# A note on the integrality gap of the configuration LP for restricted Santa Claus*

Klaus Jansen        Lars Rohwedder

Christian-Albrechts-Universität, Kiel, Germany
{kj, lro}@informatik.uni-kiel.de

### Abstract

In the restricted Santa Claus problem we are given resources $\mathcal{R}$ and players $\mathcal{P}$. Every resource $j \in \mathcal{R}$ has a value $v_j$ and every player $i$ desires a set $\mathcal{R}(i)$ of resources. We are interested in distributing the resources to players that desire them. The quality of a solution is measured by the least happy player, i.e., the lowest sum of resource values. This value should be maximized. The local search algorithm by Asadpour et al. [2] and its connection to the configuration LP has proved itself to be a very influential technique for this and related problems.

In the original proof, a local search was used to obtain a bound of 4 for the ratio of the fractional to the integral optimum of the configuration LP (integrality gap). This bound is non-constructive since the local search has not been shown to terminate in polynomial time. On the negative side, the worst instance known has an integrality gap of 2.

Although much progress was made in this area, neither bound has been improved since. We present a better analysis that shows the integrality gap is not worse than $3 + 5/6 \approx 3.8333$.

## 1 Introduction

A generalization of the problem we consider goes back to Bansal and Srividenko [3]. In the Santa Claus problem there are players $\mathcal{P}$ and resources $\mathcal{R}$. Every resource $j$ has a value $v_{ij} \geq 0$ for player $i$. The goal is to find an assignment $\sigma : \mathcal{R} \to \mathcal{P}$ such that $\min_{i \in \mathcal{P}} \sum_{j \in \sigma^{-1}(i)} v_{ij}$ is maximized.

In the restricted variant, we consider only values $v_{ij} \in \{0, v_j\}$ where $v_j > 0$ is a value depending only on the resource. This can also be seen as each player desiring a subset $\mathcal{R}(i)$ of resources which have a value of $v_j$ for him, whereas other resources cannot be assigned to him.

For the restricted Santa Claus problem there exists a strong LP relaxation, the configuration LP. The proof that this has a small integrality gap (see [2]) is not trivial. It works by defining an exponential time local search algorithm which is guaranteed to return an integral solution of value not much less than the fractional optimum. This technique has since been used in other problems, like the minimization of the makespan [7, 4]. Significant research has also gone into making the proof constructive [6, 1, 5]. Yet, no improvement of the bound of 4 on the integrality gap has been found. We show that the original analysis is not tight and can be improved to $3 + 5/6 \approx 3.8333$.

### 1.1 Configuration LP

the configuration LP is an exponential size LP relaxation, but it can be approximated in polynomial time with a rate of $(1 + \epsilon)$ for every $\epsilon > 0$ [3]. For every player $i$ and every value $\tau$ let

$$\mathcal{C}(i, \tau) = \{S \subseteq \mathcal{R}(i) : v(S) \geq \tau\}.$$

These are the configurations for player $i$ and value $\tau$. They are a selection of resources that have value at least $\tau$ and are desired by player $i$. The optimum OPT* of the configuration LP is the highest $\tau$ such that

---

the following linear program is feasible.

---

**Primal of the configuration LP for restricted Santa Claus**

$$\sum_{C \in \mathcal{C}(i,\tau)} x_{i,C} \geq 1 \qquad\qquad \forall i \in \mathcal{P}$$

$$\sum_{i \in \mathcal{P}} \sum_{C \in \mathcal{C}(i,\tau):j \in C} x_{i,C} \leq 1 \qquad\qquad \forall j \in \mathcal{R}$$

$$x_{i,C} \geq 0$$

---

**Dual of the configuration LP for restricted Santa Claus**

$$\max \sum_{i \in \mathcal{P}} y_i - \sum_{j \in \mathcal{R}} z_j$$

$$\sum_{j \in C} z_j \geq y_i \qquad\qquad \forall i \in \mathcal{P}, C \in \mathcal{C}(i,\tau)$$

$$y_i, z_j \geq 0$$

---

We derive the following condition from duality:

**Theorem 1.** *Let $y \in \mathbb{R}^{\mathcal{P}}_{\geq 0}$ and $z \in \mathbb{R}^{\mathcal{R}}_{\geq 0}$ such that $\sum_{i \in \mathcal{P}} y_i > \sum_{j \in \mathcal{R}} z_j$ and for every $i \in \mathcal{P}$ and $C \in \mathcal{C}(i,\tau)$ it holds that $\sum_{j \in C} z_j \geq y_i$, then $\mathrm{OPT}^* < \tau$.*

It is easy to see that if such a solution $y, z$ exists, then every component can be scaled by a constant to obtain a feasible solution greater than any given value. Hence, the dual must be unbounded and therefore the primal must be infeasible.

