \frac{\lambda_{i'}}{\sum_{i=1}^n \lambda_i}.$$

Proof. Using the fact that the random variable $\min_{i=1,\ldots,n} X_i$ is an exponential random variable with the intensity $\sum_{i=1}^n \lambda_i$, we estimate $Pr(X_{i'} = \min_{i=1,\ldots,n} X_i | \min_{i=1,\ldots,n} X_i \in [Y, Y+\delta])$ in the limit when $\delta \to 0$. We have

$$\begin{aligned}
\lim_{\delta\to 0} Pr(X_{i'} = \min_{i=1,\ldots,n} X_i | \min_{i=1,\ldots,n} X_i \in [Y, Y+\delta]) &= \\
\lim_{\delta\to 0} \frac{Pr(X_{i'} = \min_{i=1,\ldots,n} X_i \bigwedge \min_{i=1,\ldots,n} X_i \in [Y, Y+\delta])}{Pr(\min_{i=1,\ldots,n} X_i \in [Y, Y+\delta])} &= \\
\lim_{\delta\to 0} \frac{Pr(X_{i'} \in [Y, Y+\delta] \bigwedge (X_i > Y \text{ for } i \ne i'))}{Pr(\min_{i=1,\ldots,n} X_i \in [Y, Y+\delta])} &= \\
\lim_{\delta\to 0} \frac{\lambda_{i'} e^{-\lambda_{i'} Y}\delta \cdot e^{-\sum_{t\ne i'} \lambda_t Y}}{\sum_{i=1}^n \lambda_i \delta \cdot e^{-\sum_{i=1}^n \lambda_t Y}} &= \frac{\lambda_{i'}}{\sum_{i=1}^n \lambda_i}.
\end{aligned}$$

Note that the second and third equalities only hold in the limit when $\delta \to 0$. Since $E(B|A) = \sum_{t=1}^{k} E(B|A_i)Pr(A_i)$ where A and B are arbitrary events and $A_1, \ldots, A_k$ is an arbitrary partition of A the Lemma follows by deconditioning. $\square$

To prove the main property of our contention resolution scheme we will need the following technical Lemma.

Lemma 5 *The inequality* $z_{ijk} < s_{ijk}(1)$ *holds for any triple* $(i, j, k) \in U_{jk}, j \in V_2, k \in V_3$.

Proof. Basically, the statement of the Lemma claims the following inequality

$$\frac{-\ln\left(1 - \frac{(1-b_{jk})(1-e^{-1})}{1-ay_{ijk}}\right)}{1-b_{jk}} < \frac{-\ln(1-y_{ijk})}{y_{ijk} - ay_{ijk}^2} \tag{7}$$

where $b_{jk} = a \sum_{(s,j,k) \in U_{jk}} y_{sjk}^2$.

Let $A = \frac{1-e^{-1}}{1-ay_{ijk}}$ and $X = 1 - b_{jk}$. First we show that that the left hand side of (7) is monotonically decreasing function of b_{jk}. Indeed,

$$\left(\frac{-\ln(1 - A \cdot X)}{X}\right)' = \frac{\frac{A \cdot X}{1 - A \cdot X} + \ln(1 - A \cdot X)}{X^2} \geq 0.$$

Combined with the fact that $b_{jk} \geq ay_{ijk}^2$ it is enough to prove the inequality

$$F(y_{ijk}) = \frac{-\ln(1-y_{ijk})}{y_{ijk} - ay_{ijk}^2} - \frac{-\ln\left(1 - \frac{(1-ay_{ijk}^2)(1-e^{-1})}{1-ay_{ijk}}\right)}{1 - ay_{ijk}^2} > 0.$$

The function $F(y_{ijk})$ is monotonically increasing on the interval $(0, 1]$ and is minimized in the limit when $y_{ijk} \to 0$ and $\lim_{y \to 0} F(y) = 0$, this fact can be checked numerically in Mathematica. We leave the analytical proof of this fact to the next version of the paper. $\square$

Lemma 6 *The probability that a triple* (i, j, k) *is chosen during one phase of our algorithm is exactly* $(1 - e^{-1})y_{ijk}$.

Proof. By the construction of our algorithm the triple (i', j, k) is chosen if $\bar{y}_{i'jk} = 1$, $s_{i'jk} = \min_{i \in U_{jk}} s_{ijk}$ and $s_{i'jk} \leq z_{i'jk}$. By the Lemma 5, $z_{i'jk} < s_{i'jk}(1)$. Therefore, if $s_{i'jk} \leq z_{i'jk}$ then $\bar{y}_{i'jk} = 1$ since for all triples (i, j, k) with $\bar{y}_{ijk} = 0$ we defined $s_{ijk} = R_{ijk} + s_{ijk}(1)$.

By the Lemma 3 each random variable s_{ijk} has exponential distribution. Therefore, the minimum of random variables $\min_{i \in U_{jk}} s_{ijk}$ is distributed as an exponential distribution with the intensity equal to the sum of the intensities of variables participating in the minimum ($\sum_{i \in U_{jk}} (y_{ijk} - ay_{ijk}^2) = 1 - b_{jk}$), i.e.

$$Pr(s_{i'jk} \leq z_{i'jk}) = 1 - Pr(s_{i'jk} > z_{i'jk}) = 1 - e^{-z_{i'jk}(1-b_{jk})} = \frac{(1-b_{jk})(1-e^{-1})}{1 - ay_{i'jk}}, \tag{8}$$

where the last equality holds by the definition of $z_{i'jk}$. Using Lemma 4 we obtain

$$Pr(s_{i'jk} = \min_{i \in U_{jk}} s_{ijk} | \min_{i \in U_{jk}} s_{ijk} \leq z_{i'jk}) = \frac{y_{i'jk} - ay_{i'jk}^2}{1 - b_{jk}}. \tag{9}$$

Combining (8) and (9) we obtain the statement of the Lemma. $\square$

Corollary 1 *The probability that a triple (i,j,k) is chosen during one phase of our algorithm conditioned on the fact that it was chosen during the randomized rounding of that phase, i.e. $\bar{y}_{ijk} = 1$, is exactly* $1 - e^{-1}$.

Proof. Follows immediately from the Lemma 6, using the facts that $Pr(A|B) = Pr(A \wedge B)/Pr(B)$ and that any element chosen during one phase must have been chosen during the randomized rounding of that phase. $\square$

Lemma 7 *Each column C_{jk} contains no elements after conflict resolution with probability is at least* e^{-1}.

Proof. Our algorithm either chooses exactly one triple or none. The Lemma follows immediately from the Lemma 6 and the fact that $\sum_{i \in U_{jk}} y_{ijk} = 1$ (including the dummy element). $\square$

In the rest of the paper we will show that for our contention resolution scheme $Q \approx 0.0555$ and therefore the performance guarantee of our algorithm is at least $(1 - e^{-1})/0.9445 \approx 0.6692$ by Lemma 2. We also would like to notice that the key technique in achieving this performance guarantee is the dependent contention resolution. It is possible to show that the independent contention resolution scheme leads to an approximation algorithm with performance guarantee $< 2/3$ and therefore does not improve the best known algorithm for the unweighted case.

5 Estimating Q

What we need to show is a lower bound on the value Q used in the proof of the Lemma 2. We consider one phase of our algorithm. Recall that U_{ijk} is an event corresponding to the triple $(i,j,k) \in V_1 \times V_2 \times V_3$ during that phase such that in the solution $\tilde{y}$ obtained after the contention resolution we have $\tilde{y}_{i'j'k'} = 0$ for all triples (i',j',k') conflicting with triple (i,j,k). Let N_{ijk} be an event that $\tilde{y}_{ij'k'} = 0$ for any triple (i,j',k') with either $j' = j$ or $k' = k$ (but not both), i.e. there are no conflicting triples with the triple (i,j,k) that share the same first index i (but the event $\tilde{y}_{ijk} = 0$ is not included in N_{ijk}). Finally, let B_{jk} be an event that $\tilde{y}_{i'jk} = 0$ for any triple (i',j,k), i.e. the column C_{jk} is empty after the current iteration. Obviously, $U_{ijk} = B_{jk} \wedge N_{ijk}$. We would like to show that it is enough to estimate the probabilities of the events B_{jk} and N_{ijk} separately to derive a lower bound on the value of Q.

