Temporary Staffing at Good'OldGood's

FILIPE CAMPOS, FRANCISCO CERQUEIRA, and VERONIJA KODOVSKA

1 INTRODUCTION

The assignment of temporary sales assistants to specific positions during auctions poses a significant challenge for Good'oldgood's, a luxury goods and artwork auction house. With the need to provide exceptional service to their clients, the efficient allocation of temporary staff becomes crucial.

To address this challenge, we propose a mixed integer programming (MIP) solution. By formulating the assignment problem as an MIP model, Good'oldgood's can achieve a proper allocation of temporary staff, ensuring a smooth and efficient operation of the auctions.

The solution, developed using IBM Cplex [1], considers various factors, including the availability, skill sets, and experience of the temporary assistants.

This report begins by addressing the explicit requirements outlined in the problem description (Section 2). Subsequently, we discuss the assumptions made throughout the study (Section 3). To provide a comprehensive understanding of the formulated problems, we present a formal description of the assignment problem (Section 4) and the replacement problem (Section 5). The report then delves into the motivations and justifications behind the choices made (Section 6). Furthermore, we highlight some of the achieved results (Section 7). Finally, the report concludes with a summary of findings and conclusions (Section 8).

2 REQUIREMENTS

We collected the following requirements from the problem statement:

- **R.1** The tool must be capable of assigning candidates to each position for all of the days of the event (From here on out we will refer to this problem as "Assignment problem").
- **R.2** Each candidate can only be hired if available on that specific day.
- **R.3** Each candidate can only be hired for a single position each day.
- **R.4** Each position must have exactly the number of workers specified in its requirements.
- **R.5** There is a waiting list for candidates that could not be placed on the regular positions.
- **R.6** The solution must take five criteria into consideration: past experience, availability, presentation skills, language skills and traveling time and distance.
- **R.7** Not all criteria have the same importance across all the solutions
- **R.8** The solution quality must be similar to that of one created by a human resources staff.
- **R.9** The proposed method for the assignment problem cannot take longer than half a day, the time it takes for a human to perform the allocations manually. Ideally it should be faster.
- **R.10** There is a bound on the maximum number of staff that can be hired (See **A.5**).
- **R.11** There is a limit to the number of consecutive working days for each worker (See **A.6**).
- **R.12** For a given day and a list of missing staff the tool must be capable of finding the optimal replacements within a short time-frame (From here on out we will refer to this problem as "Replacement problem").
- **R.13** At the very least, the Replacement problem must be capable of handling the worst case scenario where 9% of the candidates are missing in the same day, but the average value will be 4%.

3 ASSUMPTIONS

- **A.1** The capabilities of a candidate for a given position can be calculated using a linear combination of their experience, language skills and presentation skills. Additionally, the manager should be capable of determining what weight should be attributed to each respective component.
- **A.2** The candidates have no positions preferences. They are willing to work in any position assigned to them by the model.
- **A.3** The regular position assignments always have priority over the waiting list assignments.
- **A.4** The manager can determine for which positions language and presentation skills are important, as shown in Table 9. These must be binary for simplicity, since determining different weights for each of the individual positions would be time consuming.
- **A.5** The manager can determine what the ideal number of staff members and what the maximum number of staff is. The waiting list position does not count toward this total. Since on previous years the number of workers varied from 200 to 300, we propose 300 as a default value.
- **A.6** The manager can determine how many consecutive days a staff can work.
- **A.7** Only the members of the waiting list can be called to replace missing staff.
- **A.8** Following assumption **A.7**, the travel time and distance value is only important for the waiting list candidates. Regular staff can plan ahead their transportation to the event premises while the waiting list members will need to be capable of reaching the event on short notice.
- **A.9** In the event of missing staff members, the other staff members will continue to work on their previously assigned positions. That is, there cannot be a position re-arrangement on the day of the event other than placing waiting list members as substitutes for the missing staff. This condition was added for two reasons – first because those workers must already have some contract which specifies what job they will be doing and second, because large last minute changes can destabilize the event organization.

Authors' address: Filipe Campos, up201905609@edu.fe.up.pt; Francisco Cerqueira, up201905337@edu.fe.up.pt; Veronija Kodovska, up202211733@edu.fe.up.pt.

A.10 Candidates on the waiting list do not count toward the maximum consecutive number of working days. If this was not the case it would be impossible to know on which days the candidate would actually work so having someone on the waiting list for the 9 days of the event would be invalid due to the number of consecutive days, while in actuality the staff would probably not be hired every day.

4 ASSIGNMENT PROBLEM

4.1 Formulation

We will now formally define the problem. Consider a set of positions P, a set of days D and a set of candidates C.

4.1.1 Parameters.

The proposed solution has a set of parameters that should be fine-tuned by the manager, as mentioned in assumptions **A.1**, **A.5** and **A.6**.

To summarize the capabilities of a given candidate for a given position we use the following parameters:

- $w_{\text{experience}}$ - Weight of the experience.
- w_{language} - Weight of the candidate's language skills.
- $w_{\text{presentation}}$ - Weight of the candidate's presentation skills.

which must have a combined sum value of 1.

To choose the best candidates for the waiting list, similarly to the previous parameters, the following values must be set:

- $w_{\text{wl-capability}}$ - The weight of the candidate's capabilities.
- $w_{\text{wl-travel}}$ - Weight of the candidate's travel time and distance to the event venue.

- $p_{\text{max-consecutive-days}}$ - Maximum number of days a candidate can work.
- $p_{\text{target-num-workers}}$ - The number of workers that can be hired without a penalty in the objective function.
- $p_{\text{maximum-num-workers}}$ - The hard limit on the number of workers that can be hired.
- $p_{\text{extra-worker-penalty}}$ - The penalty in the objective function for each worker over the value of $p_{\text{maximum-num-workers}}$.

The default values used in our solution are present in Table 1.