# 2 Algorithm

We consider the local search algorithm from [2]. It is the same algorithm with a slightly different presentation that is inspired by [6]. Throughout this section we will denote by $\alpha = 3 + 5/6$ the bound on integrality we want to prove.

We model our problem as a hypergraph matching problem: There are vertices for all players and all resources and the hyperedges $\mathcal{H}$ each consist of exactly one player $i$ and a set of resources $C \subseteq \mathcal{R}(i)$ where $v(C) \geq \mathrm{OPT}^*/\alpha$. However, we restrict $\mathcal{H}$ to edges that are minimal, that is to say $v(C') < \mathrm{OPT}^*/\alpha$ for all $C' \subset C$. It is easy to see that a matching (a set of non-overlapping edges) such that every player is in one matching edge corresponds to a solution of value $\mathrm{OPT}^*/\alpha$.

For a set of edges $F$ we write $F_{\mathcal{P}}$ as the set of players in these edges and $F_{\mathcal{R}}$ as the resources in the edges. The algorithm maintains a partial matching $M$ and extends it one player at a time. After $|\mathcal{P}|$ many calls to the algorithm the desired matching is found. Two types of edges play a crucial role in the algorithm: An ordered list $A = \{e_1^A, \ldots, e_\ell^A\} \subseteq \mathcal{H} \setminus M$ (the addable edges) and sets $B_M(e_1^A)$, $B_M(e_2^A)$, $\ldots$, $B_M(e_\ell^A) \subseteq M$ (the blocking edges for each addable edge). An addable edge is a edge that the algorithm hopes to add to $M$ - either to cover the new player or to free the player of a blocking edge. A blocking edge is an edge in $M$ that conflicts with an addable edge, i.e., that has a non-empty overlap with an addable edge. For each addable edge $e_k^A$ we define the blocking edges $B_M(e_k^A)$ as $\{e' \in M : e'_{\mathcal{R}} \cap (e_k^A)_{\mathcal{R}} \neq \emptyset\}$. From the definition of the algorithm it will be clear that $B_M(e_k^A) \cap B_M(e_{k'}^A) = \emptyset$ for $k \neq k'$. We write $B_M(A)$ for $\bigcup_{e \in A} B_M(e)$.

## 2.1 Detailed description of the algorithm

In each iteration the algorithm first adds a new addable edge that does not overlap in resources with any existing addable or blocking edge. Then it consecutively swaps addable edges that are not blocked for the blocking edge they are supposed to free. Also, addable/blocking edges added at a later time are removed, since they might be obsolete. The swap does not create new blocking edges, since the new matching

edge does not overlap with addable edges. Also, by adding only addable edges that do not overlap with the previous addable/blocking edges, the resources of addable edges are disjoint and the resources of each blocking edge overlap only with one addable edge. The blocking edges $B_M(e_k^A)$ and $B_M(e_{k'}^A)$ for two addable edges with $k < k'$ must be disjoint, because otherwise $e_{k'}^A$ could not have been added in the first place.