Recall, that after randomized rounding step our algorithm generates a collection of partial permutation matrices that in turn generate an infeasible solution $(\bar{y}_{ijk}, (i,j,k) \in V_1 \times V_2 \times V_3)$. Let $Y_{jk} = (\bar{y}_{ijk} | i \in V_1)$ be a $\{0,1\}$-vector corresponding to column C_{jk}. Let D_{jk} be the event that we chose Y_{jk} for column C_{jk} after randomized rounding and Ω_D be the set of all possible events D_{jk}.

Lemma 8 *The events B_{jk} and N_{ijk} are independent conditioned on the event D_{jk}.*

Proof. Indeed, assume we fixed vector D_{jk}. We claim that events B_{jk} and N_{ijk} are defined by non-intersecting sets of independent random variables. Indeed, let $Y^1_{jk} = \{(i',j,k) \in C_{jk} | \bar{y}_{i'jk} = 1\}$ and $Y^0_{jk} = \{(i',j,k) \in C_{jk} | \bar{y}_{i'jk} = 0\}$. Then the event B_{jk} depends only on random variables $R_{i'jk}$ for $(i',j,k) \in Y^0_{jk}$ and $u_{i'}$ for $(i',j,k) \in Y^1_{jk}$. All these random variables are independent.

Consider now event N_{ijk}. If $\bar{y}_{ijk} = 1$ then $Pr(N_{ijk}) = 1$ since the fact that we choose a partial permutation matrix (or a matching) implies that there are no conflicting triples sharing the same first index i with the triple (i,j,k) and there is nothing to prove. Assume that $\bar{y}_{ijk} = 0$. Then there could be at most one triple (i,j',k) for all $k \in V_3$ such that $\bar{y}_{ij'k} = 1$ and at most one triple (i,j,k') for all $j \in V_2$ such that $\bar{y}_{ijk'} = 1$ since we choose a partial permutation for each index i. The event of N_{ijk} happens when both such triples are not chosen in the end of the iteration due to the contention resolution, i.e. $\tilde{y}_{ij'k} = \tilde{y}_{ijk'} = 0$. These events depend either on random variables R_{pqr} for the triples $(p, q, r) \in Y^0_{j'k} \cup Y^0_{jk'}$ or variables u_p for the triples $(p, q, r) \in Y^1_{j'k} \cup Y^1_{jk'}$. All these variables are independent of each other.

The key observation is to notice that if a triple $(p, j, k) \in Y^1_{jk}$ then the triples $(p, j', k) \notin Y^1_{j'k}$ and $(p, j, k') \notin Y^1_{jk'}$ since for each $p \in V_1$ the elements (p, q, r) with $\bar{y}_{pqr} = 1$ form a partial permutation matrix. Therefore, the sets of variables u_p for $p \in V_1$ that events B_{jk} and N_{ijk} depend on are non-intersecting for these two events. Therefore, both events depend on different sets of independent random variables u_p for $p \in V_1$ and therefore these events are independent. $\square$

Corollary 2

$$Pr(B_{jk} \wedge N_{ijk}|D_{jk}) = Pr(B_{jk}|D_{jk})Pr(N_{ijk}|D_{jk}),$$
$$Pr(B_{jk} \wedge N_{ijk} \wedge D_{jk})Pr(D_{jk}) = Pr(B_{jk} \wedge D_{jk})Pr(N_{ijk} \wedge D_{jk}).$$

Let $D_{0jk} \in \Omega_D$ be an event that the vector Y_{jk} consists of zeros only.

Lemma 9 *$Pr(D_{jk}|B_{jk}) \leq Pr(D_{jk})$ for any $D_{jk} \in \Omega_D \setminus D_{0jk}$.*

The proof of this Lemma can be found in Appendix.

Lemma 10 *$Pr(N_{ijk}|D_{0jk}) \geq Pr(N_{ijk}|D_{jk})$ for any $D_{jk} \in \Omega_D \setminus D_{0jk}$;*

Proof. The proof this lemma is very similar with the proof of the item 3 in the Lemma 9. Let I be the set of indices such that $\bar{y}_{sjk} = 1$ when the event D_{jk} happens, i.e. I corresponds to the set of chosen triples during the event D_{jk}. We again use the conditioning on the choice of random variables. We fix all random variables u and R. We also fix the choice of random matchings during the randomized rounding for the indices $p \in V_1 \setminus I$ and we call the set of those fixed matchings $\mathcal{M}$. The only random choice left is the choice of random matchings by randomized rounding for the indices $p \in I$.

Recall, that N_{ijk} is the event in the end of the current phase there is no triples (i, j', k') chosen that conflict with the triple (i, j, k). We claim that $Pr(N_{ijk}|D_{0jk} \wedge u \wedge R \wedge \mathcal{M}) \geq Pr(N_{ijk}|D_{jk} \wedge u \wedge R \wedge \mathcal{M})$ for any $D_{jk} \in \Omega_D \setminus D_{0jk}$. First notice that under this conditioning $Pr(N_{ijk}|D_{jk} \wedge u \wedge R \wedge \mathcal{M}) \in \{0, 1\}$ since any matching we can choose randomly for index $p \in I$ must have 1 in the column C_{jk} and therefore $\bar{y}_{pj'k} = \bar{y}_{pjk'} = 0$ for all indices $j' \neq j$ and $k' \neq k$. Therefore, the choice of such a matching cannot influence the contention resolution for the triples (i, j', k') conflicting with the triple (i, j, k) (i.e. either $j' = j$ or $k' = k$).

Assume now that $Pr(N_{ijk}|D_{jk} \wedge u \wedge R \wedge \mathcal{M}) = 1$ and consider $Pr(N_{ijk}|D_{0jk} \wedge u \wedge R \wedge \mathcal{M})$. Again the only random choice left is to choose matchings for $p \in I \subseteq V_1$ such that these matchings have zeros in the column C_{jk} and ones in the columns $C_{j'k}$ and $C_{jk'}$ of the triples that can potentially conflict with (i, j, k). It can can only increase the probability of the event N_{ijk} since having additional ones (instead of zeros) in the columns $C_{j'k}$ and $C_{jk'}$ can only decrease the probability of the triples $(i, j'k)$ and (i, j, k') winning the contention resolution. $\square$

Finally, combining the previous Lemmas we derive the main Lemma in this Section.

Lemma 11 *$Pr(N_{ijk}|B_{jk}) \geq Pr(N_{ijk})$*

Proof. Obviously, $Pr(N_{ijk}) = \sum_{D_{jk} \in \Omega_D} Pr(N_{ijk}|D_{jk})Pr(D_{jk})$. By Corollary 2, we have

$$Pr(N_{ijk}|B_{ijk}) = \frac{Pr(N_{ijk} \wedge B_{ijk})}{Pr(B_{ijk})} = \sum_{D_{jk} \in \Omega_D} \frac{Pr(N_{ijk} \wedge B_{ijk} \wedge D_{jk})}{Pr(B_{jk})} =$$
$$\sum_{D_{jk} \in \Omega_D} \frac{Pr(B_{jk} \wedge D_{jk})Pr(N_{ijk} \wedge D_{jk})}{Pr(B_{jk})Pr(D_{jk})} = \sum_{D_{jk} \in \Omega_D} Pr(N_{ijk}|D_{jk})Pr(D_{jk}|B_{jk}). \quad (10)$$

Therefore, $Pr(N_{ijk}|B_{jk}) - Pr(N_{ijk}) = \sum_{D_{jk}\in\,\Omega_D} Pr(N_{ijk}|D_{jk})\,(Pr(D_{jk}|B_{jk}) - Pr(D_{jk})))$. We now prove that

$$\sum_{D_{jk}\in\,\Omega_D} Pr(N_{ijk}|D_{jk})\,(Pr(D_{jk}|B_{jk}) - Pr(D_{jk}))) \geq 0.$$

By using Lemmas 9 and 10 and the fact that $\sum_{D_{jk}\in\,\Omega_D} Pr(D_{jk}|B_{jk}) = \sum_{D_{jk}\in\,\Omega_D} Pr(D_{jk}) = 1$, we derive

$$\begin{aligned}
\sum_{D_{jk}\in\,\Omega_D} Pr(N_{ijk}|D_{jk})\,(Pr(D_{jk}|B_{jk}) - Pr(D_{jk}))) &= \\
Pr(N_{ijk}|D_{0jk})\,(Pr(D_{0jk}|B_{jk}) - Pr(D_{0jk}))) &+ \\
\sum_{D_{jk}\in\,\Omega_D\setminus D_{0jk}} Pr(N_{ijk}|D_{jk})\,(Pr(D_{jk}|B_{jk}) - Pr(D_{jk}))) &\geq \\
Pr(N_{ijk}|D_{0jk})\,(Pr(D_{0jk}|B_{jk}) - Pr(D_{0jk}))) &+ \\
\sum_{D_{jk}\in\,\Omega_D\setminus D_{0jk}} Pr(N_{ijk}|D_{0jk})\,(Pr(D_{jk}|B_{jk}) - Pr(D_{jk}))) &= \\
Pr(N_{ijk}|D_{0jk}) \sum_{D_{jk}\in\,\Omega_D} (Pr(D_{jk}|B_{jk}) - Pr(D_{jk}))) &= 0.
\end{aligned}$$

□

The Lemmas 7 and 11 imply that $Q = Pr(B_{jk})Pr(N_{ijk}|B_{jk}) \geq Pr(B_{jk})Pr(N_{ijk}) \geq e^{-1}Pr(N_{ijk})$. The next section is devoted to estoimating the quantity $Pr(N_{ijk})$.