Parameter	**Value**
$w_{\text{experience}}$	0.60
$w_{\text{presentation}}$	0.20
w_{language}	0.20
$w_{\text{wl-capability}}$	0.65
$w_{\text{wl-travel}}$	0.35
$p_{\text{max-consecutive-days}}$	7
$p_{\text{target-num-workers}}$	250
$p_{\text{maximum-num-workers}}$	300
$p_{\text{extra-worker-penalty}}$	0.02

Table 1. Default parameter values

4.1.2 Data.

- The availability of a candidate c for day d is represented by

$$a_{cd} = \begin{cases} 1, & \text{if candidate } c \text{ is available on day } d \\ 0, & \text{otherwise} \end{cases}, \ c, d \in D, C$$

.

- The number of staff required for each position p for day d is represented by $r_{pd} \in \mathbb{Z}^{\geq}, \quad d, p \in D, P$.
- The experience of each candidate in each position [1] is represented as $e_{cp} \in \{0..5\},\ c, p \in C, P$.
- The language and presentation skill values of each candidate are defined as $sk_{sc} \in \{0..5\},\ c \in C, s \in \{\text{language,presentation}\}$.
- The time and travel value of each candidate is defined as $t_c \in \{0..5\},\ c \in C$
- To identify whether or not a skill is important for a given position we define a skills match matrix

$$sm_{sp} = \begin{cases} 1, & \text{if skill } s \text{ is used in position } p \\ 0, & \text{otherwise} \end{cases}, \quad p \in P, s \in \{\text{language,presentation}\}$$

4.1.3 Decision variables.

The assignment decision variable is defined as

$$x_{dcp} = \begin{cases} 1, & \text{if on day } d \text{ candidate } c \text{ is assigned to position } p \\ 0, & \text{otherwise} \end{cases}, \quad d, c, p \in D, C, P$$

For the soft constraint, the following $s_{\text{num-workers}} \in \mathbb{Z}^{\geq}$ slack variable was added.

4.1.4 Objective function.

Assumption **A.3** states that regular assignments always take priority over waiting list assignments. Due to this condition, we decide the use a lexicographic objective function [2] where, firstly, the optimal assignments will be determined for the stated positions, and secondly the waiting list choices will be optimized. This ensures that under no circumstance the regular positions will deteriorate in favor of a better waiting list choice.

Therefore there are two functions that should be maximized, Z_1 for the regular positions and Z_2 for the waiting list.

$$\text{lex max} \quad Z_1, Z_2$$

$\mathbf{Z_1}$ function:

As explained in assumption **A.1**, we define a capability value for each of the candidates which is a linear combination of their experience and skills. So for position p (for simplicity of notation we are omit that $p \in P \setminus \{\text{"}WaitingList\text{"}\}$) the capability of candidate c working in it is obtained as:

[1] In the original data the experience is shown for each **position type**, we assume that the conversion to a specific **position** was performed according to Table 10

$$
\begin{aligned}
capability_{pc} = (&\\
& w_{\text{experience}} \cdot e_{cp} & + \\
& w_{\text{language}} \cdot sk_{\text{language},c} \cdot sm_{\text{language},p} & + \\
& w_{\text{presentation}} \cdot sk_{\text{presentation},c} \cdot sm_{\text{presentation},p} & + \\
& (1 - sm_{\text{language},p}) \cdot w_{\text{language}} \cdot e_{cp} & + \\
& (1 - sm_{\text{presentation},p}) \cdot w_{\text{presentation}} \cdot e_{cp} & \\
&),\ \text{for } p, c \in P, C &
\end{aligned}
$$

Note that this value should always be in the integer interval $\{0..5\}$ because the sum of all the weights must add up to 1 and both the skills and experience values are within this interval. The two last terms of the expression signify that, if the skills are not being used in that specific position, then their "weight" is transferred to the experience component. If a position does not require any skill and a candidate has 5 experience then the capability value should be 5 and not $w_{\text{experience}} \cdot 5$.

Now to summarize the overall capabilities of our solution we define a value *capability_value* which takes into account all the allocations in the solution

$$capability\_value = \sum_{d \in D} \sum_{p \in P : r_{pd} \geq 0} \sum_{c \in C} x_{dcp} \cdot c_{pc}$$

To make this value more interpretable we normalize it so it also fits within the interval $\{0..5\}$.

$$capability\_value_{\text{normalized}} = \frac{capability\_value}{|D| \cdot |P|}$$

.

Other than this value, we will also add the penalties for using slack variables to the function. Therefore Z1 is defined as:

$$
\begin{aligned}
\mathbf{Z_1} = capability\_value_{\text{normalized}} & \quad - \\
s_{\text{num-workers}} \cdot p_{\text{extra-worker-penalty}} &
\end{aligned}
$$

$\mathbf{Z_2}$ function:

There are two main considerations we took into account to improve the waiting list allocations. Firstly, the chosen candidates must be capable in a large range of positions, so that any missing staff can be substituted by someone with a good skill set for the job. Secondly, the travel time it takes for the candidate to reach the venue is also important because they will be contacted on very short notice.

