MISTA 2011

The Airport Baggage Sorting Station Allocation Problem

Amadeo Ascó · Jason A. D. Atkin ·
Edmund K. Burke

Abstract In this paper we consider the problem of assigning available baggage sorting stations to flights which have already been scheduled. A description and model for the problem are presented, illustrating the different objectives which have to be considered. A number of constructive algorithms for sorting station allocation are then presented and their effects are compared and contrasted for different numbers of sorting station that are available. In particular, it can be observed that the appropriate algorithm selection is highly dependent upon whether reductions in service times are permitted or not and upon the flight density in relation to the number of sorting stations.

Keywords: Airport Baggage Sorting Stations, Scheduling, Heuristics, Constructive Algorithms, Greedy Algorithm

1 Introduction

The mishandling of baggage in airports has been one of the more important issues for passengers for several years in both Europe and the U.S.A. It was ranked third in complaints after cancellations and delays in the 2009 report of the Air Transport Users Council (2009) and its importance was emphasized further in the April 2010 report of the Office of Aviation Enforcement and Proceedings (U.S. Department of Transportation (2010)) where over a hundred thousand baggage reports were logged, ranking baggage complaints in second place. Expected increases in civil air traffic (ICAO (2010) and Federal Aviation Administration (2010)) will continue to increase the complexity of these problems.

Baggage which is checked in will travel through the baggage system to baggage sorting stations. Baggage handlers at the sorting stations will sort and load the baggage

---

Amadeo Ascó
School of Computer Science
University of Nottingham
E-mail: a.asco@cs.nott.ac.uk

Dr. Jason A. D. Atkin
School of Computer Science
University of Nottingham
E-mail: jaa@cs.nott.ac.uk

Prof. Edmund K. Burke
School of Computer Science
University of Nottingham
E-mail: ekb@cs.nott.ac.uk

onto baggage carts or into special containers that go straight into the aircraft. These sorting stations are not usually co-located with the aircraft, so the pairing of sorting station to gate is important. Furthermore, baggage often arrives or accumulates at the sorting station before the aircraft arrive at the gate, so the sorting station is needed earlier than the gate, and usually for longer than the gate.

We present a model for the baggage sorting station allocation problem at airports, along with a consideration of the various objectives. The problem can be observed to be a multi-objective resource constrained allocation problem, where the aim is to allocate the limited baggage handling resources amongst the various flights which have to be serviced. Research into a similar problem was performed in Abdelghany et al (2006), but various questions were left unanswered. This paper aims to answer these questions and to perform a rigorous analysis of the effects and benefits of various different constructive algorithms for the problem, with a view to potentially utilising these to provide initial solutions for further search methods. The intention is not to determine the 'perfect' algorithm for constructing a sorting station allocation, but instead to understand the effects and trade-offs of different choices.

In the model considered here, which was presented by Abdelghany et al (2006), flights are first allocated to stands (i.e. the Airport Gate Assignment Problem (GAP) is solved) before the baggage allocation problem is considered. An example allocation of flights is shown in Figure 1, which can be considered to be a type of GANTT chart, where the vertical axis represents the stands and the horizontal axis shows the time of day. Each rectangle on the diagram represents a specific flight and shows the times that the flight will use the stand. Each stand is numbered: the first digit is the terminal number, the second digit is the pier number, and the last two digits are the individual stand identification. For example, the top row shows four flights assigned to stand 1101, which refers to terminal 1, pier 1 and stand 1.

The root of the problem for baggage sorting station allocation is that baggage sorting stations are required for longer than flights, so there can be no one-to-one correspondence between baggage sorting stations and stands, and ideal locations cannot be guaranteed. Indeed, ideally there should also be a buffer time between sorting station usages, to reduce the risk of small perturbations affecting the allocations and of the mixing of baggage between flights, but the contention for baggage sorting stations means that this sometimes has to be reduced or eliminated. One of the purposes of this paper is to better understand the way in which the potential reductions in buffer times affect the various algorithms.

There are a number of objectives to consider in the baggage sorting station allocation problem (for example, maximising the allocations, maximising available buffer times and allocating flights to the closest sorting stations) and these are in obvious conflict with each other. Any solution method needs to take this into account. In particular, different constructive algorithms will be observed in this paper to perform better for differing objectives. Hybridisation of the algorithms themselves or the appropriate utilisation or recombination of solutions from different algorithms may potentially lead to allocations which better reflect the overall objectives.

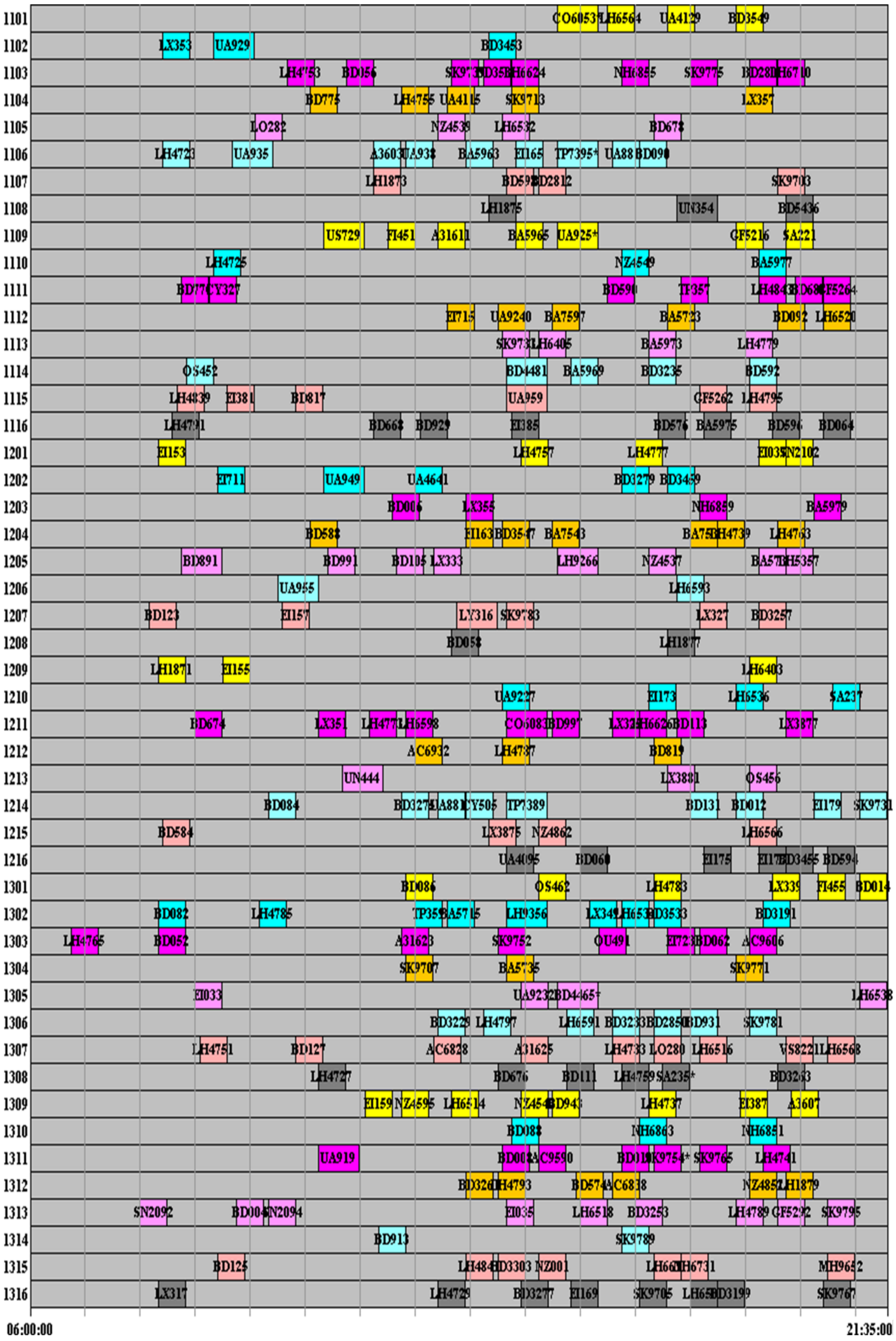

Fig. 1 Random allocation to stands of 270 flights from 1st March 2010.

This paper is structured as follows: Firstly, the problem description and model are presented, followed by a description of the considered algorithms. The results from applying the presented algorithms to the problem are then provided and various observations are made and explanations given. Finally some conclusions are presented in Section 5.

2 Problem Description and Model

The problem which is considered in this paper can be summarised as the assignment of available baggage sorting stations to already scheduled flights. For the baggage sorting station allocation problem, the flights will have already been assigned to stands, which are often grouped along piers around the terminals.