6 Estimating $Pr(N_{ijk})$

In this section, we give a lower bound on the probability of the event N_{ijk}. More precisely, the goal of this section is to show that $Pr(N_{ijk}) > LB = 0.1508$. Recall, that the event N_{ijk} is the event that all triples with the same index i that conflict with the triple (i, j, k) were not chosen on the current phase of our rounding algorithm. We lower bound the expression for $Pr(N_{ijk})$ from below by using a complicated function of four variables which then is minimized by using Mathematica. We also wrote our own independent optimizer to check the validity of the obtained answer. Our optimizer found the same minimum solution. This bound leads us to the estimate on the value of $Q > e^{-1}\cdot 0.1508 \approx 0.0555$. The proof of this fact is highly technical and is given in the Appendix due to lack of space.

7 Conclusion

The most interesting open question left unresolved in this paper is to find a rounding algorithm for the linear programming relaxation of the planar three-index assignment problem that matches its integrality gap.

References

[1] G. Appa, D. Magos and I. Mourtos: A new class of facets for the Latin square polytope. Discrete Applied Mathematics 154(6): 900-911 (2006)

[2] E. Arkin and R. Hassin, On local search for weighted k-set packing, *Math. of Oper. Research* 23(3) (1998), 640–648.

[3] E. Balas and P. Landweer, Traffic assignment in communication sattelites, Operations Research Letters v. 2 (1983), pp. 141-147.

[4] R. Barry and P. Humblet, Latin routers, design and implementation, IEEE/OSA J. Lightwave Tech. v.11 (1993), pp. 891-899.

[5] P. Berman, A $d/2$ approximation for maximum weight independent set in d-claw free graphs, *Nordic J. Comput.* 7(3) (2000), 178–184.

[6] R. Burkard, M. Dell'Amico and S. Martello, Assignment Problems, Society for Industrial and Applied Mathematics, Philadelphia, 2009.

[7] R. Burkard and E. Cela, Linear assignment problems and extensions. Handbook of combinatorial optimization, Supplement Vol. A, 75–149, Kluwer Acad. Publ., Dordrecht, 1999.

[8] J. Denes and A. Keedwell, Latin squares and their applications. Academic Press, New York-London, 1974.

[9] J. Denes and A. Keedwell, Latin squares. New developments in the theory and applications. With contributions by G. B. Belyavskaya, A. E. Brouwer, T. Evans, K. Heinrich, C. C. Lindner and D. A. Preece. With a foreword by Paul Erdos. Annals of Discrete Mathematics, 46. North-Holland Publishing Co., Amsterdam, 1991.

[10] R. Euler, R. Burkard and R. Grommes, On latin squares and the facial structure of related polytopes. Discrete Mathematics 62(2), 155-181 (1986).

[11] U. Feige, On maximizing welfare when utility functions are subadditive. STOC'06: Proceedings of the 38th Annual ACM Symposium on Theory of Computing, 41–50, ACM, New York, 2006.

[12] U. Feige and J. Vondrak, Approximation algorithms for allocation problems: Improving the factor of 1-1/e, in FOCS 2006, 667-676.

[13] A. Frieze, Complexity of a 3-dimensional assignment problem. European J. Oper. Res. 13 (1983), no. 2, 161–164.

[14] K. Gilbert and R. Hofstra, An algorithm for a class of three-dimensional assignment problems arising in scheduling applications, IIE Transactions, v.8 (1987), pp. 29-33.

[15] C. Gomes, R. Regis and D. Shmoys, An improved approximation algorithm for the partial Latin square extension problem. Oper. Res. Lett. 32 (2004), no. 5, 479–484.

[16] I. Hajirasouliha, H. Jowhari, R. Kumar and R. Sundaram, On completing Latin squares. STACS 2007, 524–535, Lecture Notes in Comput. Sci., 4393, Springer, Berlin, 2007.

[17] A. Hilton, The reconstruction of Latin squares with applications to school timetabling and to experimental design. Combinatorial optimization, II (Proc. Conf., Univ. East Anglia, Norwich, 1979). Math. Programming Stud. No. 13 (1980), 68–77.

[18] C. Hurkens and A. Schrijver, On the size of systems of sets every t of which have an SDR, with an application to the worst-case ratio of heuristics for packing problems. SIAM J. Discrete Math. 2 (1989), no. 1, 68–72.

[19] S. Kumar, A. Russell and R. Sundaram, Approximating Latin square extensions. Algorithmica 24 (1999), no. 2, 128–138.

[20] D. Magos and I. Mourtos, The wheels of the OLS polytope: Facets and separation. Discrete Mathematics 308(16), 3634-3651 (2008).

[21] M. Queyranne and F.C.R. Spieksma, Multi-index transportation problems, in: "the Encyclopedia of Optimization", edited by C. Floudas and P. Pardalos, Kluwer (2001), Dordrecht.

[22] F.C.R. Spieksma, Multi index assignment problems: complexity, approximation, applications, in: "Nonlinear Assignment Problems, Algorithms and Applications", edited by L. Pitsoulis and P. Pardalos, (2000), Kluwer, Dordrecht, pages 1-12.

A Proof of Lemma 9

Proof. Instead of the statement in the Lemma we will show that $Pr(B_{jk}|D_{jk}) \leq Pr(B_{jk})$ holds for any $D_{jk} \in \Omega_D \setminus D_{0jk}$. Then the statement of the Lemma follows from the equality $Pr(A|B) = Pr(B|A) \cdot Pr(A)/Pr(B)$. We fix an event $D_{jk} \in \Omega_D \setminus D_{0jk}$. Let I be the corresponding set of indices such that $\bar{y}_{sjk} = 1$ for $s \in I$. Let (i', j, k) be a triple such that $i' \in I$ and $z_{i'jk} = \min_{s \in I} z_{sjk}$.

We now define a collection of various events and show relationships between them. Let E_{jk} be the event that $s_{i'jk} > z_{i'jk}$. Let F_{jk} be the event that $s_{i'jk} \leq z_{i'jk}$ and the winner of the contention resolution for the column C_{jk} is some triple (w, j, k) for $w \in V_1 \setminus I$. Let $G_{jk}(s)$ be the event that $\bar{y}_{sjk} = 1$. Note that $D_{jk} = \wedge_{s \in I} G_{jk}(s) \wedge_{s \in V_1 \setminus I} \overline{G_{jk}(s)}$. Finally, let H_{jk} be the event that in the final solution the element $\tilde{y}_{i',j,k} = 0$, i.e. the triple (i', j, k) is not a winner of the contention resolution scheme.