---

**Input:** Partial matching $M$ and unmatched player $i_0$
**Result:** Partial matching $M'$ with $M'_{\mathcal{P}} = M_{\mathcal{P}} \cup \{i_0\}$
$\ell \leftarrow 0$ ;
**loop**
    $\ell \leftarrow \ell + 1$ ;
    let $e_\ell^A \in \mathcal{H} \setminus M$ with $(e_\ell^A)_{\mathcal{R}} \cap (A \cup B_M(A))_{\mathcal{R}} = \emptyset$ and $(e_\ell^A)_{\mathcal{P}} \in B_M(A)_{\mathcal{P}} \cup \{i_0\}$ ;
    `// existence of` $e_\ell^A$ `is proved in the analysis`
    $A \leftarrow A \cup \{e_\ell^A\}$ ; `//` $e_\ell^A$ `is added as the last addable edge`
    **while** $B_M(e_\ell^A) = \emptyset$ **do**
        **if** *there is an edge* $e' \in B_M(A)$ *with* $e'_{\mathcal{P}} = (e_\ell^A)_{\mathcal{P}}$ `// unambiguous since` $M$ `is matching`
        **then**
            let $e_k^A \in A$ such that $e' \in B_M(e_k^A)$ ; `// unambiguous`
            $M \leftarrow M \setminus \{e'\} \cup \{e_\ell^A\}$ ; `// swap` $e'$ `for` $e_\ell^A$
            $A \leftarrow \{e_1^A, \ldots, e_k^A\}; \ell \leftarrow k$ ; `// Forget` $e_{k+1}^A, \ldots, e_\ell^A$
        **else**
            $M \leftarrow M \cup \{e_\ell^A\}$ ; `//` $(e_\ell^A)_{\mathcal{P}} = i_0$
            **return** $M$ ;
        **end**
    **end**
**end**

**Algorithm 1:** Local search for restricted SANTA CLAUS

# 3 Analysis

**Theorem 2** ([2])**.** *The algorithm terminates after at most* $2^{|M|-1}$ *many iterations of the main loop.*

*Proof.* Consider the signature vector

$$s(A) = \{|B_M(e_1^A)|, |B_M(e_2^A)|, \ldots, |B_M(e_\ell^A)|, \infty\}.$$

This vector decreases after every iteration of the main loop: If the inner loop is never executed, then the last component is replaced by a finite value. Hence, assume the inner while loop is executed at least once. Let $\ell'$ be the cardinality of $A$ after the last execution of the inner loop. Then $|B(e_{\ell'}^A)|$ has decreased by the swap operation. It follows that the signature vectors in each iteration of the main loop are pairwise different.

The number of signature vectors can be trivially bounded by $|M|^{|M|}$. A clever idea from [2] even gives a bound of $2^{|M|-1}$: We have that $\sum_{k=1}^{\ell} |B_M(e_k^A)| = |B_M(A)| \leq |M|$ and $|B_M(e_k^A)| \geq 1$ for all $k$. There is an bijection between signature vectors and possibilities of placing separators on a line of $|M|$ elements. $|B_M(e_k^A)|$ is the number of elements between the $(k-1)$-th and $k$-th separator. The number of possibilities of placing separators between the $|M|$ elements is the number of subsets of $|M|-1$ elements, i.e. $2^{|M|-1}$. $\quad\square$

Clearly the inner loop also terminates after finitely many iterations, since in each iteration $\ell$ is decreased.

**Theorem 3.** *If the configuration LP is feasible and there will an edge that can be added to $A$ as long as $i_0$ is not matched.*

*Proof.* In the proof we use the constant $\beta = 1 + 8/15 \approx 1.53333$, that has been chosen so as to minimize $\alpha$. Assume toward contradiction that edge remains that can be added to $A$, but $i_0$ is not covered. In the remainder of the proof we will write $B$ instead of $B_M(A)$ and $B(e)$ instead of $B_M(e)$, since $M$ and $A$ are

constant throughout the proof. Define

$$y_i = \begin{cases} 1 & \text{if } i \in B_{\mathcal{P}} \cup \{i_0\}, \\ 0 & \text{otherwise.} \end{cases}$$

$$z_j = \begin{cases} 1 & \text{if } v_j \geq \text{OPT}^*/\alpha \text{ and } j \in A_{\mathcal{R}} \cup B_{\mathcal{R}}, \\ \min\{1/3, \beta \cdot v_j/\text{OPT}^*\} & \text{if } v_j < \text{OPT}^*/\alpha \text{ and } j \in A_{\mathcal{R}} \cup B_{\mathcal{R}}, \\ 0 & \text{otherwise.} \end{cases}$$

We refer to the resources $j$ where $v_j \geq \text{OPT}^*/\alpha$ as fat resources and to others as thin resources. Note that by minimality of edges in $\mathcal{H}$, each edge containing a fat resource does not contain any other resources. We call these the fat edges. Likewise, edges that contain only thin resources are referred to as thin edges.