The average capability across all positions can be calculated using the expression:

$$
\begin{aligned}
avg_{\text{wl\_capability}} = \sum_{d \in D} (&\\
& \frac{\sum_{p \in P} \sum_{c \in C} x_{d,c,\text{"}WaitingList\text{"}} \cdot capability_{pc}}{r_{\text{"}WaitingList\text{"},d} \cdot |P|} \\
&)/|D|
\end{aligned}
$$

To calculate the time and travel distance for the waiting list members we used the following expression:

$$
\begin{aligned}
waiting\_list\_travel = \sum_{d \in D} (&\\
& \frac{\sum_{c \in C} x_{d,c,\text{"}WaitingList\text{"}} \cdot t_c}{r_{\text{"}WaitingList\text{"},d}} \\
&)/|D|
\end{aligned}
$$

Finally both of these expressions were combined to create the $\mathbf{Z_2}$ objective function:

$$
\begin{aligned}
\mathbf{Z_2} = avg_{\text{wl\_capability}} \cdot w_{wl-capability} & \quad + \\
waiting\_list\_travel \cdot w_{wl-travel} &
\end{aligned}
$$

4.1.5 Constraints.

C.1.1 All decision variables are positive

$$x_{dcp} \geq 0,\ \forall\ d, c, p \in D, C, P$$
$$s_{\text{num-workers}} \geq 0$$

C.1.2 Each candidate can only be assigned to a single position for each day

$$\sum_{p \in P} x_{dcp} \leq 1\ \forall\ d, c \in D, C$$

C.1.3 Each positions' requirements must be satisfied

$$\sum_{c \in C} x_{dcp} = r_{pd}\ \forall\ d, p \in D, P$$

C.1.4 Candidates can only be hired if they are available on that day

$$x_{dcp} \leq a_{cd},\ \forall\ d, c, p \in D, C, P$$

C.1.5 For each staff, the maximum number of consecutive working days must not exceed a certain value.

$$mcd = p_{\text{max-consecutive-days}}$$
$$\sum_{m=n}^{m+mcd+1} (\sum_{p \in P} x_{mcp}) \leq mcd$$
$$\forall\ c \in C, n \in [0, |D| - mcd - 1]$$

If there is a sub-array of size $p_{\text{max-consecutive-days}} + 1$ whose sum is greater than $p_{\text{max-consecutive-days}}$ it means the consecutive days limit is being violated.

C.1.6 The number of workers cannot exceed the defined limit.

$$nw = \sum_{c \in C} \begin{cases} 1, & if \sum_{d \in D} \sum_{p \in P \setminus \{\text{"}WaitingList\text{"}\}} x_{dcp} \geq 1 \\ 0, & otherwise \end{cases}$$
$$nw \leq p_{\text{target-num-workers}} + s_{\text{num-workers}}$$

C.1.7 The slack of number of workers cannot exceed the maximum limit.

$$s_{\text{num-workers}} \leq p_{\text{maximum-num-workers}} - p_{\text{target-num-workers}}$$

5 REPLACEMENT PROBLEM

Given the solution of the previous assignment problem, a day and a list of missing staff we need to determine which are the ideal allocations to substitute the missing staff on that day using members of the waiting list.

5.1 Formulation

We keep most of the data from the first problem and add two new parameters, $current_{day}$ and a set $missing_{staff}$.

We also have a new data matrix, $prev_{dcp}$, which is the decision variable x from the previous solution.

$$current_{day} \in 1..9$$

$$missing_{staff} \subseteq \{c \in C \mid \exists p \in prev_{current_{day},c,p} = 1\}$$

We use only **objective function $\mathbf{Z_1}$** from the previous section, because the waiting list is no longer necessary.

5.1.1 Decision variable. Similarly to the assignment problem we also have a decision variable x defined as:

$$x_{dcp} = \begin{cases} 1, & \text{if on day } d \text{ candidate } c \text{ is assigned to position } p \\ 0, & \text{otherwise} \end{cases}, \quad d, c, p \in D, C, P$$

5.1.2 Constraints.

We kept all the constraints defined for the previous problem and added the following ones:

C.2.1 The assignments for days different than $current_{day}$ must remain unchanged

$$x_{dcp} = prev_{cdp},\ \forall\ c \in C, p \in P, d \in D \setminus \{current_{day}\}$$

C.2.2 Candidates that are already assigned cannot be moved to another position

$$x_{current_{day}cp} = 1,$$
$$\forall\ c, p \in \{x, y \in C, P \setminus \{\text{"}WaitingList\text{"}\} : prev_{current_{day}xy} = 1\}$$

C.2.3 Missing staff cannot work

$$x_{current_{day}cp} = 0,\ \forall\ c \in missing_{staff}, p \in P$$

C.2.4 Staff that are not working on this day cannot be re-assigned

$$x_{current_{day}cp} = 0,$$
$$\forall\ c, p \in \{x, y \in C \setminus missing_{staff}, P$$
$$\mid prev_{current_{day}xy} = 0 \wedge prev_{current_{day},x,\text{"}WaitingList\text{"}} = 0\}$$

6 DISCUSSION

In this section we will discuss some of the considerations that were taken during the project and were not discussed during the previous sections.

We assume that the model parameters can and will be tweaked by a domain expert to best optimize the allocations. As stated in requirement **R.1**, we need to propose a tool to solve this problem, not necessarily the best solution.

The motivation behind assumption **A.3**, is that it does not make sense to not hire someone capable because they would be an excellent person to have in the waiting list. The most capable candidates should always be hired first, only then should the waiting list be created.

The enforce a limit on the number of staff hired we proposed assumption **A.5**, we designed this as a soft constraint to allow the flexibility of hiring more specialized candidates at the cost of a small penalty to the objective function.

Similarly to the previous assumption, having a soft constraint to choose the maximum number of consecutive days would be interesting and make our tool even more flexible. Unfortunately all the implementations we tested using cplex became too complex or confusing to read, so we decided the benefit did not outweigh the cost and settled on a simpler hard constraint.

Only the waiting list candidates should have a formal agreement to work on short notice, so we added assumption **A.7**.

Finally, for the Z_2 objective function we considered using different ways of balancing the capabilities across all the positions as, for example, maximizing the minimum capability on all positions. In the end we settled on simply using the average capability across all positions because the complexity of the other methods did not seem lead to tangible benefits in the solution quality.

7 RESULTS

We tested a variety of setups so we can both validate our choices of constraints/parameters and provide the reader with different out of the box solutions. For each setup we will take a look at their advantages and disadvantages.