Then we claim the following relations between the above defined events:

1. $Pr(B_{jk}|D_{jk}) \leq Pr(E_{jk}|D_{jk}) + Pr(F_{jk}|D_{jk})$, this fact follows directly from the observations that $E_{jk} \cap F_{jk} = \emptyset$ and $B_{jk} \subseteq E_{jk} \cup F_{jk}$;

2. $Pr(E_{jk}|D_{jk}) = Pr(E_{jk}|G_{jk}(i'))$, this equality is a direct consequence of the fact that E_{jk} is completely independent on the triples of the column C_{jk} other than the triple (i', j, k) and therefore

$$
\begin{aligned}
Pr(E_{jk}|D_{jk}) = Pr(E_{jk}| \wedge_{s \in I} G_{jk}(s) \wedge_{s \in V_1 \setminus I} \overline{G_{jk}(s)})) &= \\
\frac{Pr\left(E_{jk} \wedge (\wedge_{s \in I} G_{jk}(s) \wedge_{s \in V_1 \setminus I} \overline{G_{jk}(s)})\right)}{Pr\left(\wedge_{s \in I} G_{jk}(s) \wedge_{s \in V_1 \setminus I} \overline{G_{jk}(s)}\right)} &= \\
\frac{Pr\left(E_{jk} \wedge G_{jk}(i'))\right)}{Pr\left(G_{jk}(i')\right)} &= Pr(E_{jk}|G_{jk}(i'));
\end{aligned}
$$

3. $Pr(F_{jk}|D_{jk}) \leq Pr(F_{jk}|G_{jk}(i'))$, this inequality is a consequence of the following observations. For each $i \in V_1$, fix random variables u_i and R_{ijk} then $Pr(F_{jk}|D_{jk} \wedge u \wedge R) \in \{0, 1\}$ while $Pr(F_{jk}|G_{jk}(i') \wedge u \wedge R) \in [0, 1]$ since once we fix random variables u and R and vector D_{jk} our contention resolution scheme becomes deterministic. Moreover, we claim that if $Pr(F_{jk}|D_{jk} \wedge u \wedge R) = 1$ then $Pr(F_{jk}|G_{jk}(i') \wedge u \wedge R) = 1$. To show this consider a specific choice of u and R when $Pr(F_{jk}|D_{jk} \wedge u \wedge R) = 1$. We would like to show that for any vector $\bar{D}_{jk}$ such that $\bar{y}_{i'jk} = 1$ we have that $Pr(F_{jk}|\bar{D}_{jk} \wedge u \wedge R) = 1$, i.e. it does not matter which matchings were chosen on the first rounding step as long as we condition on the event $G_{jk}(i')$.

   By the definition of F_{jk}, we know that $s_{i'jk}(u_{i'}) \leq z_{i'jk}$ and at the same time $R_{wjk} + s_{wjk}(1) \leq s_{i'jk}(u_{i'})$ for some $w \in V_1 \setminus I$ (w is the winner of the contention resolution for column C_{jk}). That is there is an index $w \in V_1 \setminus I$ with smallest value of s_{wjk}. In the vector $\bar{D}_{jk}$ some zeros in the set $V_1 \setminus I$ can become ones and their parameter s will go down (by the definition of s).

This can change the winner of the contention resolution but since the winner will still be from the set $V_1 \setminus I$, the event F_{jk} will occur. Also some ones in the set I can become zeros and their s-value will increase and therefore, if they were not winners before they cannot win the contention resolution after that change. In any case for a new vector some element of the set $V_1 \setminus I$ is the winner of the contention resolution, i.e. the triple with smallest s-value will be from the set $V_1 \setminus I$ which is exactly the definition of the event F_{jk} . Therefore, $Pr(F_{jk}|\bar{D}_{jk} \wedge u \wedge R) = 1$.

The inequality $Pr(F_{jk}|D_{jk}) \leq Pr(F_{jk}|G_{jk}(i'))$ now follows by rewriting the inequality using the conditioning on u and R.

4. $Pr(E_{jk}|G_{jk}(i')) + P(F_{jk}|G_{jk}(i')) \leq Pr(H_{jk}|G_{jk}(i'))$, since $E_{jk} \cap F_{jk} = \emptyset$ and under both events the triple (i', j, k) is not a winner of the contention resolution scheme;

5. Combining items 1-4 we obtain $Pr(B_{jk}|D_{jk}) \leq Pr(E_{jk}|G_{jk}(i'))+P(F_{jk}|G_{jk}(i')) \leq Pr(H_{jk}|G_{jk}(i')) = e^{-1} \leq P(B_{jk})$, where the last equality and inequality follow from Corollary 1 and Lemma 7.

□

B Estimating N_{ijk}

In this section, we give a lower bound on the probability of the event N_{ijk}. More precisely, the goal of this section is to show that $Pr(N_{ijk}) > LB = 0.1508$. Recall, that the event N_{ijk} is the event that all triples with the same index i that conflict with the triple (i, j, k) were not chosen on the current phase of our rounding algorithm. Let $\prod(M_i)$ be the partial permutation matrix (matching) chosen on the randomized rounding step of the current phase for the index $i \in V_1$. If $(i, j, k) \in M_i$, i.e. $\bar{y}_{ijk} = 1$ then there are no conflicting triples (i, j', k') such that either $j' = j$ or $k' = k$. Therefore, in this case $Pr(N_{ijk}|M_i) = 1 > LB$. Another easy case is when matching M_i contains only one conflicting triple either (i, j, k') or (i, j', k) (it can happen since our matching is not necessarily perfect). In this case by Corollary 1, $Pr(N_{ijk}|M_i) = e^{-1} > LB$.

We now assume that the matching M_i contains exactly two triples (i, j_1, k) and (i, j, k_2) conflicting with (i, j, k). The event N_{ijk} occurs if both triples (i, j_1, k) and (i, j, k_2) lose the contention resolution. We fix the random variable u_i corresponding to the matching M_i. We will estimate the probability $Pr(N_{ijk}|M_i \wedge u_i)$ from below depending on different values of u_i. Recall, that if $z_{ij_1k} < s_{ij_1k}$ and $z_{ijk_2} < s_{ijk_2}$ then both triples (i, j_1, k) and (i, j, k_2) lose the contention resolution. We define $w_1 = u(z_{ij_1k})$ and $w_2 = u(z_{ijk_2})$ to be the critical values for random variable u_i (we define $u(s)$ to be the inverse of $s(u)$). Let $u_{min} = \min(w_1, w_2)$ and $u_{max} = \max(w_1, w_2)$. First we show lower bounds in easy cases.

Lemma 12

1. *if* $u_i \geq u_{max}$ *then* $Pr(N_{ijk}|M_i \wedge u_i) = 1$*;*

2. *if* $u_i \in [u_{min} = w_1, u_{max} = w_2]$ *then* $Pr(N_{ijk}|M_i \wedge u_i) = 1 - e^{-s_{ijk_2}(1-b_{jk_2}-y_{ijk_2}+ay^2_{ijk_2})}$,

3. *if* $u_i \in [u_{min} = w_2, u_{max} = w_1]$ *then* $Pr(N_{ijk}|M_i \wedge u_i) = 1 - e^{-s_{ij_1k}(1-b_{j_1k}-y_{ij_1k}+ay^2_{ij_1k})}$

Proof. If $u_i \geq u_{max}$ then both triples (i, j_1, k) and (i, j, k_2) cannot win the contention resolution since in this case $s_{ij_1k} > z_{ij_1k}$ and $s_{ijk_2} > z_{ijk_2}$. Therefore, the event N_{ijk} conditioned on the choices of M_i and u_i always occurs.

In the case when $u_i \in [u_{min} = w_1, u_{max} = w_2]$, the triple (i, j_1, k) cannot win the contention resolution by the same argument. The triple (i, j, k_2) wins the contention resolution if for any triple $(i', j, k_2) \in U_{jk_2}$, i.e. any triple that participates in the rounding on the current stage including the dummy triple $(0, j, k_2)$ we have $s_{i'jk_2} > s_{ijk_2}$. This events happens with probability $e^{-s_{ijk_2}(y_{i'jk_2} - ay^2_{i'jk_2})}$ by the Lemma 3. Since the random variables $u_{i'}$ are chosen independently for different indices i' we obtain that the probability of not winning the contention resolution for the triple (i, j, k_2) is exactly

$$1 - \prod_{(i',j,k_2) \in U_{jk} \setminus \{(i,j,k_2)\}} e^{-s_{ijk_2}(y_{i'jk_2} - ay^2_{i'jk_2})} = 1 - e^{-s_{ijk_2}(1 - b_{jk_2} - y_{ijk_2} + ay^2_{ijk_2})}.$$

The last case is proved analogously to the case 2. $\square$

The next lemma corresponds to the most difficult case when $u_i < u_{min}$. In this case the events in the columns C_{j_1k} and C_{jk_2} are dependent and we need to use a more sophisticated argument.