2.1 Airport Layout

The airport geometry plays an important role in the allocation of resources and the safety of the airport operations. An overview of the airport configurations and technologies for the transportation of passengers and baggage was presented by Pitt et al (2002) who concentrated on airport configurations and the availability of different types of resources. Rijsenbrij and Ottjes (2007) provided an overview of different elements of the baggage handling system and gave a description of the way in which baggage is currently handled, identifying potential areas of improvement.

Figure 2 provides a stylised diagram of an example layout. The stands are grouped on piers, which contain their baggage sorting stations at their base, placed perpendicularly to the pier. For any stand, it will be better to allocate the luggage to the sorting stations on the same side of the same pier. Alternatively, more distant sorting stations could be used, but these are less preferable. A ‘cost’ can be associated for a stand-sorting station pairing and one aim is to reduce this cost, by allocating as many flights as possible to their preferred sorting stations.

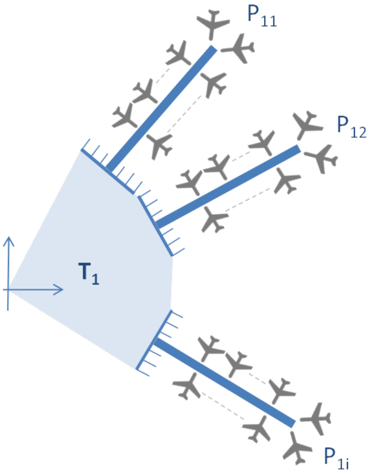

Fig. 2 Airport Layout.

The model which is utilised in this paper is appropriate for any airport where there are groupings of aircraft/gates which enforce a sorting station group preference (such as when aircraft are on piers) and where there is a distance or cost metric for the allocation of a sorting station to a flight, rather than being specific to the example layout described above. For example, at some airports, the sorting stations may be between the gates, in which case the preference for the distance/cost for allocating flight-sorting station pairs may be much stronger, but the group/pier preference may not be so strong.

2.2 Input Data and Constants

Table 1 summarises the various constants which are used in the model described in this paper. The relationship between the timing values is illustrated in Figure 3. Given these definitions, the service time of a flight is the duration from s_j to e_j and the ideal service time is the duration from t_j to e_j.

Table 1 List of the constants and input values for the model.

Name	Description
n	The total number of baggage sorting stations under consideration.
m	The total number of flights to which sorting stations should be allocated.
T_j	The base service time for flight j.
B_j	The desired buffer time for flight j.
R_j	The maximum reduction of service time allowed for flight j (we assume $R_j = B_j$ for this paper, so that the buffer time can be reduced but the base service time cannot).
e_j	The end service time for flight j.
t_j	The target starting service time for flight j, $t_j = e_j - T_j - B_j$, assuming the full buffer time is available.
C_j	A flight specific constant representing the amount of baggage to be processed for flight j. This determines the difficulty involved with allocating the flight to a sorting station which is further away. For example, this may represent the number of delivery trips required to move the baggage from the sorting station to the aircraft. In the absence of baggage load figures, we set $C_j = 1$ for all aircraft for this paper.
d_{ij}	The distance between baggage sorting station i and flight j.

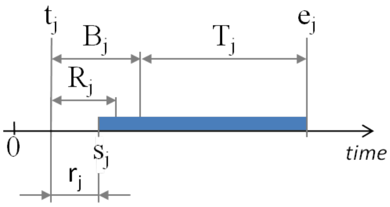

Fig. 3 View of the components of the service time.

2.3 Service Period

A service period is associated with each departing flight, during which the baggage for the flight is accumulated at the assigned baggage sorting station and finally loaded onto baggage carts for transfer to the flight. This service period may (optionally) be extended by applying a buffer time to prefer having a gap between the servicing of consecutive flights by the same sorting station.

2.4 Buffer Time

The buffer time is applied between two consecutive flights on the same baggage sorting station in order to absorb small disturbances in the real system behaviour. Buffer times are a common means of increasing robustness to avoid delays, as studied by Nikulin (2006) and Mulvey et al (1995). Buffer times were used in the scheduling of baggage sorting stations by Abdelghany et al (2006), and Wu and Caves (2004) used them in the optimization of the aircraft turnaround process. The Airport Gate Assignment Problem (GAP) has some similar characteristics to the baggage sorting station allocation problem and buffer times have been considered for the GAP by Hassounah and Steuart (1993), Yan and Chang (1997), Bolat (2000), Yan et al (2002) and Wu and Caves (2004). Yan and Huo (2001) and Yan et al (2002) analysed the effect and sensitivity of the buffer time, and Wei and Liu (2009) compared three distribution functions of buffer time with fixed buffer times using Fuzzy models. Buffer times have also been considered in other areas, such as the Resource Constrained Project Schedule Problem (RCPSP), where Long and Ohsato (2008) applied Fuzzy Critical Chain methods with buffer times.

Two constant ideal buffer time values have been considered in this paper, depending upon the flight destination type, namely European (short haul) or long distance, but buffer times could be introduced which depend upon individual aircraft types or even upon individual airline operators, routes or baggage handlers.

Given that the intention of using a buffer time is to reduce the impact from small perturbations in previous flight allocations to the same baggage sorting station, the buffer time has been allocated solely to the starting time of the service in this paper. If a previous flight closes its sorting station late due to delays, no re-scheduling will be required until all of the buffer time allocated to the next flight has been consumed by the delay. Of course, a flight could potentially consume the buffer on both sides of its service period, potentially opening a sorting station early and consuming the buffer time allocated to that flight, then closing the sorting station late and consuming the buffer time for the following aircraft.

2.5 Decision Variables

Table 2 lists the decision variables which are used in this model. The solution algorithms will attempt to find values of y_{ij} and r_j such that the constraints in Section 2.6 are met and the relevant objectives (e.g. maximising the allocations and minimising the reductions in service times) in Section 2.7 are improved.

Table 2 List of the decision variables which are used in this model.

Name	Description
y_{ij}	Specifies the allocation of flights to sorting stations. $y_{ij} = 1$ if baggage sorting station $i \in \{1, \ldots, n\}$ is allocated to flight $j \in \{1, \ldots, m\}$, and 0 otherwise.
r_j	Specifies the necessary reduction in service time for flight $j \in \{1, \ldots, m\}$, given the allocated starting service time, s_j.
s_j	The allocated starting service time for flight $j \in \{1, \ldots, m\}$, given that a sorting station can only service one flight at a time. s_j can be determined from r_j since $s_j = t_j + r_j$.

2.6 Constraints

Various constraints apply to the allocation of baggage sorting stations and can be summarised as follows.

2.6.1 Assignment Limits

Each flight must be assigned to at most one baggage sorting station, as expressed by Inequality (1). In normal operations, each flight should be allocated to exactly one sorting station, in which case Inequality (1) would instead be an equality. However, in extreme situations, where there are insufficient sorting stations (as considered in this paper) there may be no feasible allocation of flights to sorting stations such that all flights can be allocated, hence the inequality.

$$\sum_{i=1}^{n} y_{ij} \leq 1 \qquad \forall j \in \{1, \ldots, m\} \tag{1}$$

2.6.2 Reduction in Service

Baggage sorting stations can only be simultaneously used by one flight, so it may be necessary to reduce the service times of flights (usually by reducing the buffer times between flights) in order to allocate flights to the same sorting station. The principal objective is usually to maximise the assignment of baggage sorting stations to flights, as expressed by Formula (4).

For any pair of different flights j and l whose service times overlap (where $e_j \leq e_l$), if the overlap in service times is greater than the maximum allowed reduction R_l, then both flights cannot be assigned to the same baggage sorting station. Otherwise they may be assigned to the same baggage sorting station as long as the service time of flight l is reduced sufficiently to remove the overlap.

$$\begin{array}{ll} \multicolumn{2}{l}{\forall j, l \in \{1, \ldots, m\},\ j \neq l,\ s.t.\ \ e_j < e_l\ and\ t_l < e_j,\ \forall i \in \{1, \ldots, n\}:} \\ r_l \geq (y_{ij} + y_{il} - 1) * (e_j - t_l) & \text{if } (e_j - t_l) \leq R_l \\ y_{ij} + y_{il} \leq 1 & \text{otherwise} \end{array} \tag{2}$$

2.6.3 Limit Service Reduction

The service time reduction may not exceed a limit, as expressed by Inequality (3).

$$0 \leq r_j \leq R_j \quad \forall j \in \{1, \ldots, m\} \tag{3}$$

2.7 Objectives

A number of objectives need to be considered for this problem, and there is a trade-off to be made between them.