**Claim 1.** $(y, z)$ *is a feasible solution for the dual of the configuration LP.*

**Claim 2.** $(y, z)$ *has a negative objective value, that is to say* $\sum_{j \in \mathcal{R}} z_j < \sum_{i \in \mathcal{P}} y_i$.

By Theorem 1 this implies that the configuration LP is infeasible for $\text{OPT}^*$. A contradiction. $\qquad\square$

*Proof of Claim 1.* Let $i \in \mathcal{P}$ and $C \in \mathcal{C}(i, \text{OPT}^*)$. We need to show that $y_i \leq z(C)$. If $y_i = 0$ or $C$ contains a fat resource, this is trivial. Hence, assume w.l.o.g. that $C$ consists solely of thin resources and $y_i = 1$.

Since no addable edge for $i$ remains, $v(C \setminus (A_{\mathcal{R}} \cup B_{\mathcal{R}})) < \text{OPT}^*/\alpha$. Let $S \subseteq C$ be the resources $j \in C$ which have $z_j = 1/3$.

**Case 1:** $|S| = 3$. Then $z(C) \geq z(S) \geq 3 \cdot 1/3 = 1$.

**Case 2:** $|S| \leq 2$. Define $C' := (C \cap (A_{\mathcal{R}} \cup B_{\mathcal{R}})) \setminus S$. Then

$$v(C') > v(C) - v(C \setminus (A_{\mathcal{R}} \cup B_{\mathcal{R}})) - v(S) \geq \text{OPT}^* - \text{OPT}^*/\alpha - v(S).$$

Therefore,

$$z(C) \geq 1/3 \cdot |S| + \beta/\text{OPT}^* \cdot v(C') > 1/3|S| + \beta(1 - 1/\alpha - v(S)/\text{OPT}^*)$$
$$\geq 1/3|S| + \beta(1 - (|S| + 1)/\alpha) =: (*).$$

Since $\beta/\alpha = 0.4 > 1/3$, the coefficient of $|S|$ in $(*)$ is negative and thus we can substitute $|S|$ for its upper bound, i.e., 2. By inserting the values of $\alpha$ and $\beta$ we get,

$$(*) \geq 2/3 + \beta(1 - 3/\alpha) = 1. \qquad\square$$

*Proof of Claim 2.* We write in the following $F^f$ ($F^t$) for the fat edges (thin edges, respectively) in a set of edges $F$. First note that every fat edge with positive $z$ value must be in a fat blocking edge and therefore

$$\sum_{j \in \mathcal{R}^f} z_j \leq |B^f|.$$

Now consider thin addable edges. Since every addable edge is blocked, $|B(e)| \geq 1$ for every $e \in A^t$. We now proceed to show that for every $e \in A^t$

$$z(e_{\mathcal{R}} \cup B(e)_{\mathcal{R}}) \leq |B(e)|.$$

Note that for every thin resource $j$, we have $z_j \leq \beta/\text{OPT}^* \cdot v_j$. By minimality of edges in $\mathcal{H}$, it holds that $v(e_{\mathcal{R}}) \leq 2\text{OPT}^*/\alpha$ (each element in $e_{\mathcal{R}}$ has value at most $\text{OPT}^*/\alpha$). Also $v(e'_{\mathcal{R}} \setminus e_{\mathcal{R}}) \leq \text{OPT}^*/\alpha$ for each $e' \in B(e)$, since the intersection of $e_{\mathcal{R}}$ and $e'_{\mathcal{R}}$ is non-empty. If $|B(e)| \geq 2$, this implies

$$z(e_{\mathcal{R}} \cup B(e)_{\mathcal{R}}) \leq \beta/\text{OPT}^* \cdot (v(e_{\mathcal{R}}) + v(B(e)_{\mathcal{R}} \setminus e_{\mathcal{R}}))$$
$$\leq \beta \cdot (2/\alpha + |B(e)| \cdot 1/\alpha) \leq |B(e)| \cdot \beta \cdot 2/\alpha = 0.8 \cdot |B(e)| < |B(e)|.$$