Lemma 13 *If $u_i < u_{min}$ then*

$$\begin{aligned} P(N_{ijk} | M_i \wedge u_i) \geq & \left(1 - e^{-s_{ij_1k}(1 - b_{j_1k} - y_{ij_1k} + ay^2_{ij_1k})}\right) \times \\ & \left(1 - e^{-s_{ijk_2}(1 - b_{jk_2} - y_{ijk_2} + ay^2_{ijk_2})}\right) \times \\ & \left(1 - \frac{\sqrt{(b_{j_1k}/a - y^2_{ij_1k})(b_{jk_2}/a - y^2_{ijk_2})}}{(1 - y_{ij_1k})(1 - y_{ijk_2})}\right). \end{aligned}$$

To prove Lemma 13, we will need the following technical facts.

Lemma 14 *Given two sequences of real numbers $0 < a_1 \leq \cdots \leq a_n$ and $b_1 \geq \cdots \geq b_n > 0$ such that $a_1 b_1 \leq a_2 b_2 \leq \cdots \leq a_n b_n$ then*

$$\frac{\sqrt{\sum_{i=1}^n (a_i b_i)^2}}{\sum_{i=1}^n a_i b_i} \leq \frac{\sqrt{\sum_{i=1}^n a_i^2}}{\sum_{i=1}^n a_i}$$

Proof. Obviously if $b_i = b$ for all $i = 1, \ldots, n$ then the inequality holds with equality. Assume now that $b_1 = \cdots = b_k = b$ for some $k \geq 1$ and $b_k > b_{k+1}$. We define $b'_1 = \cdots = b'_k = b - \varepsilon$ and $b'_i = b_i + \varepsilon'$ for $i = k+1, \ldots, n$. Where ε and ε' are the roots of the following system of linear equations

$$\varepsilon \sum_{i=1}^k a_i = \varepsilon' \sum_{i=k+1}^n a_i,$$
$$b - \varepsilon = b_{k+1} + \varepsilon'.$$

Therefore,

$$\varepsilon' = \frac{b - b_{k+1}}{\sum_{i=1}^n a_i} \cdot \sum_{i=1}^k a_i \text{ and } \varepsilon = \frac{b - b_{k+1}}{\sum_{i=1}^n a_i} \cdot \sum_{i=k+1}^n a_i.$$

Moreover, the new collection satisfies the conditions of the lemma, i.e. $b'_1 \geq \cdots \geq b'_n > 0$ and

$a_1 b'_1 \le a_2 b'_2 \le \cdots \le a_n b'_n$. We now show that $\sum_{i=1}^n (a_i b'_i)^2 \ge \sum_{i=1}^n (a_i b_i)^2$. Indeed,

$$\begin{aligned}
\sum_{i=1}^n (a_i b'_i)^2 = \sum_{i=1}^n (a_i b_i)^2 + \varepsilon^2 \sum_{j=1}^k a_j^2 + \varepsilon'^2 \sum_{j=k+1}^n a_j^2 - 2\varepsilon \sum_{j=1}^k a_j^2 b_j + 2\varepsilon' \sum_{j=k+1}^n a_j^2 b_j &\ge \\
\sum_{i=1}^n (a_i b_i)^2 - 2\varepsilon \sum_{j=1}^k a_j^2 b_j + 2\varepsilon' \sum_{j=k+1}^n a_j^2 b_j &\ge \\
\sum_{i=1}^n (a_i b_i)^2 - 2\varepsilon a_{k+1} b_{k+1} \sum_{j=1}^k a_j + 2\varepsilon' a_{k+1} b_{k+1} \sum_{j=k+1}^n a_j &= \sum_{i=1}^n (a_i b_i)^2.
\end{aligned}$$

Repeating this process, we get the final collection $b'_1 = \cdots = b'_n = b > 0$ such that $\sum_{i=1}^n (a_i b'_i)^2 \ge \sum_{i=1}^n (a_i b_i)^2$ and $\sum_{i=1}^n a_i b_i = \sum_{i=1}^n a_i b'_i$. The Lemma follows from the observation in the beginning of the proof that

$$\frac{\sqrt{\sum_{i=1}^n (a_i b'_i)^2}}{\sum_{i=1}^n a_i b'_i} \le \frac{\sqrt{\sum_{i=1}^n a_i^2}}{\sum_{i=1}^n a_i}.$$

□

Lemma 15 *For any A and B such that $0 < A < B$ the following inequality holds*

$$(1 - e^{-A}) \ge \frac{A}{B}(1 - e^{-B})$$

Proof. The inequality follows from the monotonicity of the function $(1 - e^{-x})/x$ for $x > 0$. □

B.1 Correlation Property

Let $f_{ijk} = y_{ijk} - a y_{ijk}^2$. For any index $i' \in V_1 \setminus \{i\}$, let $J_{i'}$ be the event that $s_{i'j_1k} < s_{ij_1k}$ and let $K_{i'}$ be the event that $s_{i'jk_2} < s_{ijk_2}$. Notice that any set of J and K events with distinct indices are independent, e.g. $\{J_1, J_2, \ldots, J_n\}$ or $\{K_1, J_2, K_3, J_4, ..\}$ since they depend on different independent random variables, but events J_i and K_i might be correlated. Since would like to estimate $Pr(N_{ijk} | M_i \wedge u_i)$, we will use the notation J'_i, K'_i and N'_{ijk} for the events K_i, J_i and N_{ijk} conditioned on M_i and u_i. By the Lemma 3, we have $Pr(J_{i'}) = 1 - e^{-s_{ij_1k} f_{i'j_1k}}$ and $Pr(K_{i'}) = 1 - e^{-s_{ijk_2} f_{i'jk_2}}$. Using these two fact we derive

$$Pr(\vee_{i' \ne i} J_{i'}) = 1 - \prod_{i' \ne i} P(\bar{J}_{i'}) = 1 - \prod_{i' \ne i} e^{-s_{ij_1k} f_{i'j_1k}} = 1 - \prod_{i' \ne i} e^{-s_{ij_1k}(1 - b_{j_1k} - f_{ij_1k})}.$$

Analogously we derive

$$Pr(\vee_{i' \ne i} K_{i'}) = 1 - \prod_{i' \ne i} e^{-s_{ijk_2}(1 - b_{jk_2} - f_{ijk_2})}.$$

Moreover, the same formulas hold for the events involving $K'_{i'}$ and $J'_{i'}$, because of independence from level i.

Let $J'_{ii'} = \vee_{q \ne i, i'} J'_q$ and $K'_{ii'} = \vee_{q \ne i, i'} K'_q$. As we noted before the events involving different indices $i' \in V_1 \setminus \{i\}$ are independent. In particular, the events $J'_{i'}$ and $J'_{ii'}$ are mutually independent. Similarly, the following pairs of events are mutually independent $(J'_{ii'}, K'_{i'})$, $(K'_{ii'}, J'_{i'})$, $(K'_{ii'}, K'_{i'})$.

Notice that $N'_{ijk} = (J'_{ii'} \vee J'_{i'}) \wedge (K'_{ii'} \vee K'_{i'})$ that basically means that N'_{ijk} occurs in this case only when both triples (i, j_1, k) and (i, j, k_2) lose the contention resolution to one of the triples in their respective columns. By the standard distributive law of set union and intersection we get

$$(J'_{ii'} \vee J'_{i'}) \wedge (K'_{ii'} \vee K'_{i'}) = (J'_{ii'} \wedge K'_{ii'}) \vee (J'_{ii'} \wedge K'_{i'}) \vee (K'_{ii'} \wedge J'_{i'}) \vee (J'_{i'} \wedge K'_{i'}).$$

We need an analog of above equality such that the right hand side contains the union of disjoint events, in this case we could write the probability of the left hand side as the sum of probabilities of the events in the right hand side. The analog of the standard distributive law using the disjoint unions only is

$$(J'_{ii'}\vee J'_{i'})\wedge(K'_{ii'}\vee K'_{i'}) = (J'_{ii'}\wedge K'_{ii'})\vee(J'_{ii'}\wedge\bar{K}'_{ii'}\wedge K'_{i'})\vee(\bar{J}'_{ii'}\wedge K'_{ii'}\wedge J'_{i'})\vee(\bar{J}'_{ii'}\wedge\bar{K}'_{ii'}\wedge J'_{i'}\wedge K'_{i'}). \quad (11)$$