2.7.1 Maximise Assignment of Baggage Sorting Stations

The first and most important objective is to maximise the number of flights assigned to baggage sorting stations, as expressed by Formula (4). In practice at airports, this objective would probably be a hard constraint at most times, since all flights would normally have to be serviced, but we wish to observe the performance of the algorithms when there are too few sorting stations as well as when these are sufficient or plentiful.

$$max \sum_{i=1}^{n} \sum_{j=1}^{m} y_{ij} \tag{4}$$

2.7.2 Minimise Distance

The distance between the assigned baggage sorting stations and the flights should be as short as possible. This objective aims to minimise the inconvenience, work and time involved in getting baggage to the aircraft, as previously discussed and could relate to preferences rather than strictly to distances, as discussed later. It can be expressed by Formula (5) where $\sum_{i=1}^{n} \left(y_{ij} * d_{ij} \right)$ corresponds to the distance between flight j and its allocated baggage sorting station.

$$min \sum_{j=1}^{m} \left(C_j * \sum_{i=1}^{n} \left(y_{ij} * d_{ij} \right) \right) \tag{5}$$

2.7.3 Fair Workload

The fairness objective corresponds to the minimisation of the total deviation of the actual usage of each baggage sorting station from the mean usage of all baggage sorting stations. This is expressed by Formula (6), where $e_j - s_j$ corresponds to the actual service time for the flight j, which is the usage time of the baggage sorting station. This objective aims to find a fairer allocation across sorting stations, as discussed in Abdelghany et al (2006).

$$min \sum_{i=1}^{n} \left| \sum_{j=1}^{m} (y_{ij} * (e_j - s_j)) - \frac{\sum_{i=1}^{n} \sum_{j=1}^{m} (y_{ij} * (e_j - s_j))}{n} \right| \tag{6}$$

2.7.4 Minimise Reduction in Service

Given the detrimental effects that the reduction in service time has for the robustness of the allocation against real-life delays, it is advisable to minimise the total reduction in service time, thus maximising total buffer times. This objective can be expressed by Formula (7).

$$min \sum_{j=1}^{m} r_j \tag{7}$$

2.7.5 Preferred Piers

Flights may have preferred piers and these should be considered when assigning baggage sorting stations. For this paper, we assume that it is preferable to allocate to each flight, sorting stations which are on the same pier. This objective is correlated to the distance minimisation objective, Formula (5) and is, therefore, not considered separately. The assignment of baggage sorting stations to preferred piers is considered in different ways by the different sorting station allocation algorithms which are described in Section 3.2.1, which differ in whether they first consider pier preferences or avoiding buffer time reductions.

2.7.6 Flights to the Same Destination

It is preferable that flights from the same carrier to the same destination be assigned to the same baggage sorting station, so that, for example, any delayed baggage could be transported on the next flight. However, flights would also normally be allocated to stands according to carrier, and potentially according to destination (or at least long-haul vs short-haul). Since we have utilised random stand allocation (see Section 4.1) for the experiments in this paper, it does not make sense to consider this objective and so it has been ignored.

2.7.7 Other Objectives

Other objectives are also possible, such as a reduction in the number of open sorting stations (to reduce the number of baggage handlers required). However, these are in direct conflict with equity and reduction in service considerations. These are not considered in this paper for reasons of space, although there are some observations made about these in Section 3.2.2, in the Last In First Out (LIFO) discussion.

A non-linear cost for service time reduction could also be used, so that fewer larger reductions are penalised greater than many lower reductions, since large reductions in the buffer time are far less favourable than smaller reductions. This will be an interesting area for future research but is not considered in this paper.

3 Algorithms

The constructive algorithms considered in this paper allocate baggage sorting stations to flights one at a time until no more allocations are possible. Flights are first ordered according to one of the considered flight ordering methods. A sorting station is then selected for each in turn. The flight ordering and baggage sorting station allocation problems are considered bellow.

3.1 Flight Ordering Methods

The Flight Ordering method determines the order in which flights are selected for assignment. The following different sorting approaches are considered here:

1. **Order by Starting Time (OST)**. From the algorithm pseudo code presented therein, this appears to be what was previously used in Abdelghany et al (2006) and orders flights into ascending order of their t_j values.
2. **Order by Departure Time (ODT)**. This was previously used by Ding et al (2005) for the Airport Gate Assignment Problem (GAP). This orders flights into ascending order of their e_j values. When two flights have the same service end times, they are sorted by their target starting time t_j. When service time reductions are not permitted, sorting by service end times provides maximum allocations when using Last In First Out (LIFO) baggage sorting station selection and not constraining the set of sorting stations from which to select (see Section 3.2.2).

Order by Departure Time Lookahead and Improvement (ODTLI). Haralick and Elliott (1980) considered the concept of “Lookahead and anticipate the future in order to succeed in the present” and “Lookahead to the future in order not to worry about the past”. A type of look-ahead was also used in Voß et al (2005). The ODT flight ordering method could potentially perform badly on the objectives other than the maximisation of allocations and, in particular, the reuse of sorting stations can lead to an extremely inequitable allocation across sorting stations. The aim of the ODTLI look-ahead is to keep the ODT flight ordering but to look ahead when allocating sorting stations, to improve one of the other objectives while maintaining the maximal allocation of flights to sorting stations. The developed ODTLI algorithm looks ahead to find an allocation of future flights to sorting stations according to the ODT ordering, then attempts to find an improved allocation using the current Baggage Sorting Station Selection (BSSS) method, by exchanging some of the baggage sorting station allocations (using one-for-one swaps) without reducing the number of flights which are allocated to sorting stations.

3.2 Baggage Sorting Station Allocation

Once the flight to allocate has been identified, the next stage is to determine which sorting station to allocate it to. Baggage Sorting Station Allocation involves two stages. The first decision is upon which sets of baggage sorting stations to consider for allocation and in what order. In particular, whether only those for the same pier should be considered first, and whether service time reductions should be considered. The second decision involves the ranking of baggage sorting stations within each set, to enable the selection of an individual baggage sorting station to allocate.

3.2.1 Baggage Sorting Station Allocation Algorithms

The Baggage Sorting Station Allocation Algorithm determines which sets of baggage sorting stations (for example only those on the same pier, or on all piers) are considered, in which order, and at what point reduction in service is considered within each set. The baggage sorting stations within each set are then considered according to a selection priority given in Section 3.2.2. Algorithms A to E express different priorities. Algorithm A will attempt to allocate all aircraft to their own piers before considering allocating any aircraft to other piers, and Algorithm B will attempt to allocate all aircraft anywhere without reductions in service prior to reducing the service of any aircraft. Algorithms C and D are similar to A and B (respectively) but consider alternative piers or reductions in service for the current aircraft prior to considering the next aircraft, giving a much weaker preference overall. Algorithm E will not impose any restriction.

Algorithm A: Baggage Sorting Station Allocation Algorithm *A*

```
Order all the flights based on the current flight ordering algorithm, Section 3.1.
forall the flights do
    Consider sorting stations on own pier, without reduced service times.
    Otherwise, consider sorting stations on own pier allowing reduced service.
end
forall the unassigned flights do
    Consider any sorting station in airport, without reduced service times.
    Otherwise, consider any sorting station in airport, allowing reduced service.
end
```

Algorithm B: Baggage Sorting Station Allocation Algorithm *B*

```
Order all the flights based on the current flight ordering algorithm, Section 3.1.
forall the flights do
    Consider sorting stations on own pier, without reduced service times.
    Otherwise, consider any sorting station in airport, without reduced service.
end
forall the unassigned flights do
    Consider sorting stations on own pier allowing reduced service times.
    Otherwise, consider any sorting station in airport, allowing reduced service.
end
```

Algorithm C: Baggage Sorting Station Allocation Algorithm *C*

```
Order all the flights based on the current flight ordering algorithm, Section 3.1.
forall the flights do
    Consider sorting stations on own pier, without reduced service times.
    Otherwise, consider sorting stations on own pier allowing reduced service.
    Otherwise, consider any sorting station in airport, without reduced service.
    Otherwise, consider any sorting station in airport, allowing reduced service.
end
```

Algorithm D: Baggage Sorting Station Allocation Algorithm *D*

```
Order all the flights based on the current flight ordering algorithm, Section 3.1.
forall the flights do
    Consider sorting stations on own pier, without reduced service times.
    Otherwise, consider any sorting station in airport, without reduced service.
    Otherwise, consider sorting stations on own pier allowing reduced service.
    Otherwise, consider any sorting station in airport, allowing reduced service.
end
```

Algorithm E: Baggage Sorting Station Allocation Algorithm *E*

```
Order all the flights based on the current flight ordering algorithm, Section 3.1.
forall the flights do
    Consider any sorting station in airport, without reduced service times.
    Otherwise, consider any sorting station in airport, allowing reduced service.
end
```

In each case, once the algorithm has determined the set of sorting stations to consider, the appropriate sorting station to allocate is determined by the baggage sorting station selection method which is being used (see Section 3.2.2).