Assume in the following that $|B(e)| = 1$. Let $v_{\min}$ be the value of the smallest element in $e_{\mathcal{R}} \cup B(e)_{\mathcal{R}}$. Then $v(e_{\mathcal{R}} \cup B(e)_{\mathcal{R}}) \leq 2\mathrm{OPT}^*/\alpha + v_{\min}$: If the smallest element is in $e_{\mathcal{R}}$, then

$$v(e_{\mathcal{R}} \cup B(e)_{\mathcal{R}}) \leq \underbrace{\mathrm{OPT}^*/\alpha + v_{\min}}_{\geq v(e_{\mathcal{R}})} + v(B(e)_{\mathcal{R}} \setminus e_{\mathcal{R}}) \leq 2\mathrm{OPT}^*/\alpha + v_{\min}.$$

The same argument (swapping the role of $e$ and $B(e)$) holds, if the smallest element is in $B(e)$. Therefore, if $|B(e)| = 1$ and $v_{\min} \leq 1/2 \cdot \mathrm{OPT}^*/\alpha$,

$$z(e_{\mathcal{R}} \cup B(e)_{\mathcal{R}}) \leq \beta \cdot (2/\alpha + v_{\min}) \leq \beta \cdot 5/2 \cdot 1/\alpha = 1 = |B(e)|.$$

If $|B(e)| = 1$ and $v_{\min} > 1/2 \cdot \mathrm{OPT}^*/\alpha$, then $B(e)_{\mathcal{R}} \setminus e_{\mathcal{R}}, B(e)_{\mathcal{R}} \cap e_{\mathcal{R}}$, and $e_{\mathcal{R}} \setminus B(e)_{\mathcal{R}}$ have at least one and by minimality of edges at most one element. Since the $z$ value of each thin edge is at most $1/3$,

$$z(e_{\mathcal{R}} \cup B(e)_{\mathcal{R}}) = z(e_{\mathcal{R}} \setminus B(e)_{\mathcal{R}}) + z(e_{\mathcal{R}} \cap B(e)_{\mathcal{R}}) + z(B(e)_{\mathcal{R}} \setminus e_{\mathcal{R}}) \leq 3 \cdot 1/3 = 1 = |B(e)|.$$

We conclude that

$$\sum_{i \in \mathcal{P}} y_i = |B^f| + |B^t| + 1 > |B^f| + \sum_{e \in A^t} |B(e)| \geq \sum_{j \in \mathcal{R}^f} z_j + \sum_{e \in A^t} z(e_{\mathcal{R}} \cup B(e)_{\mathcal{R}}) = \sum_{j \in \mathcal{R}} z_j. \qquad \square$$

# References

[1] Chidambaram Annamalai, Christos Kalaitzis, and Ola Svensson. Combinatorial algorithm for restricted max-min fair allocation. *ACM Trans. Algorithms*, 13(3):37:1–37:28, 2017.

[2] Arash Asadpour, Uriel Feige, and Amin Saberi. Santa claus meets hypergraph matchings. *ACM Trans. Algorithms*, 8(3):24:1–24:9, 2012. See Asadpour's homepage for the bound of 4 instead of 5 as in the paper.

[3] Nikhil Bansal and Maxim Sviridenko. The santa claus problem. In Jon M. Kleinberg, editor, *Proceedings of the 38th Annual ACM Symposium on Theory of Computing, Seattle, WA, USA, May 21-23, 2006*, pages 31–40. ACM, 2006.

[4] Klaus Jansen and Lars Rohwedder. On the configuration-lp of the restricted assignment problem. In Philip N. Klein, editor, *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2017, Barcelona, Spain, Hotel Porta Fira, January 16-19*, pages 2670–2678. SIAM, 2017.

[5] Klaus Jansen and Lars Rohwedder. A quasi-polynomial approximation for the restricted assignment problem. In Friedrich Eisenbrand and Jochen Könemann, editors, *Integer Programming and Combinatorial Optimization - 19th International Conference, IPCO 2017, Waterloo, ON, Canada, June 26-28, 2017, Proceedings*, volume 10328 of *Lecture Notes in Computer Science*, pages 305–316. Springer, 2017.

[6] Lukás Polácek and Ola Svensson. Quasi-polynomial local search for restricted max-min fair allocation. *ACM Trans. Algorithms*, 12(2):13:1–13:13, 2016.

[7] Ola Svensson. Santa claus schedules jobs on unrelated machines. *SIAM J. Comput.*, 41(5):1318–1341, 2012.