Therefore, using the observation that the events on right hand side of (11) are disjoint and the independence we obtain for u_i such that $s_{ij_1k}(u_i) < z_{ij_1k}$ and $s_{ijk_2}(u_i) < z_{ijk_2}$

$$\begin{aligned} Pr(N'_{ijk}) = Pr(J'_{ii'} \wedge K'_{ii'}) + Pr(J'_{ii'} \wedge \bar{K}'_{ii'} \wedge K'_{i'}) &\quad + \\ Pr(\bar{J}'_{ii'} \wedge K'_{ii'} \wedge J'_{i'}) + Pr(\bar{J}'_{ii'} \wedge \bar{K}'_{ii'} \wedge J'_{i'} \wedge K'_{i'}) &\quad = \\ Pr(J'_{ii'} \wedge K'_{ii'}) + Pr(K'_{i'}) \cdot Pr(J'_{ii'} \wedge \bar{K}'_{ii'}) &\quad + \\ Pr(J'_{i'}) \cdot Pr(\bar{J}'_{ii'} \wedge K'_{ii'}) + Pr(J'_{i'} \wedge K'_{i'}) \cdot Pr(\bar{J}'_{ii'} \wedge \bar{K}'_{ii'}). & \end{aligned} \quad (12)$$

Consider now some index $i' \in V_1 \setminus \{i\}$ and two corresponding variables $y_{i'j_1k}$ and $y_{i'jk_2}$. In the decomposition of the matrix $Y_{i'}$ into partial permutation matrices there are matchings that contain both edges corresponding to the triples (i', j_1, k) and (i', j, k_2) and there are matchings that contain only single such edge. Then for some $\Delta_{i'} \le \min\{y_{i'jk_2}, y_{i'j_1k}\}$ we have

$$\begin{aligned} \sum_{M_{it}|(j,k_2)\in M_{it},(j_1,k)\in M_{it}} \lambda_{it} &= \Delta_{i'}, \\ \sum_{M_{it}|(j,k_2)\in M_{it},(j_1,k)\notin M_{it}} \lambda_{it} &= y_{i'jk_2} - \Delta_{i'}, \\ \sum_{M_{it}|(j,k_2)\notin M_{it},(j_1,k)\in M_{it}} \lambda_{it} &= y_{i'j_1k} - \Delta_{i'}, \\ \sum_{M_{it}|(j,k_2)\notin M_{it},(j_1,k)\notin M_{it}} \lambda_{it} &= 1 - y_{i'j_1k} - y_{i'jk_2} + \Delta_{i'}. \end{aligned}$$

That is $\Delta_{i'}$ is the total weight of matchings that contains ones corresponding to both triples (i', j_1, k) and (i', j, k_2).

We claim that $Pr(N'_{ijk})$ is minimized when $\Delta_{i'} = \max\{0, y_{i'j_1k} + y'_{i'jk_2} - 1\}$, i.e. for each i' the amount of matchings that contain both edges (j_1, k) and (j, k_2) is as small as possible. We prove it iteratively. Consider one index i' and the equation (12). Out of all terms in the equation (12) only the term $Pr(J'_{i'} \wedge K'_{i'})$ depends on $\Delta_{i'}$. Moreover, we can compute $Pr(J'_{i'} \wedge K'_{i'})$ by the formula

$$\begin{aligned} Pr(J'_{i'} \wedge K'_{i'}) = \Delta_{i'} Pr(J'_{i'} \wedge K'_{i'}|\bar{y}_{i'j_1k} = 1 \wedge \bar{y}_{i'jk_2} = 1) + \\ (y_{i'jk_2} - \Delta_{i'}) Pr(J'_{i'} \wedge K'_{i'}|\bar{y}_{i'j_1k} = 1 \wedge \bar{y}_{i'jk_2} = 0) + \\ (y_{i'j_1k} - \Delta_{i'}) Pr(J'_{i'} \wedge K'_{i'}|\bar{y}_{i'j_1k} = 0 \wedge \bar{y}_{i'jk_2} = 1) + \\ (1 - y_{i'j_1k} - y_{i'jk_2} + \Delta_{i'}) Pr(J'_{i'} \wedge K'_{i'}|\bar{y}_{i'j_1k} = 0 \wedge \bar{y}_{i'jk_2} = 0) = \end{aligned}$$

we simplify the formula by using the independence that follows from the fact that events are defined by non-crossing sets of independent random variables are independent

$$\begin{aligned} \Delta_{i'} Pr(J'_{i'} \wedge K'_{i'}|\bar{y}_{i'j_1k} = 1 \wedge \bar{y}_{i'jk_2} = 1) + \\ (y_{i'jk_2} - \Delta_{i'}) Pr(J'_{i'}|\bar{y}_{i'j_1k} = 1 \wedge \bar{y}_{i'jk_2} = 0) Pr(K'_{i'}|\bar{y}_{i'j_1k} = 1 \wedge \bar{y}_{i'jk_2} = 0) + \\ (y_{i'j_1k} - \Delta_{i'}) Pr(J'_{i'}|\bar{y}_{i'j_1k} = 0 \wedge \bar{y}_{i'jk_2} = 1) Pr(K'_{i'}|\bar{y}_{i'j_1k} = 0 \wedge \bar{y}_{i'jk_2} = 1) + \\ (1 - y_{i'j_1k} - y_{i'jk_2} + \Delta_{i'}) Pr(J'_{i'}|\bar{y}_{i'j_1k} = 0 \wedge \bar{y}_{i'jk_2} = 0) Pr(K'_{i'}|\bar{y}_{i'j_1k} = 0 \wedge \bar{y}_{i'jk_2} = 0) = \\ \Delta_{i'} Pr(J'_{i'} \wedge K'_{i'}|\bar{y}_{i'j_1k} = 1 \wedge \bar{y}_{i'jk_2} = 1) + \\ (y_{i'jk_2} - \Delta_{i'}) Pr(J'_{i'}|\bar{y}_{i'j_1k} = 1) Pr(K'_{i'}|\bar{y}_{i'jk_2} = 0) + \\ (y_{i'j_1k} - \Delta_{i'}) Pr(J'_{i'}|\bar{y}_{i'j_1k} = 0) Pr(K'_{i'}|\bar{y}_{i'jk_2} = 1) + \\ (1 - y_{i'j_1k} - y_{i'jk_2} + \Delta_{i'}) Pr(J'_{i'}|\bar{y}_{i'j_1k} = 0) Pr(K'_{i'}|\bar{y}_{i'jk_2} = 0). \end{aligned} \quad (13)$$

We can now define some of these probabilities exactly $Pr(J'_{i'}|\bar{y}_{i'j_1k} = 1) = u_{i'j_1k}(s_{ij_1k}) = a$, $Pr(K'_{i'}|\bar{y}_{i'jk_2} = 1) = u_{i'jk_2} = b$ and $Pr(J'_{i'} \wedge K'_{i'}|\bar{y}_{i'j_1k} = 1 \wedge \bar{y}_{i'jk_2} = 1) = \min\{u_{i'j_1k}(s_{ij_1k}), u_{i'jk_2}(s_{ijk_2})\} \geq u_{i'j_1k}(s_{ij_1k}) \cdot u_{i'jk_2}(s_{ijk_2}) = ab$. Moreover, $Pr(K'_{i'}|\bar{y}_{i'jk_2} = 1) = b \geq Pr(K'_{i'}|\bar{y}_{i'jk_2} = 0) = b'$ and $Pr(J'_{i'}|\bar{y}_{i'j_1k} = 1) = a \geq Pr(J'_{i'}|\bar{y}_{i'j_1k} = 0) = a'$ since we defined s_{ijk} to be always smaller in the case when $\bar{y}_{ijk} = 1$ then in the case when $\bar{y}_{ijk} = 0$ (a, b, a', b' are just shorter notations). All these imply that $\Delta_{i'}$ is multiplied by the term $ab - ab' - a'b + a'b' = (b - b')(a - a')$ in (13) which is nonnegative and therefore $Pr(J'_{i'} \wedge K'_{i'})$ is minimized when $\Delta_{i'}$ is as small as possible which is $\max\{0, y_{i'j_1k} + y_{i'jk_2} - 1\}$. We repeat this argument iteratively for all $i' \in V_1 \setminus \{i\}$ and derive that $Pr(N'_{ijk})$ is minimized when $\Delta_{i'} = \max\{0, y_{i'j_1k} + y_{i'jk_2} - 1\}$ for all $i' \in V_1 \setminus \{i\}$.