For this paper, we have concentrated on baggage sorting station allocation algorithms A, C and E since these consider the effect of prioritising the pier or not. Future work will consider algorithms B and D, which prioritise avoiding service time reductions. These have been included here for completeness.

3.2.2 Baggage Sorting Station Selections

The Baggage Sorting Station Selection method determines which of the baggage sorting stations in the current set should be assigned to the current flight. The following methods are considered:

1. **First In First Out (FIFO)**: The baggage sorting station with the earliest free service time amongst all of the baggage sorting stations in the set under consideration is selected. This will initially keep opening new service stations, while they exist, since a new one would always be least recently used. This is useful for meeting the fairness objective expressed by Formula (6).
2. **Last In First Out (LIFO)**: The most recently used baggage sorting station amongst those in the set being considered is selected. This selection reduces the number of baggage sorting stations in use at any time, since a new baggage sorting station is only opened when the previous ones cannot be assigned to the flight. With the ODT and ODTLI flight ordering methods, no reduction in service time and allocation algorithm E (so that all sorting stations are considered, rather than only the ones on the preferred pier), this selection method guarantees the maximum allocations (maximising the objective expressed by Formula (4)), by minimising the wasted/idle time between flights, Ding et al (2004) and Cormen et al (2001).
3. **Closest**: The baggage sorting station with the smallest distance to the current flight is selected from those in the set under consideration. This considers both new and previously used service stations. This method is useful for meeting the distance reduction objective expressed by Formula (5).

4 Results

This section details the experiments which were performed to evaluate the differences between the algorithms which were described in Section 3, with differing numbers of sorting stations available for allocation to flights. The behaviour can, therefore, be studied when there are too few sorting stations as well as when the sorting stations are plentiful.

4.1 Problem Data

Since it would be unrealistic to assume that baggage from a flight at a stand in a terminal is serviced by a baggage sorting station in another terminal (e.g. passengers usually go through security and board flights from the same terminal where they checked in their baggage), it was decided to centre the analysis on a single terminal. In order to obtain a more realistic spread of flights (since flights are rarely evenly spread over the day), publically available flight time data from the British Airports Authority (BAA) website was used for these experiments, consisting of 219 flights from the 16^{th} December 2009 and 270 flights from the 1^{st} March 2010, all departing from London Heathrow Airport. The data for the 16^{th} December has a large number of flights clustered around the middle of the day, whereas the data for the 1^{st} March has flights which are more spread out over a long period. The 1^{st} March is a more normal day, but the clustering of the 16^{th} December is extremely useful for the purposes of this paper, where we wish to consider the effects under very resource constrained conditions.

Since the data did not contain the assignment of stands to flights, different problems were generated by allocating the flights randomly to the stands avoiding any overlap on a single stand. In order to minimise any introduced bias from these random allocations, a hundred different random allocations were generated. We realise, of course, that real schedules will have some bias, for airlines preferences, and future work will consider the real stand allocations. One example allocation is illustrated by Figure 1. The box-and-whisker diagrams in the following sections (from Figure 6 onwards) illustrate the results of the sorting station allocation algorithms (which are themselves deterministic) across the 100 different stand allocations. In each diagram, results are shown next to each other for each number of baggage sorting stations and are in the same order as they are listed in the key. In allocating flights to stands, it has been assumed that all of the stands were suitable for any of the aircraft. The available stands were assumed to be equally distributed over three piers, with 16 stands per pier.

Various experiments were executed using a single threaded Java application, 14, 16, 18, 20, 22, 24, 26, 28, 30, 34 and 36 baggage sorting stations per pier, changing the total number of sorting stations in increments of 6. For the purpose of the distance reduction objective, a distance of one unit was assumed between different sides of a pier and a distance of two units was assumed between different piers (see Figure 2), so that it is preferable to use the other side of the same pier before considering sorting stations for other piers. For the moment, we consider that reductions in service time can only reduce the buffer time rather than the base service time, so $R_j = B_j$. Service times were set so that $T_j = 1$ hour and $B_j = 15$ minutes for European flights, and $T_j = 1\frac{3}{4}$ hours and $B_j = 30$ minutes for non European (longer haul) flights, since these are usually larger flights with more baggage and requirements to check in earlier.

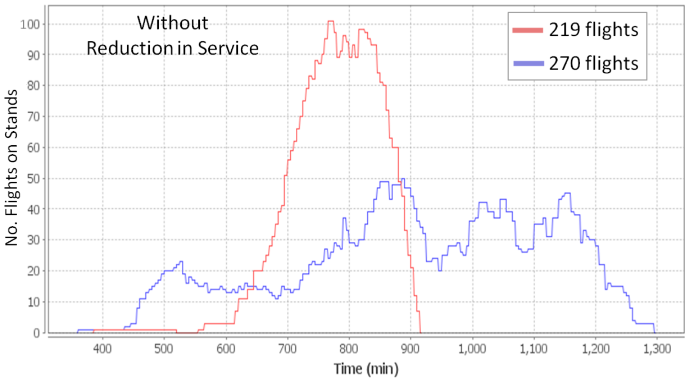

Fig. 4 Without Reduction in Service (including buffer times)

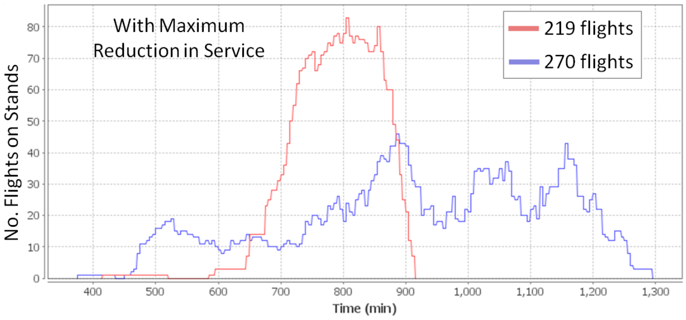

Fig. 5 With Maximum Reduction in Service (i.e. no buffer times)

Figures 4 and 5 show the total number of flights which required service at different times of the day, with and without the buffer times. Figure 4 shows the number of sorting stations which are needed when full buffer times are used (i.e. where there is no service time reduction allowed) and Figure 5 represents the number of flights actually requiring service at that time (i.e. no buffer times are included). It is possible to draw the following conclusions:

1. The number of sorting stations required differs greatly over the day.
2. With a limited number of baggage sorting stations, the maximum heights of the lines in Figures 4 and 5 can be considered to be an indication of the difficulty of the allocation problem.
3. The 219 flight dataset is more concentrated, requiring far more sorting stations over a shorter time period.
4. Fewer sorting stations are required at the peaks when buffer times are not included, but the absence of the buffer times would result in less robust solutions.

Various experiments were executed using a single threaded Java application, running on a 3GHz Intel(R) Core(TM)2 Duo CPU, desktop with 2GB RAM, running Windows XP (SP3). Each execution of the constructive algorithm took no more that 9 milliseconds.

4.2 Optimal allocations

As discussed in Section 3.2.2, Order by Departure Time (ODT) with, Last In First Out (LIFO) sorting station selection, no reductions in service permitted and consideration of all sorting stations at once (Baggage Sorting Station Allocation Algorithm (BSSAA) E) provides optimal allocations.

The same problem was formulated as an Integer Linear Programme and solved using CPLEX. Unsurprisingly, CPLEX provided the same (optimal) number of allocations. However, it took a hundred times as long to solve.

We wished to investigate whether CPLEX could also find an optimal solution for a weighted sum of the different objectives. We generated a combined objective function using relative weights of 10 for the maximum assignment objective (expressed by Formula (4)), 0.005 for the minimising reduction in service objective (expressed by Formula (7)) and 1 for the distance minimisation objective (expressed by Formula (5)), so that these would be considered as primary, secondary and tertiary objectives, respectively, taking into account the relative magnitude of each component. Experiments were executed for the 219 and 270 flight cases using CPLEX but the program was stopped after 8 hours with no improvement, if it had not run out of memory by that point (having consumed up to 87GB of virtual memory). CPLEX was not able to achieve proven optimality in any of the cases and a lot of memory was utilised in each case.

4.3 Initial Observations

Experiments were executed for different numbers of baggage sorting stations, with each of the Baggage Sorting Station Allocation Algorithms (BSSAAs) and Baggage Sorting Station Selection (BSSS) methods. Two cases were considered: without reduction in service times (i.e. requiring full buffer times) and with reduction in service times (i.e. allowing buffer times to be reduced).