7.1 Default Solution

For the default parameters shown in Table 1 we obtained the metrics shown in Table 2. On average the chosen candidates have a lot experience and are skilled in the positions they have been assigned to.

Metric	**Value**
Z_1	2.8166
Z_2	2.8261
Number of workers	250
Average experience	4.6009
Average language skills	2.8506
Average presentation skills	3.0745

Table 2. Metrics for the proposed solution

In Figure 1 we can see that for all the days most staff have experience between 4 and 5. There are also a few less experienced outliers with a reasonable experience level of 2.

As shown in Figure 2 there is not much variability in the average experience across the different positions therefore every position has equally capable staff members.

In Figure 3 we see how many consecutive working days each candidate has, so we can validate that our constraint **C.1.5** is being effective.

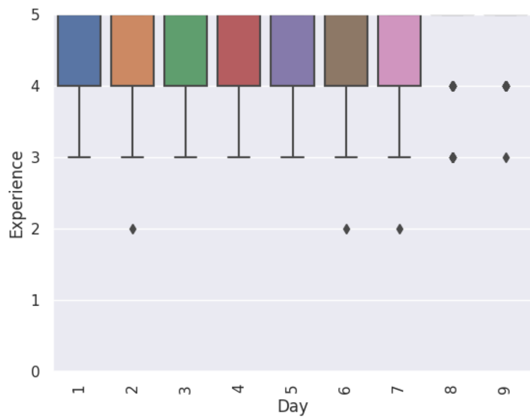

Fig. 1. Experience distribution per day

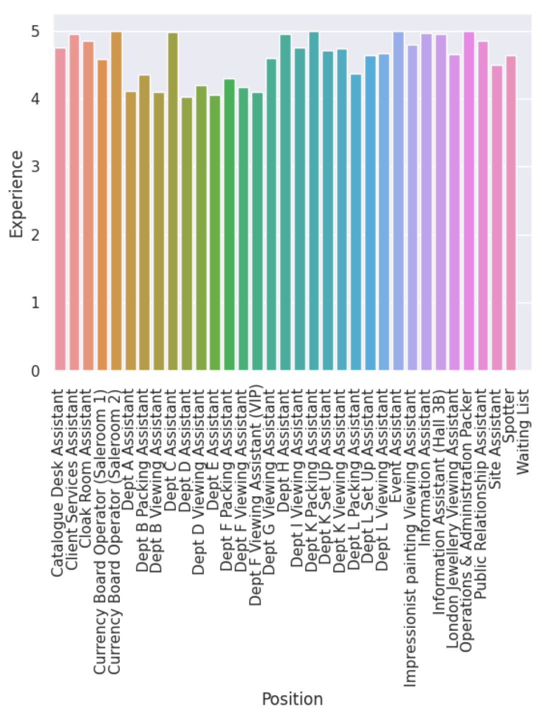

Fig. 2. Average experience per position

This solution's allocations are show in Figures 9, 10, 11 and 12. Visually we can see that there is a trend of placing the same workers on the same position across multiple days which means they are most likely ideal for the job regardless of the competition. But there is also the flexibility of occupying different positions on each of the days. Here we can also note that the constraint that limits consecutive working days is also working as expected.

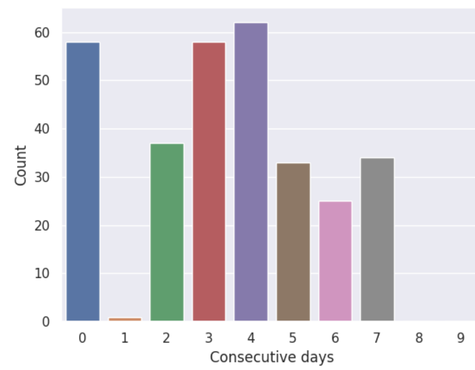

Fig. 3. Number of consecutive working days per candidate

7.2 Constraint Comparison

To test what impact our constraints had in the final solution we tested the model without them and compared them to the baseline default solution.

7.2.1 *Number of workers.*

For the model without constraints **C.1.6** and **C.1.7** we got the results shown in Table 3. Compared to the baseline solution there was a increase in some of the metrics. But this change came at the cost of an increase in the number of workers which changed from 250 to 302 exceeding the maximum of 300 workers usually hired by the company.

Metric	**Value**
Z_1	2.8695
Z_2	2.7326
Number of workers	302
Average experience	4.7003
Average language skills	2.9061
Average presentation skills	3.2107

Table 3. Metrics for the solution without number of workers constraint

7.2.2 *Unlimited consecutive days.*

For the model without constraints **C.1.5** we obtain results marginally better than the baseline solution (Table 4). But this increase comes at the cost of overworking some of the staff, as we can see in Figure 4. Whether or not this solution is viable would depend on the company's policies and labor laws of its country.

7.3 Parameter Comparison

Since our solution has many different parameters each configuration can lead to very different results. While the best person to decide

Metric	Value
Z_1	2.8291
Z_2	2.8286
Number of workers	250
Average experience	4.6350
Average language skills	2.8608
Average presentation skills	3.0335

Table 4. Metrics for the solution without consecutive days constraint

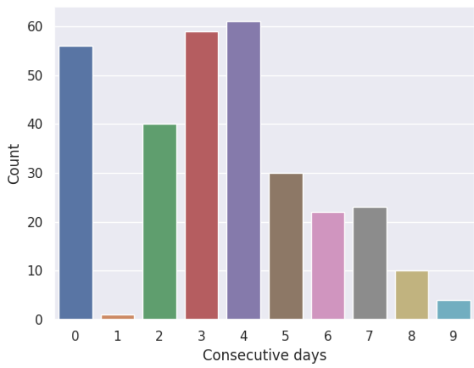

Fig. 4. Number of consecutive working days per candidate without constraint

which values are the best is the company's human resources expert we can still analyze some possible pre-defined solutions under different assumptions.