More formally, the event N'_{ijk} depends only on the value of LP variables and random variables restricted to two column C_{j_1k} and C_{jk_2}. We can also restrict matchings defined for each $i' \in V_1$ to just these two columns. After that we can just try to change the random process in such a way that $Pr(N'_{ijk})$ decreases. In the previous paragraph, we noticed that if we replace our original random process with the process where matchings corresponding to the same index $i' \in V_1$ have smallest possible overlap then $Pr(N'_{ijk})$ decreases. After changing this process $|V_1 \setminus \{i\}|$ times (once for each index $i' \in V_1 \setminus \{i\}$) we get a different random process with probability of the event N'_{ijk} in this process lower bounding the original probability.

Another such trick that we will need in the next section is to assume that each triples (i', j_1, k) and (i', j, k_2) have independent variables u_{ij_1k} and $u_{i'jk_2}$ instead of having a single random variable $u_{i'}$. It follows from (13) that when we define two independent random variables instead of one we will only change $Pr(J'_{i'} \wedge K'_{i'}|\bar{y}_{i'j_1k} = 1 \wedge \bar{y}_{i'jk_2} = 1)$ in the expression (13) from $\min\{u_{i'j_1k}(s_{ij_1k}), u_{i'jk_2}(s_{ijk_2})\}$ to $u_{i'j_1k}(s_{ij_1k}) \cdot u_{i'jk_2}(s_{ijk_2})$ and therefore the probability of the event N'_{ijk} decreases in the new random process. From now on we consider an above defined random process when we are bounding $Pr(N'_{ijk})$ corresponding to that process.

Let $J' = \vee_{i' \neq i} J_{i'}$, $K' = \vee_{i' \neq i} K_{i'}$ and let $L'_{i'}$ be the event that i' is the index with lowest $s_{i'j_1k}$, i.e. $s_{i'j_1k} = \min_q s_{qj_1k}$ conditioned on M_i and u_i. The next subsection is devoted to proving that $P(K'_{ii'}|L'_{i'}) \geq P(K'_{ii'})$.

B.2 Proving $P(K'_{ii'}|L'_{i'}) \geq P(K'_{ii'})$.

Recall, that $G_{jk}(i')$ is the event that $\bar{y}_{i'jk} = 1$, i.e. the triple (i', j, k) was chosen on the randomized rounding phase (before the contention resolution). Using the conditioning argument that we used in the proofs of Lemmas 9 and 10 we claim

$$Pr(G_{j_1k}(i'')|L_{i'}) \leq Pr(G_{j_1k}(i'')) \tag{14}$$

and equivalently $Pr(L_{i'}|G_{j_1k}(i'')) \leq Pr(L_{i'})$ for all indices $i'' \neq i, i'$.

Indeed, if we condition on all random variables and events except the choice of the random matching for i'' and call this event $\mathcal{E}_{i''}$ we obtain that $Pr(L_{i'}|G_{j_1k}(i'') \wedge \mathcal{E}_{i''}) \in \{0, 1\}$ since the event $G_{j_1k}(i'')$ basically fixes the last remaining random choice for column C_{j_1k}. If $Pr(L_{i'}|G_{j_1k}(i'') \wedge \mathcal{E}_{i''}) = 1$ then $Pr(L_{i'}|\mathcal{E}_{i''}) = 1$ because if under all fixed random choices the triple (i', j_1, k) won the contention resolution in the column C_{j_1k} it will win this process even if $\bar{y}_{i''jk} = 0$ (since the value $s_{i''j_1k}$ can only increase in this case).

The second fact that we will need in this subsection is

$$Pr(G_{j_1k}(i'')|G_{jk_2}(i'')) \leq Pr(G_{j_1k}(i'')). \tag{15}$$

This fact follows directly from the discussion in the end of the previous subsection since

$$Pr(G_{j_1k}(i'') \wedge G_{jk_2}(i'')) = \Delta_{i''} = \max\{0, y_{i''j_1k} + y_{i''jk_2} - 1\} \leq y_{i''j_1k} \cdot y_{i''jk_2} = Pr(G_{j_1k}(i''))Pr(G_{jk_2}(i'')).$$

The third fact used in this subsection is

$$Pr(G_{jk_2}(i'')|L_{i'}) \geq Pr(G_{jk_2}(i'')) \tag{16}$$

for any index $i'' \neq i, i'$. Notice the difference with (14), here we have events corresponding to column C_{jk_2} while the conditioning is done on the events defined for column C_{j_1k}. Intuitively, if in the column C_{j_1k} we know that a winner of the contention resolution is the triple (i', j_1, k) it could only increase the probability of the the triple (i'', j, k_2) to be chosen on the randomized rounding stage. Formally,

$$\begin{aligned}
&Pr(G_{jk_2}(i'')|L_{i'}) = \\
&Pr(G_{jk_2}(i'')|G_{j_1k}(i''))Pr(G_{j_1k}(i'')|L_{i'}) + Pr(G_{jk_2}(i'')|\bar{G}_{j_1k}(i''))Pr(\bar{G}_{j_1k}(i'')|L_{i'}) = \\
&Pr(G_{jk_2}(i'')|G_{j_1k}(i''))Pr(G_{j_1k}(i'')|L_{i'}) + Pr(G_{jk_2}(i'')|\bar{G}_{j_1k}(i''))(1 - Pr(G_{j_1k}(i'')|L_{i'})) =
\end{aligned}$$

we continue by adding and subtracting the same quantity

$$\begin{aligned}
&Pr(G_{jk_2}(i'')|G_{j_1k}(i''))\left[Pr(G_{j_1k}(i'')|L_{i'}) + Pr(G_{j_1k}(i'')) - Pr(G_{j_1k}(i''))\right] + \\
&Pr(G_{jk_2}(i'')|\bar{G}_{j_1k}(i''))\left[1 - Pr(G_{j_1k}(i'')|L_{i'}) + Pr(G_{j_1k}(i'')) - Pr(G_{j_1k}(i''))\right] = \\
&Pr(G_{jk_2}(i'')|G_{j_1k}(i''))Pr(G_{j_1k}(i'') + Pr(G_{jk_2}(i'')|\bar{G}_{j_1k}(i''))Pr(\bar{G}_{j_1k}(i'')) + \\
&(Pr(G_{jk_2}(i'')|G_{j_1k}(i'')) - Pr(G_{jk_2}(i'')|\bar{G}_{j_1k}(i'')))(Pr(G_{j_1k}(i'')|L_{i'}) - Pr(G_{j_1k}(i''))) = \\
&Pr(G_{jk_2}(i'')) + \\
&(Pr(G_{jk_2}(i'')|G_{j_1k}(i'')) - Pr(G_{jk_2}(i'')|\bar{G}_{j_1k}(i'')))(Pr(G_{j_1k}(i'')|L_{i'}) - Pr(G_{j_1k}(i''))) \geq Pr(G_{jk_2}(i'')),
\end{aligned}$$

where the last inequality is the implication of inequalities (14), (15) and the fact that $Pr(A|B) \leq Pr(A)$ implies $Pr(A|B) \leq Pr(A|\bar{B})$.

Finally we are proving $Pr(K'_{ii'}|L'_{i'}) \geq Pr(K'_{ii'})$. We will prove this inequality for each index $i'' \neq i, i'$ separately, i.e. we will prove that $Pr(K'_{i''}|L'_{i'}) \geq Pr(K'_{i''})$. The main inequality will immediately follow since $K'_{ii'} = \vee_{i'' \neq i,i'} K'_{i''}$ and events $K'_{i''}$ are disjoint. Observe that $Pr(K'_{i''}|G_{jk_2}(i'')) \geq Pr(K'_{i''}|\bar{G}_{jk_2}(i''))$ because if we condition on all events and random variables except the choice of the matching for i'' and the triple (i'', j, k_2) wins the contention resolution when $\bar{y}_{i''jk_2} = 0$, it must win when $\bar{y}_{i''jk_2} = 1$.

Let $\alpha = Pr(G_{jk_2}(i'')|L'_{i'}) - Pr(G_{jk_2}(i''))$. The inequality (16) implies that $\alpha \geq 0$. Note also that $-\alpha = Pr(\bar{G}_{jk_2}(i'')|L'_{i'}) - Pr(\bar{G}_{jk_2}(i''))$.