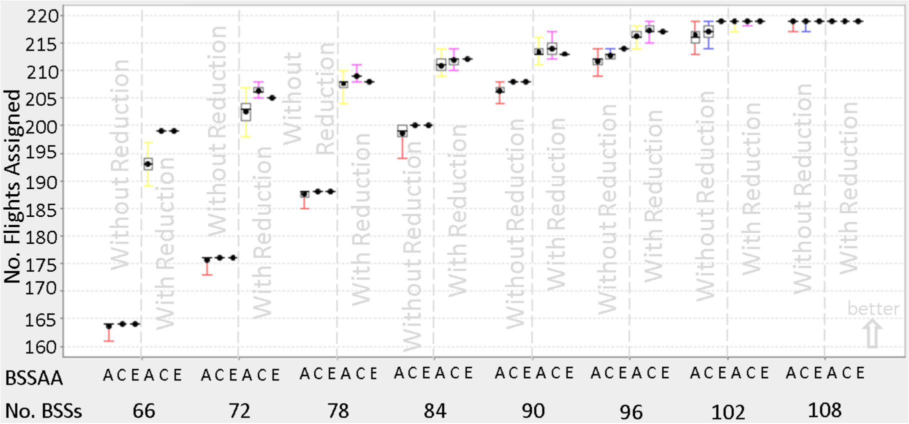

Fig. 6 Number of sorting stations assigned for 219 flights for ODTLI flight ordering and LIFO sorting station selection, with and without permitting reductions in service.

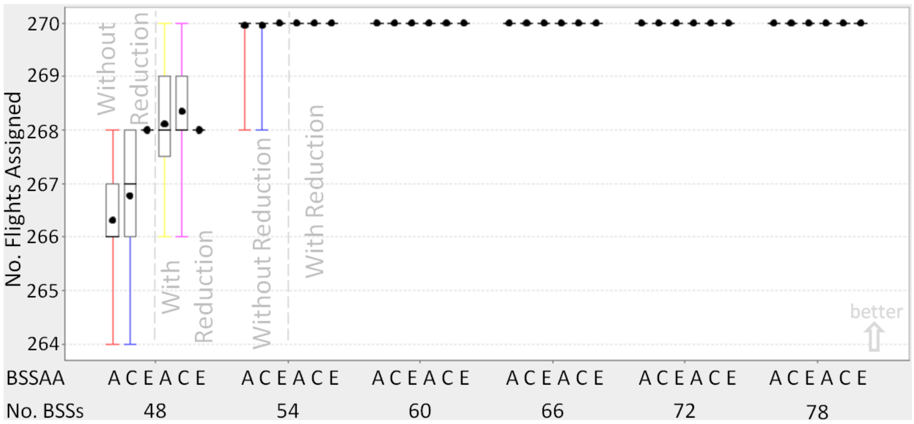

Fig. 7 Number of sorting stations assigned for 270 flights for ODTLI flight ordering and LIFO sorting station selection, with and without permitting reductions in service.

Figures 6 and 7 show (for the 219 and 270 flight problems, respectively) the number of sorting stations which could be allocated to flights for the Order by Departure Time Lookahead and Improvement (ODTLI) flight ordering with LIFO sorting station selection, and both when no reduction in service is allowed and when reduction in service is allowed for various numbers of available sorting stations.

These figures give an idea of the behaviour of the allocation as the flight density and number of baggage sorting stations change. The following can be observed:

1. As expected, more assignments are achieved when allowing reductions in service time, as long as there are insufficient sorting stations available. Maximising allocations is probably the most important objective.
2. When service time reductions are permitted, the ODTLI flight ordering, with LIFO sorting station selection and allocation algorithm E no longer guarantees the maximum allocation of sorting stations. Algorithms A and C, which consider reductions in service prior to considering other piers (thus reducing service times more often), sometimes achieve better allocations (e.g. 72 and 78 baggage sorting stations Figure 6 and with 48 sorting stations Figure 7).

In order to determine the maximum sorting station allocations with reduction in service times permitted, experiments were executed with the buffer times removed (equivalent to maximal service time reduction), ODTLI flight ordering, sorting station allocation algorithm E and LIFO sorting station selection. The results are shown in Figures 8 and 9, for the 219 and 270 flight problems, respectively. More flights can be allocated when reductions are permitted, as expected, until sufficient sorting stations are available to allocate the maximum allocation even without needing reductions. In most cases, allowing reduction is almost as good as using maximum reductions.

The results for maximal reductions can be compared to the earlier results where reductions were and were not allowed. With maximum reductions (i.e. no buffer times), the maximum allocation occurs when there are 84 and 48 baggage sorting stations for the 219 and 270 flight problems respectively. Without reductions, the maximum allocation occurs when there are 102 and 54 baggage sorting stations for the 219 and 270 flight problems respectively. These values are surprisingly close to the lower bounds for the allocations which were given in Figures 4 and 5, which were 83, 46, 101 and 50 for these four cases, respectively. In fact, given that sorting stations were increased in multiples of 6 (2 extra at a time per pier), these are the lowest values considered which met the lower bound.

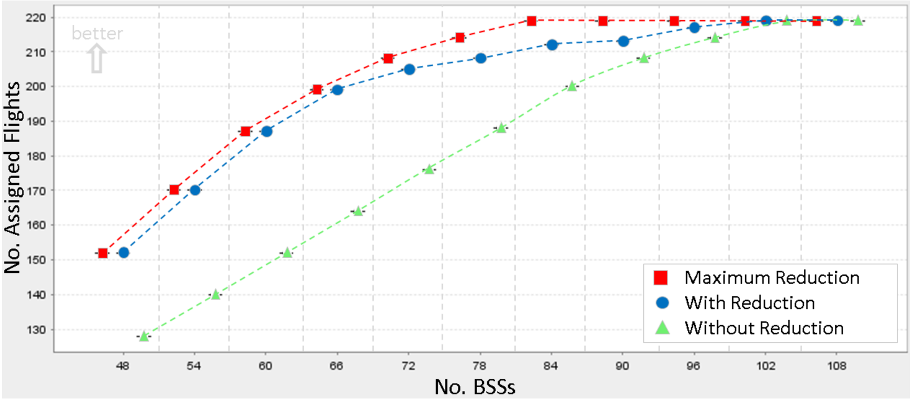

Fig. 8 Number of assignments for 219 flights for ODTLI flight ordering, sorting station allocation algorithm E with LIFO selection, contrasting the results for maximum reductions against the normal behaviour with and without permitting reductions.

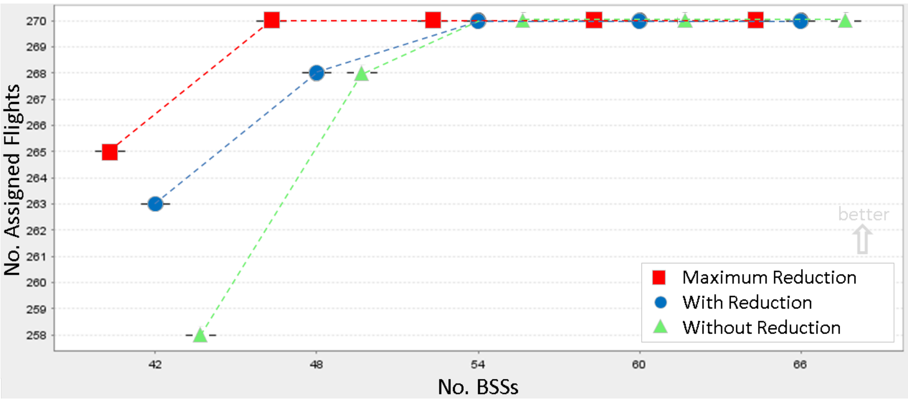

Fig. 9 Number of assignments for 270 flights for ODTLI flight ordering, sorting station allocation algorithm E with LIFO selection, contrasting the results for maximum reductions against the normal behaviour with and without permitting reductions.

Given the maximum allocation results and the obvious benefits of allowing service time reductions, the following analysis will consider only the case where reductions in service are permitted.

4.4 Comparison of Allocations With Service Reduction

Figures 10 and 11 show the number of sorting station assignments for the ODTLI and Order by Starting Time (OST) flight ordering methods, with sorting station allocation algorithms A, C and E and LIFO selection.

Figure 10 shows that ODTLI flight ordering provides a better allocation when there are fewer sorting stations, but at some point, as the number of sorting stations increases, the difference decreases and as it approaches the number necessary for optimal allocation, the OST flight ordering actually improves upon ODTLI.