7.3.1 Skills are more important than experience.

The domain expert might value the candidates' skills over their experience, to examine this scenario we used the parameters shown in Table 5.

Parameter	Value
$w_{\text{experience}}$	0.2
$w_{\text{presentation}}$	0.4
w_{language}	0.4

Table 5. Parameter values focused on skills

In Table 6 we can see that, as expected, compared to the baseline solution there is an increase in overall skills at the cost of a decrease in the experience value. This might be more aligned with the company's goals and values since in the problem statement it is mentioned that only one third of the recruits had previously worked in the event, meaning that they are willing to hire candidates with less experience but with better skills.

7.3.2 Number of workers penalty is lower.

Metric	Value
Z_1	2.6370
Z_2	2.7552
Number of workers	250
Average experience	4.2929
Average language skills	3.2060
Average presentation skills	3.7849

Table 6. Metrics for the solution with higher skill weight

The value for the number of workers penalty in the baseline solution is 0.02, which means that if 50 workers were added to make the maximum total of 300 workers the objective function would suffer a decrease of 1 unit in overall capability. Using the much lower value of 0.001 we obtain the metrics shown in Table 7.

Metric	Value
Z_1	2.8415
Z_2	2.6816
Number of workers	270
Average experience	4.6826
Average language skills	2.9047
Average presentation skills	3.1778

Table 7. Metrics for the solution with lower extra worker penalty

Hiring 20 more workers leads to better staff capabilities, as would be expected. There is a trade between hiring more specialized staff for each position or hiring less staff overall. As always, the choice of the best alternative should be taken by a human resources specialist.

7.4 Summary

To summarize the findings of the previous subsections we combined the results in Table 8.

Setup	Z_1	Z_2	$n_{workers}$	avg_{exp}	avg_{ls}	avg_{ps}
Default	2.8166	2.8261	**250**	4.6009	2.8506	3.0745
Unlimited workers	2.8695	2.7326	302	**4.7003**	2.9061	3.2107
Unlimited consecutive days	2.8291	2.8286	**250**	4.6350	2.8608	3.0335
Better skills	2.6370	2.7552	**250**	4.2929	**3.2060**	**3.7849**
Lower worker penalty	2.8415	2.6816	270	4.6826	2.9047	3.1778

Table 8. Comparison of different solution's metrics. Note that the objective function values are not comparable across all the setups

7.5 Replacement Quality

To validate both the quality of the waiting list assigned during the assignment problem and the efficacy of the replacement problem we created a experimental setup which seeks to emulate different scenarios where staff are missing.

During this experiment we assume that the chance of a given worker missing is independent from the remaining workers, even though that in practice is not always the case.

For each day we selected a percentage of the workers from that day that are missing which follows a normal distribution based on the problem statement information.

$$n_{\text{missing\_percentage}} \sim \mathcal{N}(4,\ 5)$$

Which leads to:

$$n_{\text{missing}} = \frac{n_{\text{missing\_percentage}} \cdot n_{\text{workers}}}{100}$$

Than n_{missing} randomly sampled staff members are passed as a parameter to the replacement problem model. This experiment was repeated 30 times which totaled in 270 simplex algorithm executions.

Based on this we analyze the average experience and skills of the waiting list members which were chosen as candidates. The average capability is 2.6367 ± 0.0685 and the average time and travel distance is 1.6642 ± 0.7761.

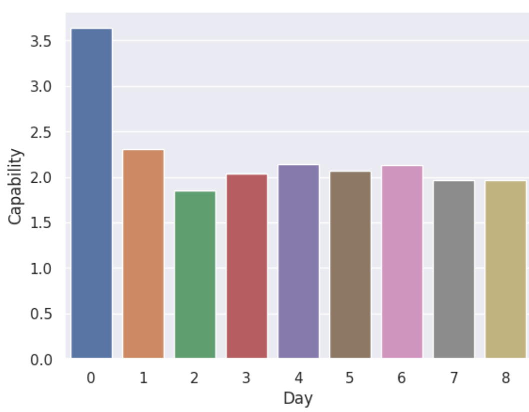

Fig. 5. Average replacement capability

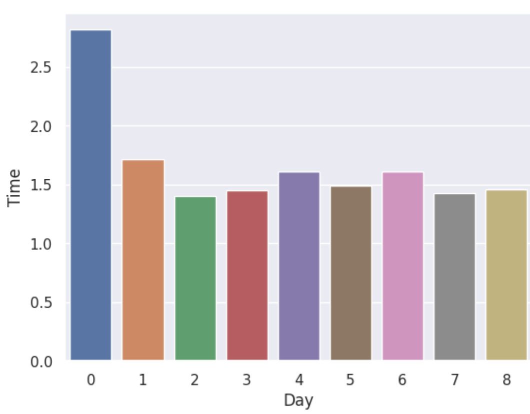

Fig. 6. Average replacement travel time

As seen in Figures 5 and 6 the metrics obtained are balanced across all the days with the notable exception of the first day. We hypothesize that this deviation is caused by the reduced number of positions which are required on this day, which allows for a over-specialization of the waiting list.

8 CONCLUSION

To conclude, in this report we took a look at the problem statement issued by Good'OldGood's auction house and gathered explicit requirements. Based on the issued information it is impossible to formulate a complete solution so we define a set of assumptions under which our tool functions.

We formally define two problems which together cover the requested requirements, one for allocating staff which combines experience, skills and other constraints to create a possible solution, and one to allow for a quick and efficient replacement of missing staff members during the days of the event.

We also analyze and propose different solutions to the problem that can be proposed to the company's human resources expert.