The events $K'_{i''}$ and $L'_{i'}$ are independent when conditioned on $G_{jk_2}(i'')$ since both events are defined by the disjoint set of random variables. Here we used the fact that we have two independent random variables $u_{i''j_1k}$ and $u_{i''jk_2}$ in the modified random process defined at the end of the previous subsection. Therefore, $Pr(K'_{i''} \wedge L'_{i'}|G_{jk_2}(i'')) = Pr(K'_{i''}|G_{jk_2}(i''))Pr(L'_{i'}|G_{jk_2}(i''))$ which in turn implies $Pr(K'_{i''}|G_{jk_2}(i'') \wedge L'_{i'}) = Pr(K'_{i''}|G_{jk_2}(i''))$. Finally,

$$\begin{aligned}
Pr(K'_{i''}|L'_{i'}) &= Pr(K'_{i''}|G_{jk_2}(i''))Pr(G_{jk_2}(i'')|L'_{i'}) + Pr(K'_{i''}|\bar{G}_{jk_2}(i''))Pr(\bar{G}_{jk_2}(i'')|L'_{i'}) = \\
&Pr(K'_{i''}|G_{jk_2}(i''))(Pr(G_{jk_2}(i'')) + \alpha) + Pr(K'_{i''}|\bar{G}_{jk_2}(i''))(Pr(\bar{G}_{jk_2}(i'')) - \alpha) = \\
&Pr(K'_{i''}|G_{jk_2}(i''))Pr(G_{jk_2}(i'')) + Pr(K'_{i''}|\bar{G}_{jk_2}(i''))Pr(\bar{G}_{jk_2}(i'')) + \\
&\alpha(Pr(K'_{i''}|G_{jk_2}(i'')) - Pr(K'_{i''}|\bar{G}_{jk_2}(i''))) \geq Pr(K'_{i''}).
\end{aligned}$$

B.3 Estimating $Pr(N'_{ijk}) = Pr(K' \wedge J')$.

We rewrite $Pr(K' \wedge J')$ as follows

$$\begin{aligned} Pr(K' \wedge J') = Pr(J')\sum_{i' \neq i} Pr(L_{i'}|J')Pr(K'|L_{i'}) \geq \\ Pr(J')\sum_{i' \neq i} Pr(L_{i'}|J')Pr(K'_{ii'}|L_{i'}) \geq \\ Pr(J')\sum_{i' \neq i} Pr(L_{i'}|J')Pr(K'_{ii'}). \end{aligned}$$

As we noticed in the previous subsections the independence of different K'_j for $j \in V_1$ and the property of the random variable s_{ijk} from Lemma 3 imply

$$\begin{aligned} Pr(K_{ii'}) = (1 - e^{-s_{ijk_2}(1-b_{jk_2}-f_{i'jk_2}-f_{ijk_2}}) \geq \\ \frac{1-b_{jk_2}-f_{i'jk_2}-f_{ijk_2}}{(1-b_{jk_2}-f_{ijk_2})}\left(1-e^{-s_{ijk_2}(1-b_{jk_2}-f_{ijk_2}}\right) = \\ \frac{1-b_{jk_2}-f_{i'jk_2}-f_{ijk_2}}{(1-b_{jk_2}-f_{ijk_2})} Pr(K'), \end{aligned}$$

where the inequality follows from the Lemma 15 with $A = s_{ijk_2}(1-b_{jk_2}-f_{i'jk_2}-f_{ijk_2})$ and $B = s_{ijk_2}(1-b_{jk_2}-f_{ijk_2})$. By Lemma 4 we have $Pr(L_{i'}|J') = \frac{f_{i'j_1k}}{1-b_{j_1k}-f_{ij_1k}}$.

Combining these two facts we derive

$$\begin{aligned} Pr(N'_{ijk}) = Pr(K' \wedge J') \quad &\geq \\ Pr(J')Pr(K')\left(\sum_{i' \neq i} \frac{f_{i'j_1k}}{(1-b_{j_1k}-f_{ij_1k})} \cdot \frac{1-b_{jk_2}-f_{i'jk_2}-f_{ijk_2}}{(1-b_{jk_2}-f_{ijk_2})}\right) \quad &= \\ Pr(J')Pr(K')\left(\frac{\sum_{i' \neq i}\left[f_{i'j_1k}(1-b_{jk_2}-f_{ijk_2}) - f_{i'j_1k}f_{i'jk_2}\right]}{(1-b_{j_1k}-f_{ij_1k})(1-b_{jk_2}-f_{ijk_2})}\right) \quad &= \\ Pr(J')Pr(K')\left(1 - \frac{\sum_{i' \neq i} f_{i'j_1k}f_{i'jk_2}}{(1-b_{j_1k}-f_{ij_1k})(1-b_{jk_2}-f_{ijk_2})}\right) \quad &\geq \end{aligned}$$

Applying the Cauchy-Schwartz inequality we continue

$$\begin{aligned} P(J')P(K')\left(1 - \frac{\sqrt{(\sum_{i' \neq i} f^2_{i'j_1k})(\sum_{i' \neq i} f^2_{i'jk_2})}}{(1-b_{j_1k}-f_{ij_1k})(1-b_{jk_2}-f_{ijk_2})}\right) \quad &= \\ P(J')P(K')\left(1 - \frac{\sqrt{(\sum_{i' \neq i} f^2_{i'j_1k})(\sum_{i' \neq i} f^2_{i'jk_2})}}{(\sum_{i' \neq i} f_{i'j_1k})(\sum_{i' \neq i} f_{i'jk_2}))}\right) \quad &\geq \end{aligned}$$

We continue by applying the Lemma 14 twice with $a_{i'} = y_{i'j_1k}$, $b_{i'} = (1 - ay_{i'j_1k})$, $a_{i'}b_{i'} = f_{i'j_1k}$ and $a_{i'} = y_{i'jk_2}$, $b_{i'} = (1 - ay_{i'jk_2})$ and $a_{i'}b_{i'} = f_{i'jk_2}$. Note that if $y_1 \leq y_2$ then $y_1 - ay_1^2 \leq y_2 - ay_2^2$ for

$a \le 0.5$ and therefore the ordering of $a_i b_i$ is consistent with the ordering of a_i.

$$P(J')P(K')\left(1-\frac{\sqrt{(\sum_{i'\neq i} y^2_{i'j_1k})(\sum_{i'\neq i} y^2_{i'jk_2})}}{(\sum_{i'\neq i} y_{i'j_1k})(\sum_{i'\neq i} y_{i'jk_2}))}\right) \quad =$$

$$P(J')P(K')\left(1-\frac{\sqrt{(b_{j_1k}/a-y^2_{ij_1k})(b_{jk_2}/a-y^2_{ijk_2})}}{(1-y_{ij_1k})(1-y_{ijk_2})}\right).$$

The Lemma 13 follows.

B.4 Putting it all together

Recall $w_1 = u(z_{ij_1k})$ and $w_2 = u(z_{ijk_2})$. We proved that for four fixed parameters $b_{j_1k}, b_{jk_2}, y_{ij_1k}, y_{ijk_2} \in (0,1]$ such that $w_1 \le w_2$ we have an estimate

$$Pr(N_{ijk}) \ge \int_{w_2}^{1} du_i + \int_{w_1}^{w_2}\left(1-e^{-s_{ijk_2}(1-b_{jk_2}-y_{ijk_2}+ay^2_{ijk_2})}\right)du_i \quad +$$
$$\int_0^{w_1}\left(1-e^{-s_{ij_1k}(1-b_{j_1k}-y_{ij_1k}+ay^2_{ij_1k})}\right)\left(1-e^{-s_{ijk_2}(1-b_{jk_2}-y'_{ijk_2}+ay^2_{ij_1k})}\right)$$
$$\left(1-\frac{\sqrt{(b_{j_1k}/a-y^2_{ij_1k})(b_{jk_2}/a-y^2_{ijk_2})}}{(1-y_{ij_1k})(1-y_{ijk_2})}du_i\right) \tag{17}$$

The above now is a well-behaving (derivatives do not get large) function of four variables, $b_{j_1k}, b_{jk_2}, y_{ij_1k}, y_{ijk_2}$ for $y \in (0,1)$ and $y^2 \le b/a \le 1$, the minimum of which can be evaluated numerically with good accuracy. Using Mathematica to minimize, we find that the minimum of (17) is achieved at $y_{ijk_2} = y_{ij_1k} = 0, b_{jk_2}/a \approx 0.32, b_{j_1k}/a \approx 0.43$, and has a value of 0.1508. Moreover, to verify the correctness of the found solution we wrote our own solver to estimate (17) by discretizing the solution space and enumerating over all possible solutions. Our solver found the same solution as Mathematica.