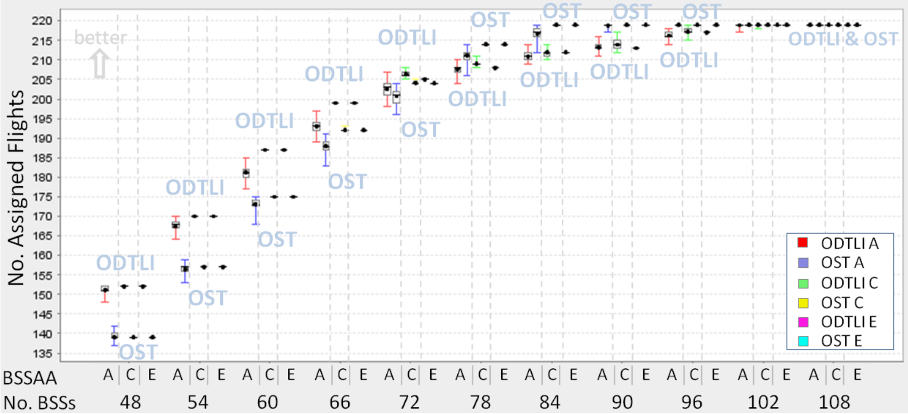

Fig. 10 Number of assignments for 219 flights for ODTLI and OST flight ordering methods, sorting station allocation algorithms A, C and E and LIFO sorting station selection, with service reduction.

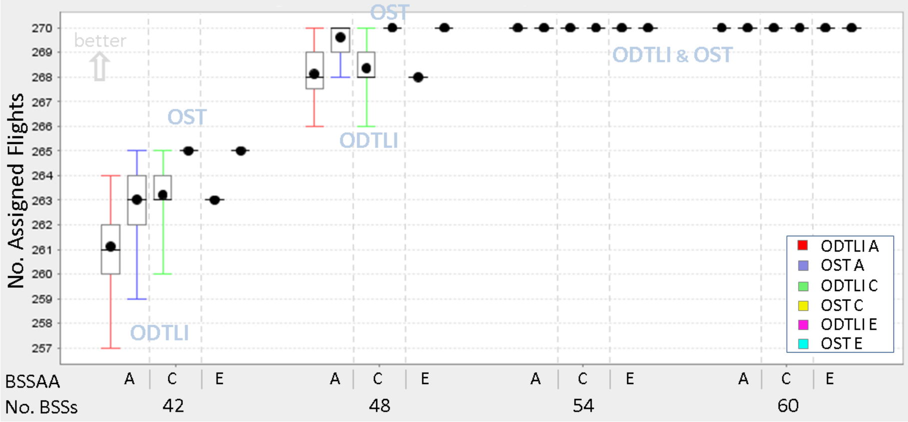

Fig. 11 Number of assignments for 270 flights for ODTLI and OST flight ordering methods, sorting station allocation algorithms A, C and E and LIFO sorting station selection, with service reduction.

For 219 flights, this point is 78 baggage sorting stations, which is close to the 84 required for maximum allocations when no buffer time is used. Since ODT and ODTLI order the flights by their departure time, where flights have similar service starting times, preference will be given to flights with shorter service times. On the other hand, the OST choice of flights could be considered to be preferring flights with longer service times (for similar departure/end of service times). Comparison of some resulting allocations showed that, perhaps counter-intuitively, not only was ODTLI failing to allocate more flights at these times, but the flights which were not allocated had longer service times than those which OST failed to allocate. Indeed, there were cases where every aircraft which OST failed to assign was a short-haul flight and every aircraft which ODTLI failed to assign was a long haul flight. The ordering appears to be important in this case. By allocating long-haul flights first, the OST algorithm is able to fit short-haul flights into the remaining gaps (with appropriate service time reductions). However, by allocating short-haul flights first the ODTLI is then unable to schedule the long-haul flights which remain, resulting in fewer allocations.

The key to understanding this behaviour is to consider the size of the remaining gaps. When there are few sorting stations, the ability of the ODTLI choice to minimise the gaps is a useful one and results in more sorting station allocations than the OST flight ordering. However, as the number of sorting stations increases, the remaining gaps start to become big enough to fit the short-haul aircraft into them and OST performs better.

Figure 11 is less interesting, since the situation was less restricted. Due to the lower density of the 270 aircraft dataset, Figure 11 effectively matches only the right hand side of Figure 10.

Further experiments showed that this behaviour was not restricted to the LIFO sorting station selection method, but also occurred for the FIFO and 'Closest' sorting station selection methods, and at the same number of sorting stations.

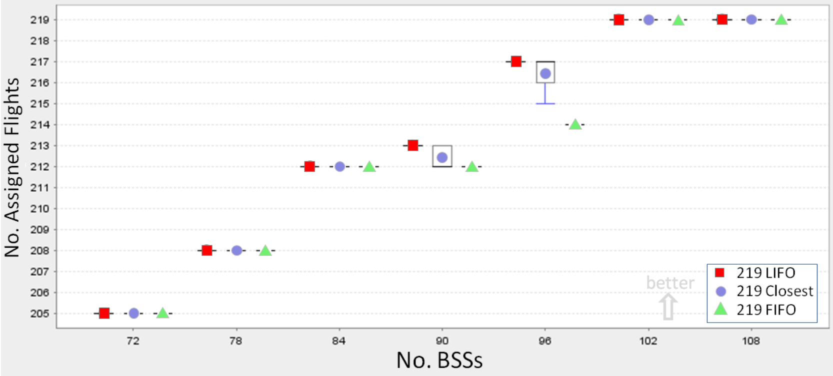

Fig. 12 Total number of sorting station assignments for 219 flights, for ODT flight ordering, sorting station allocation algorithm E, with service reduction.

Figure 12 shows the number of flights that were allocated to baggage sorting stations when ODT ordering of flights was considered and sorting station allocation algorithm E was used, with the 'Closest', First In First Out (FIFO) and LIFO sorting station selection methods, allowing a comparison to be made between the performances of the sorting station selection methods. Results are shown only for 219 flights, since this is the more restricted case and the 270 flight case has the same structure. ODT was used in this case rather than ODTLI, to avoid any influence from the look-ahead element of the algorithm, but we note that ODTLI always performed at least as well as ODT for all of these examples. It can be observed that the LIFO selection method was the best in terms of the number of sorting stations allocated (it was always at least as good as the others), followed by the 'Closest' method (which was always at least as good as the FIFO method), with the FIFO method being the worst.

4.5 Comparison of Distances With Service Reduction

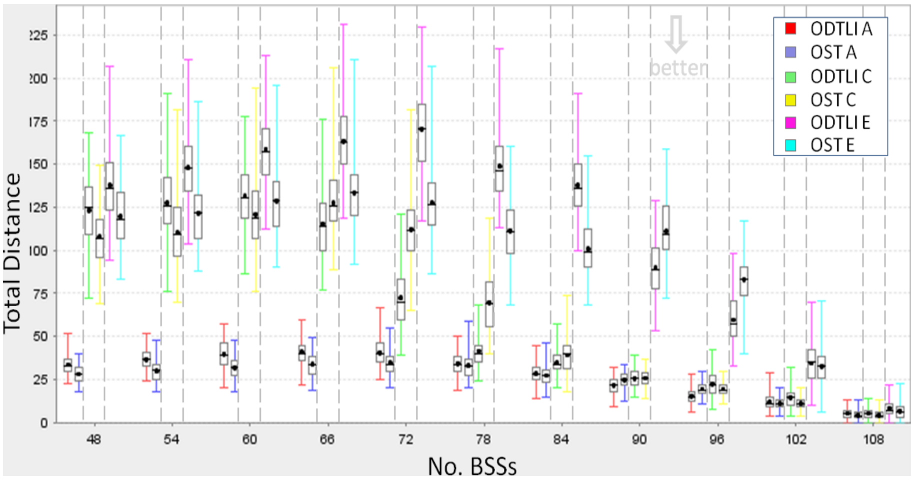

Fig. 13 Total distance results for 219 flights with ODTLI and OST flight ordering methods and 'Closest' sorting station selection, with service time reductions.

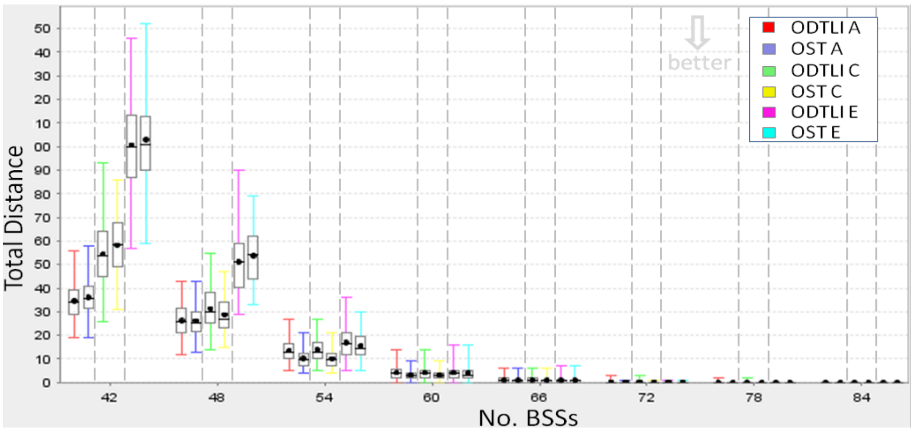

Fig. 14 Total distance results for 270 flights with ODTLI and OST flight ordering methods and 'Closest' sorting station selection, with service time reductions.