REFERENCES

[1] [Online]. Available: https://www.ibm.com/products/ilog-cplex-optimization-studio/cplex-optimizer

[2] H. Isermann, "Linear lexicographic optimization," *Operations-Research-Spektrum*, vol. 4, no. 4, pp. 223–228, Dec 1982. [Online]. Available: https://doi.org/10.1007/BF01782758

ANNEX

Position	Presentation	Language
Catalogue Desk Assistant	1	1
Client Services Assistant	0	1
Cloak Room Assistant	0	1
Currency Board Operator (Saleroom 1)	0	1
Currency Board Operator (Saleroom 2)	0	1
Dept A Assistant	0	1
Dept B Packing Assistant	0	0
Dept B Viewing Assistant	1	1
Dept C Assistant	0	1
Dept D Assistant	0	1
Dept D Viewing Assistant	1	1
Dept E Assistant	0	1
Dept F Packing Assistant	0	0
Dept F Viewing Assistant	1	1
Dept F Viewing Assistant (VIP)	1	1
Dept G Viewing Assistant	1	1
Dept H Assistant	0	1
Dept I Viewing Assistant	1	1
Dept J Assistant	0	1
Dept K Packing Assistant	0	0
Dept K Set Up Assistant	0	0
Dept K Viewing Assistant	1	1
Dept L Packing Assistant	0	0
Dept L Set Up Assistant	0	0
Dept L Viewing Assistant	1	1
Event Assistant	1	1
Impressionist painting Viewing Assistant	1	1
Information Assistant	0	1
Information Assistant (Hall 3B)	0	1
London Jewellery Viewing Assistant	1	1
Operations & Administration Packer	0	0
Public Relationship Assistant	1	1
Site Assistant	0	1
Spotter	0	0
Waiting List	0	0

Table 9. Importance of skills for each position. Value 1 indicates that the skill is used for the corresponding position

Position	Position Type
Catalogue Desk Assistant	Catalogue Desk Assistant
Client Services Assistant	Client Services Assistant
Cloak Room Assistant	Cloak Room Assistant
Currency Board Operator (Saleroom 1)	Currency Board Operator
Currency Board Operator (Saleroom 2)	Currency Board Operator
Dept A Assistant	Assistant
Dept B Packing Assistant	Packing Assistant
Dept B Viewing Assistant	Viewing Assistant
Dept C Assistant	Assistant
Dept D Assistant	Assistant
Dept D Viewing Assistant	Viewing Assistant
Dept E Assistant	Assistant
Dept F Packing Assistant	Packing Assistant
Dept F Viewing Assistant	Viewing Assistant
Dept F Viewing Assistant (VIP)	Viewing Assistant
Dept G Viewing Assistant	Viewing Assistant
Dept H Assistant	Assistant
Dept I Viewing Assistant	Viewing Assistant
Dept J Assistant	Assistant
Dept K Packing Assistant	Packing Assistant
Dept K Set Up Assistant	Set Up Assistant
Dept K Viewing Assistant	Viewing Assistant
Dept L Packing Assistant	Packing Assistant
Dept L Set Up Assistant	Set Up Assistant
Dept L Viewing Assistant	Viewing Assistant
Event Assistant	Event Assistant
Impressionist painting Viewing Assistant	Impressionist Painting Viewing Assistant
Information Assistant	Information Assistant
Information Assistant (Hall 3B)	Information Assistant
London Jewellery Viewing Assistant	London Jewellery Viewing Assistant
Operations & Administration Packer	Operations & Administration Packer
Public Relationship Assistant	Public Relationship Assistant
Site Assistant	Site Assistant
Spotter	Spotter
Waiting List	Waiting List