Figures 13 and 14 show the results as far as the distance reduction objective (expressed by Formula (5)) is concerned. These show the total distance between the assigned baggage sorting stations and the stand at which the flight is located. Results are shown for the three sorting station selection algorithms A, C and E, with 'Closest' sorting station selection and the ODTLI and OST flight ordering methods.

The distance basically measures the number of flights that could not be allocated to sorting stations on their preferred pier. It can be observed that the total distance decreases in all cases as the number of sorting stations is increased, since sorting stations become available on the preferred pier. Even after all flights can be allocated to sorting stations, the distances can be positive, since the availability of a sorting station at the terminal does not imply that it is on the correct pier for the flight.

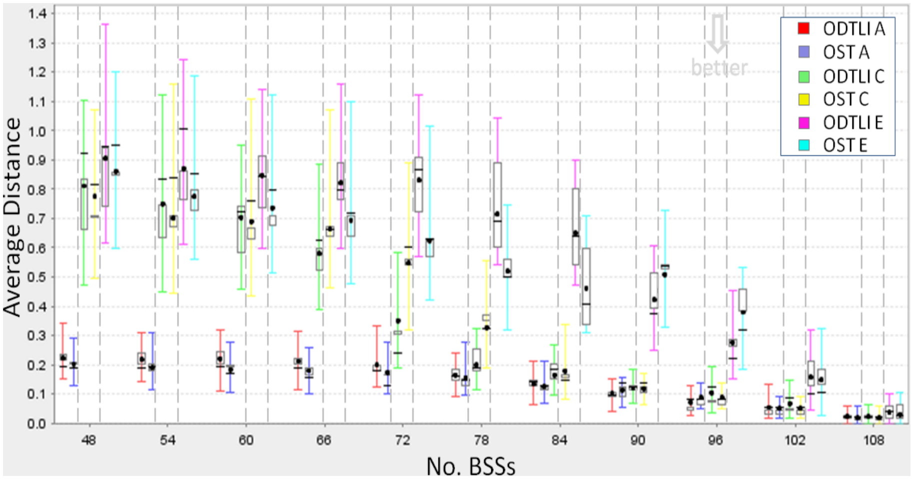

Fig. 15 Average distance for 219 flights with ODTLI and OST flight ordering methods and 'Closest' sorting station selection, with service reduction.

Algorithm A can be observed to perform better than algorithms C, and E when there is a shortage of piers. However, Algorithm E allocates more flights to sorting stations, and unallocated flights are not included, so some of the improvement could be due to the unallocated flights. Figure 15 shows the average distance per flight, to avoid this problem, and Algorithm A can be observed to also attain a lower average distance. This is expected since Algorithm A attempts to allocate to the same pier first and considers applying a service time reduction before considering other piers. For similar reasons, Algorithm C performs better than Algorithm E.

4.6 Fair Workload With Reduction in Service

It is also useful to consider the deviation of the total usage times of the sorting stations from the average usage time. This is the fairness objective which was expressed by Formula (6). Figures 16 and 17 show the total seconds deviation from the average usage across all baggage sorting stations, for allocation Algorithm E, comparing the results for the ‘Closest’, FIFO and LIFO sorting station selection methods and the ODTLI and OST flight ordering methods.

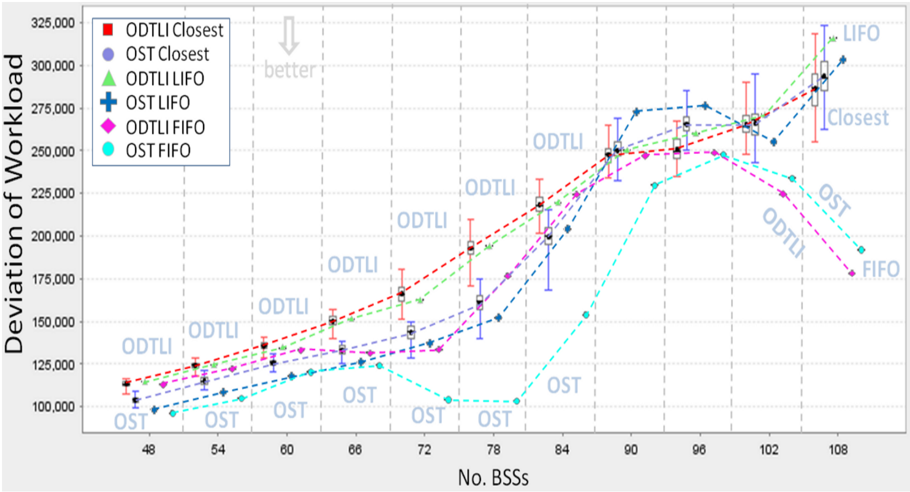

Fig. 16 Fair Workload for 219 flights with sorting station allocation algorithm E and reduction in service permitted.

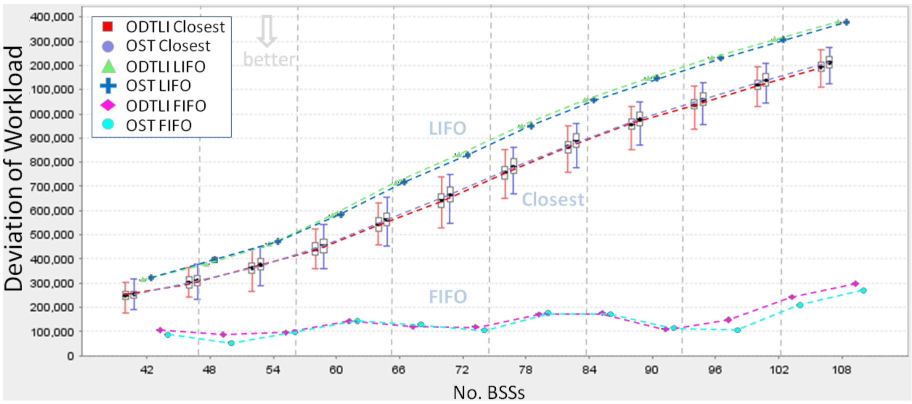

Fig. 17 Fair Workload for 270 flights with sorting station allocation algorithm E and reduction in service permitted.

The FIFO method can be considered to take this objective into account, and, indeed, consistently performs better than LIFO for both flight ordering methods.

Since the LIFO method will continue to re-use the same sorting stations where this is possible, increasing the number of sorting stations will further increase the inequity.

The FIFO method, on the other hand, will cycle through the sorting stations, giving a fairer allocation of flights to sorting stations. However, long-haul and short-haul flights are treated identically, resulting in differences in the total service times. These differences will depend upon how many of the long haul flight allocations coincide so that they are allocated to the same service stations, and a cyclic-type behaviour is observed as the number of sorting stations is increased.

4.7 Reduction in Service

Results for the reduction in service objective, expressed by Formula (7), are summarised in Figures 18 and 19 which represent the total reduction in service for all allocated flights, with differing numbers of baggage sorting stations, ODTLI and OST flight ordering and 'Closest', LIFO and FIFO baggage sorting station selection.

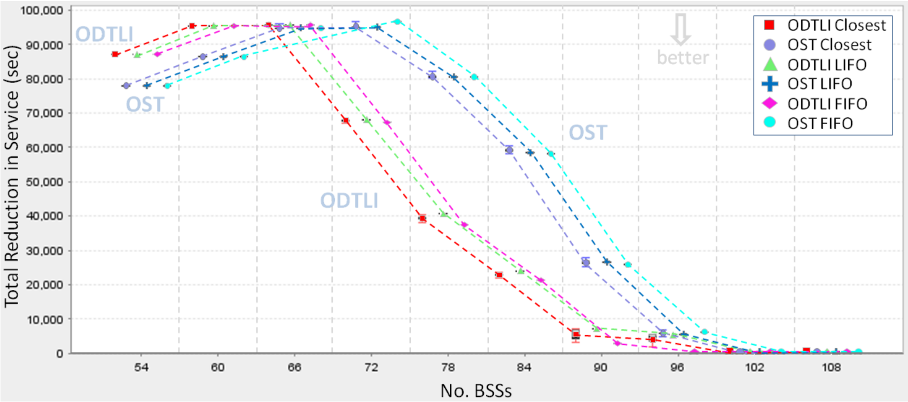

Fig. 18 Total reduction in service time for 219 flights with sorting station allocation algorithm E and reduction in service permitted.

As the number of sorting stations is increased, more of the allocations are achieved by means of the reduction in service. Thus, the seconds reduction in service increases. Eventually the point is reached where the number of sorting stations is sufficient to allow the allocations to be made with ever decreasing reductions in the service time, so the total reduction in service decreases, until eventually all flights can be allocated without any reductions in service time. Figure 20 shows the average reduction in service time for 219 flights, showing that the increase in the number of allocated flights is the origin of the initial increases in the time reduction in total service time as more sorting stations are added.

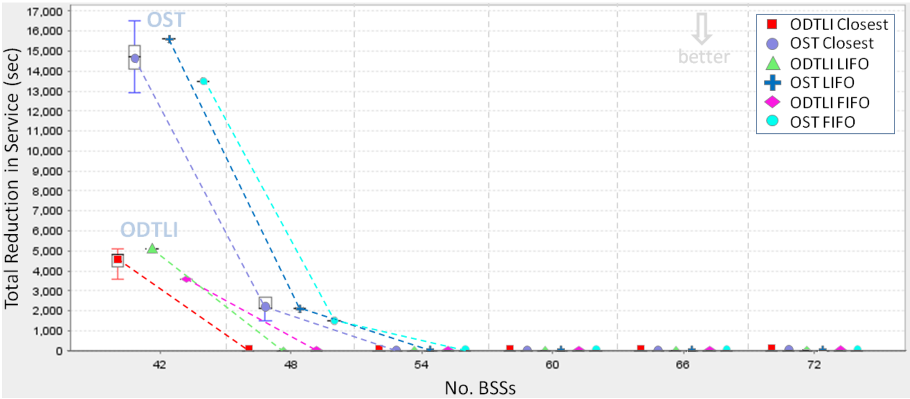

Fig. 19 Total reduction in service time for 270 flights with sorting station allocation algorithm E and reduction in service permitted.

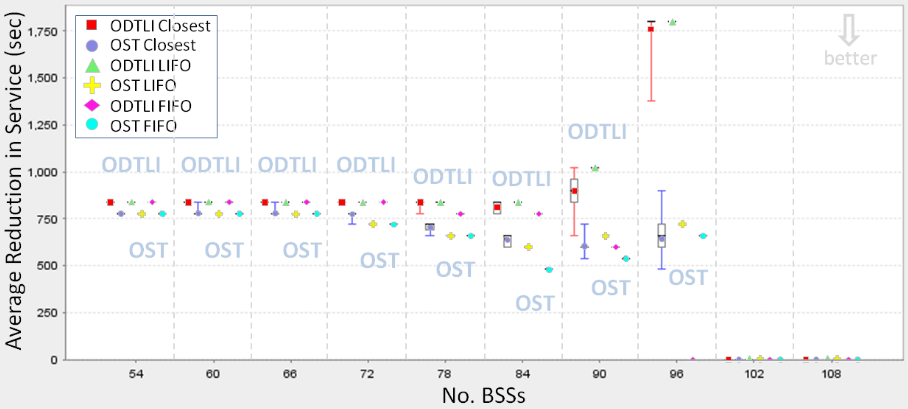

Fig. 20 Average Reduction in Services for 219 flights with sorting station allocation algorithm E and reduction in service permitted.

5 Conclusions

We have observed that (perhaps unsurprisingly) the flight density (the number of flights requiring service at any time in the day) is more important than the total number of flights when determining the number of sorting stations which are required throughout the day. We have also identified that the allocation methods (flight ordering, sorting station allocation algorithm and selection method) have different effects and can prefer different objectives. We have identified some points, in terms of the number of baggage sorting stations, where the performance of the algorithms changes and have noted that these depend upon the distribution of the flights over time. Finally, we have noted that the choice of whether or not to allow reductions in service time can affect the relative efficacy of the algorithms. In particular, if reductions in service time are to be permitted, then it may be better to select an algorithm which will not minimise the gap sizes, since these are then less likely to be usable by other flights after service time reductions have been applied. Together, these effects show that the appropriate algorithm to be used depends not just upon the objective which is under consideration but also upon the problem characteristics and the relative flight density in relation to the number of sorting stations available.

The aim of this research was not to identify the perfect constructive algorithm, but to gain insights into the differing behaviour of the algorithms, particularly when service time reductions are permitted. We plan to utilise these insights in the development of a perturbative algorithm and potentially to utilise a number of these constructive algorithms to produce high quality initial solutions according to the different objectives which are under consideration. For example, their use with metaheuristic solution methods could potentially improve the performance of such methods (in terms of convergence time even if not solution quality) by providing higher quality initial solutions.

Acknowledgements

We are grateful to National Air Traffic Services Ltd (NATS) and EPSRC for providing funding for this project, and especially John Greenwood (NATS) for his continuous support.

References

Abdelghany A, Abdelghany K, Narasimhan R (2006) Scheduling baggage-handling facilities in congested airports. Journal of Air Transport Management 12(2):76 – 81

Air Transport Users Council (2009) Annual report 2008/09. Tech. rep., AUC

Bolat A (2000) Procedures for providing robust gate assignments for arriving aircrafts. European Journal of Operational Research 120(1):63 – 80

Cormen TH, Leiserson CE, Rivest RL, Stein C (eds) (2001) Introduction to Algorithms. The Massachusetts Institute of Technology Press

Ding H, Lim A, Rodrigues B, Zhu Y (2004) Aircraft and gate scheduling optimization at airports. Hawaii International Conference on System Sciences 3:1530 – 1605

Ding H, Rodrigues AL, Zhu Y (2005) The over-constrained airport gate assignment problem. Computers and Operations Research 32(7):1867 – 1880

Federal Aviation Administration (2010) Faa aerospace forecast fiscal years 20102030 - forecast highlights. Tech. rep., Federal Aviation Administration, URL `http://www.faa.gov/data_research/aviation/aerospace_forecasts/2010-2030/`

Haralick RM, Elliott GL (1980) Increasing tree search efficiency for constraint satisfaction problems. Artificial Intelligence 14:263 – 313, URL `http://www.cs.uu.nl/docs/vakken/cons/paper_adriano.pdf`

Hassounah MI, Steuart GN (1993) Demand for aircraft gates. Transportation Research Record 1423(1423):22 – 33, URL `http://pubsindex.trb.org/view.aspx?id=390306`

ICAO (2010) Icao news release - pio 06/10. Tech. rep., International Civil Aviation Organization, URL `http://icaopressroom.files.wordpress.com/2010/07/pio-06-10-en.pdf`

Long LD, Ohsato A (2008) Fuzzy critical chain method for project scheduling under resource constraints and uncertainty. International Journal of Project Management 26:688 – 698

Mulvey JM, Vanderbei RJ, Zenios SA (1995) Robust optimization of large-scale systems. Operational Research 43:254 – 281, URL `http://coral.ie.lehigh.edu/ipseminar/robust/mulvey-et-al95.pdf`

Nikulin Y (2006) Robustness in combinatorial optimization and scheduling theory: An extended annotated bibliography. URL `http://www.optimization-online.org/DB_HTML/2004/11/995.html`

Pitt M, Wai FK, Teck PC (2002) Technology selection in airport passenger and baggage systems. Facilities 20(10):314 – 326, DOI 10.1108/02632770210442992

Rijsenbrij JC, Ottjes JA (2007) New developments in airport baggage handling systems. Transportation Planning and Technology 30(4):417 – 430

US Department of Transportation (2010) Air travel consumer report. Tech. rep., U.S. Department of Transportation, URL `http://airconsumer.dot.gov/reports/2010/April/2010AprilATCR.pdf`

Voß S, Fink A, Duin C (2005) Looking ahead with the pilot method. Annals of Operations Research 136:285 – 302, URL `http://www.smartframe.de/studentsatwork/literatur/sel_vossduin.pdf`

Wei D, Liu C (2009) Fuzzy model and optimization for airport gate assignment problem. In: IEEE International Conference on Intelligent Computing and Intelligent Systems, ICIS, IEEE, Shanghai, vol 2, pp 828 – 832, DOI 10.1109/ICICISYS.2009.5358268

Wu CL, Caves RE (2004) Modelling and optimization of aircraft turnaround time at an airport. Transportation Planning and Technology 27(1):47 – 66, DOI 10.1080/0308106042000184454

Yan S, Chang CM (1997) A network model for gate assignment. Journal of Advanced Transportation 32(2):176 – 189, DOI 10.1002/atr.5670320204

Yan S, Huo CM (2001) Optimization of multiple objective gate assignments. Transportation Research Part A: Policy and Practice 35(5):413 – 432

Yan S, Shieh CY, Chen M (2002) A simulation framework for evaluating airport gate assignments. Transportation Research Part A: Policy and Practice 36(10):885 – 898