Table 10. Conversion from position to position type

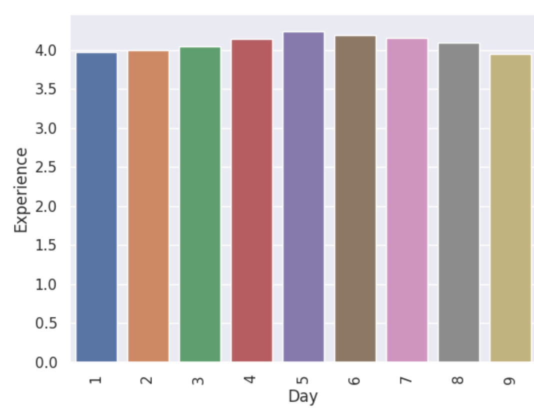

Fig. 7. Average experience per day

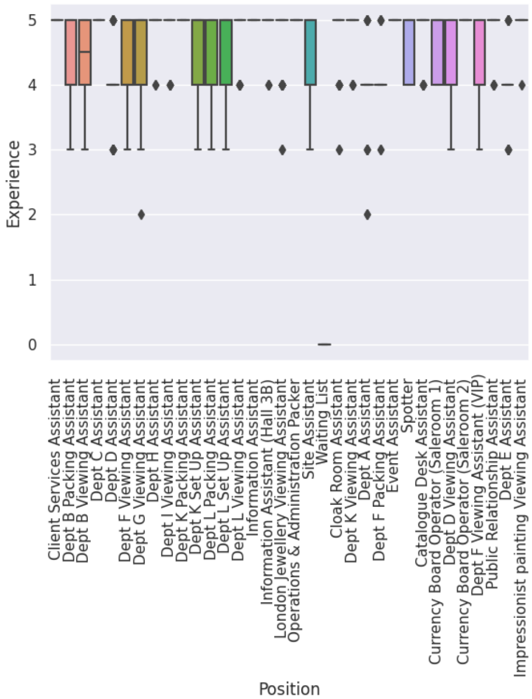

Fig. 8. Experience distribution per position

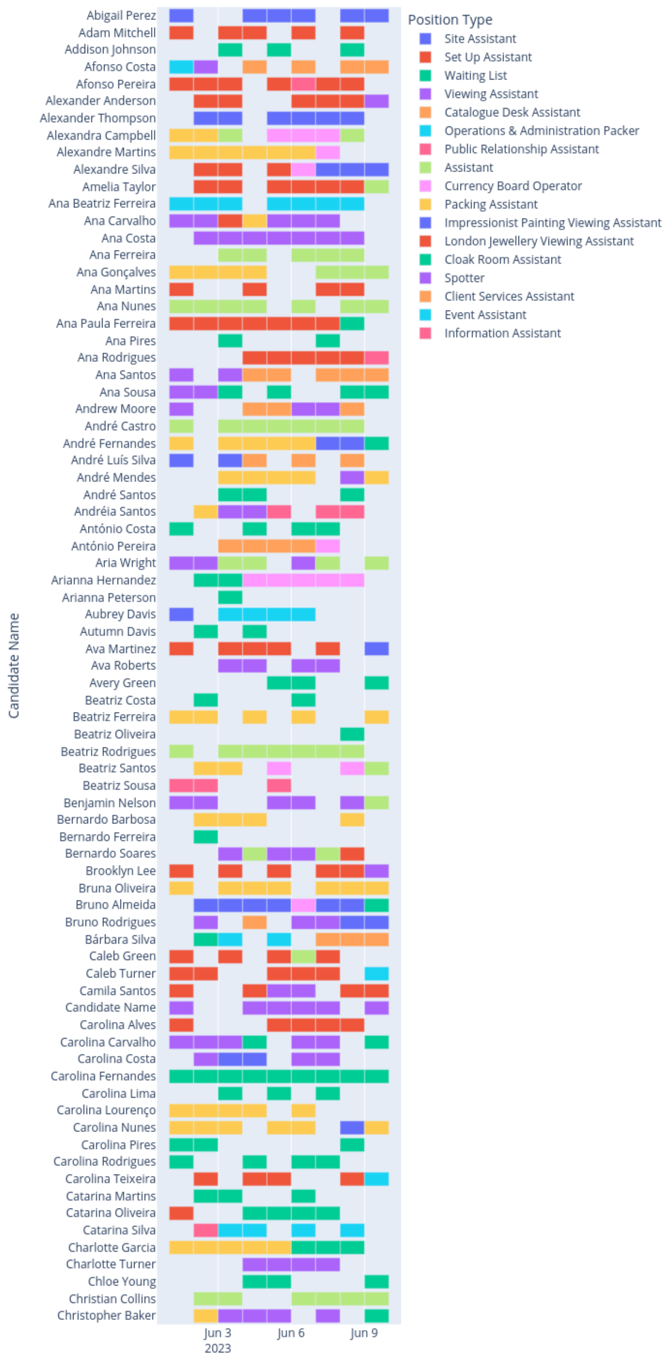

Fig. 9. Gantt chart of the allocations. Part 1/4

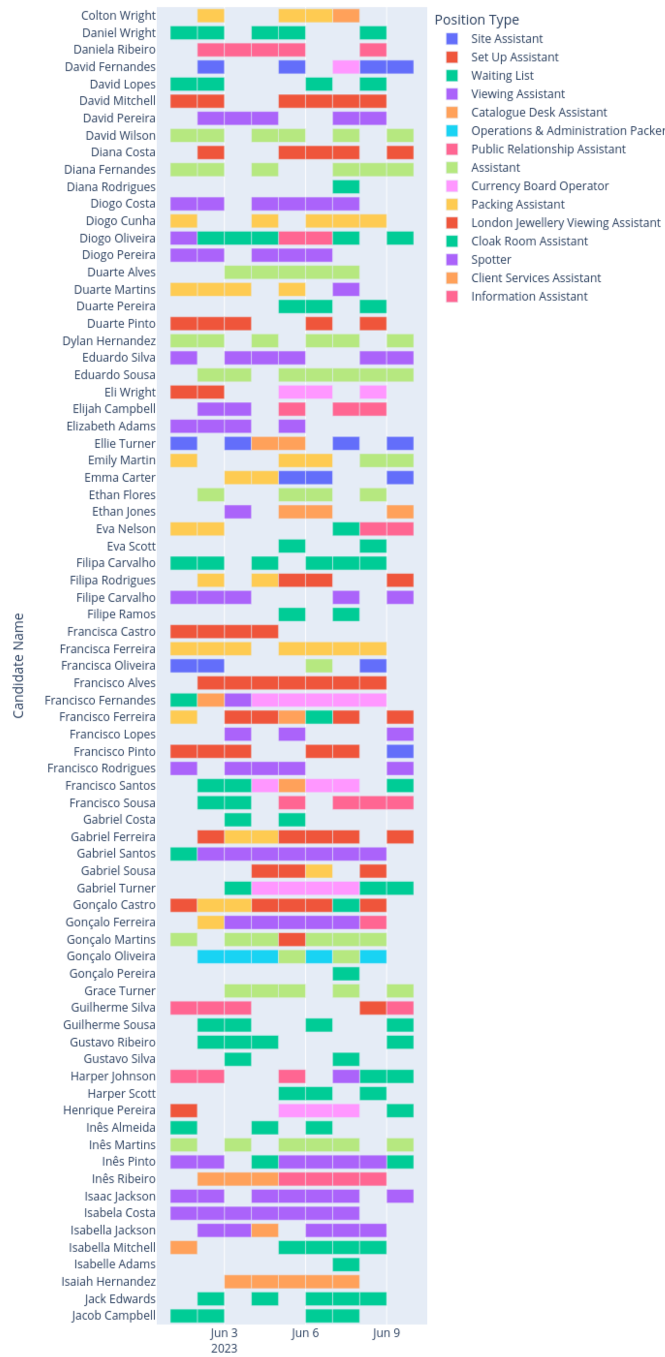

Fig. 10. Gantt chart of the allocations. Part 2/4

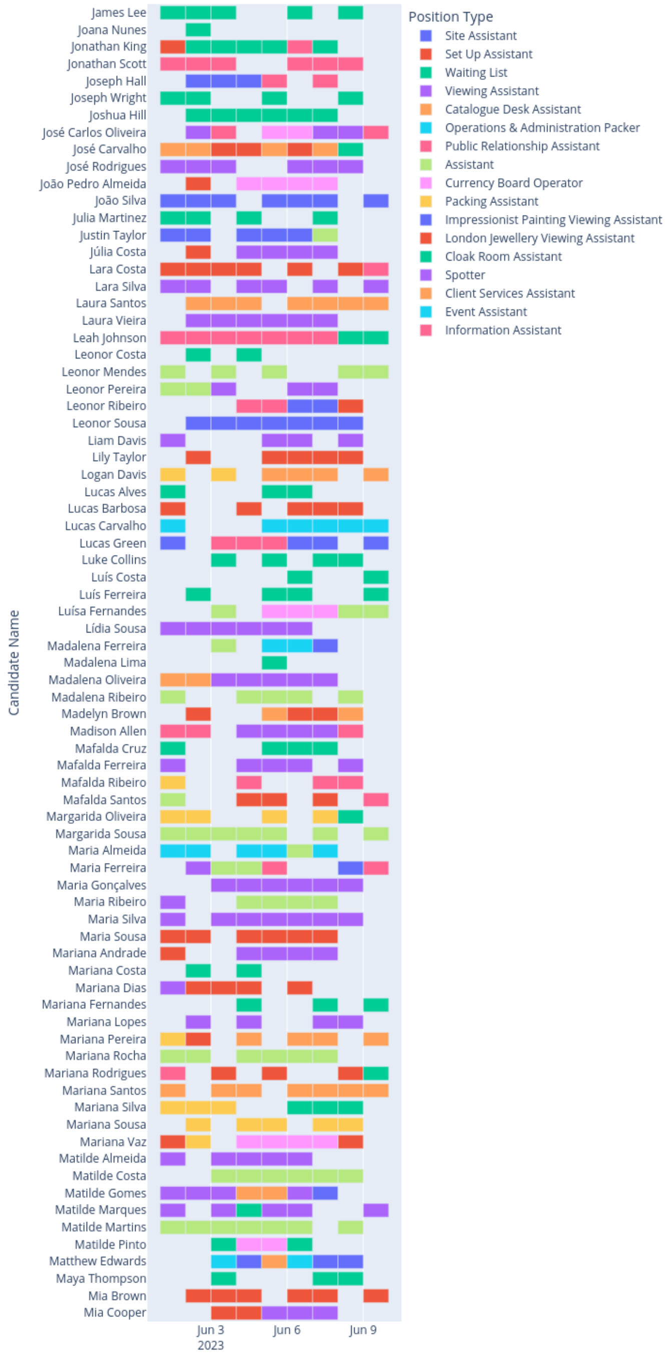

Fig. 11. Gantt chart of the allocations. Part 3/4

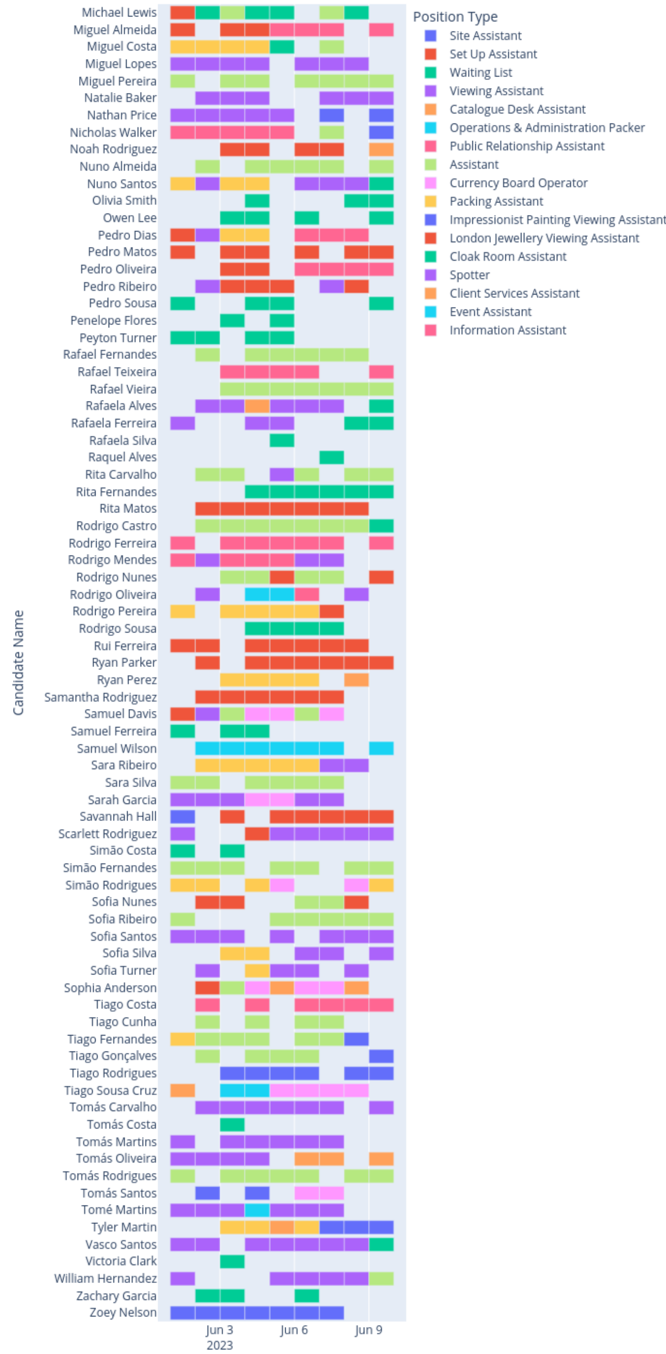

Fig. 12. Gantt chart of the allocations. Part 4/4