Considering Flexibility in the Evolutionary Dynamic Optimisation of Airport Security Lane Schedules

Darren M. Chitty
Centre for Computational Intelligence,
School of Computer Science
and Informatics,
De Montfort University,
Leicester LE1 9BH, UK
Email: darrenchitty@googlemail.com

Shengxiang Yang
Centre for Computational Intelligence,
School of Computer Science
and Informatics,
De Montfort University,
Leicester LE1 9BH, UK
Email: syang@dmu.ac.uk

Mario Gongora
Centre for Computational Intelligence,
School of Computer Science
and Informatics,
De Montfort University,
Leicester LE1 9BH, UK
Email: mgongora@dmu.ac.uk

Abstract*—Airports face pressures to reduce costs at the security lane area by reducing lane opening hours whilst maintaining a passenger service level. Evolutionary methods have been shown to design schedules that minimise both objectives. However, by reducing lane opening hours schedules have a tendency to *over-fit* the expectation of passenger arrivals at security resulting in long delays with deviations from this forecast. Evolutionary dynamic re-optimisation can mitigate for this reducing passenger waiting times but the security lane problem is an example of a *constrained* problem in that schedules cannot be significantly altered. Consequently, this paper will investigate the consideration of *flexibility* when evolving initial schedules to facilitate the evolutionary dynamic re-optimisation process. Several differing methods of measuring flexibility will be investigated alongside reducing security lane opening hours and passenger waiting times. Results demonstrate that considering *flexibility* in the initial design of schedules improves the effectiveness of evolutionary dynamic re-optimisation of schedules.

I. Introduction

In the modern world reducing costs through efficiency savings is considered standard practice. This is the case for airports whereby cost savings need to be made in areas such as the security checking area. However, reducing the degree to which security lanes are open to process passengers will naturally result in longer waits for passengers passing through security. Optimisation techniques though such as evolutionary methods can treat this as a multi-objective optimisation problem and design schedules for security lanes that minimise both these objectives [5]. A problem though with this methodology is that the optimisation process will reduce security lane opening hours to match the expected passenger arrivals. Effectively, these optimised schedules *over-fit* a forecast of the numbers of passengers arriving at security throughout the day. Therefore, should passenger arrivals deviate from this forecast perhaps as a result of bad weather or a traffic accident in the vicinity of the airport, significant passenger delays can occur raising passenger dissatisfaction. Chitty *et al.* showed that this problem can be mitigated by using evolutionary methods to dynamically re-optimise these schedules by modifying shift times reducing passenger waiting times [5]. However, changing shift times is difficult as a result of the human element involved in security checks, shifts can only be modified to a minor degree. The problem of the dynamic re-optimisation of airport security lane schedules is an example of a *constrained* scheduling problem consequently limiting the effectiveness of dynamically re-optimising schedules within a changing scenario.

Therefore, it is postulated that when evolving initial security lane schedules that minimise both passenger waiting times and opening hours based on a passenger flow forecast, consideration should be given to the need to potentially dynamically re-optimise this evolved optimal schedule. Essentially, an initially optimsed schedule not only needs to reduce forecast passenger waiting times whilst also reducing lane opening but maintain a degree of *flexibility* such that in the event of unforeseen changes to passenger arrivals at security necessitating the dynamic re-optimisation of the schedule, there are a greater number of options thereby facilitating the re-optimisation process. This paper will consider several methods of measuring the *flexibility* of a security lane schedule when searching for optimal schedules and incorporating these into the effective fitness of candidate schedules. Schedules derived by using these additional *flexibility* measures will then be tested as to their effectiveness with unforeseen differing passenger arrivals necessitating the evolutionary dynamic re-optimisation of schedules to maintain optimality.

This paper is laid out as follows. Section II briefly reviews related work to optimising airport security lane schedules and consideration of flexibility in schedules. Section III describes the airport security lane optimisation problem and the evolutionary dynamic re-optimisation of schedules. Section IV introduces several methods of measuring the *flexibility* of a candidate schedule. Section V contrasts these *flexibility* measures against each other to derive the best methodology. Finally, Section VI draws conclusions and presents ideas for further work.

II. Related Work

There is limited literature associated with the optimisation of airport security lane schedules to reduce passenger waiting times. Soukour *et al.* [17] used a memetic algorithm merged with an evolutionary algorithm to assign security staff concentrating on reducing over and undertime and raising staff satisfaction. However, the security lane problem is similar to optimising airport check-in desks to minimise passenger delays and the degree to which desks are open. Wang and Chun [18] used a Genetic Algorithm (GA) [8] for optimal counter assignment for check-in desks. Chun and Mak [6] used simulation and search heuristics to determine the optimal check-in desk allocation that reduces the time desks are open and acceptable queue lengths for Hong Kong Airport. Bruno and Genovese [4] proposed a number of optimisation models for the check-in service balancing operational costs with passenger waiting times for Naples airport. Araujo and Repolho [2] present a new methodology to optimise the check-in desk allocation problem of maintaining a service level whilst reducing operational costs. Three phases are used whereby the first optimises the number of desks based upon [4], the second uses simulation to test the service level and the third uses an optimisation model to solve an adjacent desk constraint. Integer programming is used to solve both a common and dedicated desk problem. Mota [13] uses an evolutionary algorithm and a simulation approach to establish the allocation and opening times of check-in desks to reduce passenger waiting times.

The dynamic optimisation of check-in desks has been investigated by Parlar *et al.*[15], [16] with regards the optimal opening of desks to minimise a monetary cost determined as the financial cost of waiting passengers and the cost of open check-in desks and aircraft delays solved using dynamic programming for a single flight scenario. A static policy was recommended as a dynamic policy was found to suffer from the curse of dimensionality [16]. Hsu *et al.* [9] investigated the dynamic allocation of check-in facilities and passengers to desks defined as a Sequential Stochastic Assignment Problem and solved using binary integer programming with positive results. Nandhini *et al.* [14] investigated the dynamic optimisation of check-in desks to minimise the conflicting objectives of resource allocation and passenger waiting times using a GA.

With regards the consideration of the *flexibility* of schedules in dynamic environments there has been some limited work. Jensen [12] postulated that a more robust or flexible schedule may be more valuable than an inflexible optimal schedule. A robustness measure is used whereby a schedule is compared to others in its neighbourhood the premise being that a schedule open to flexibility will have many good solutions within its neighbourhood and the fitness is the average quality of solutions in this neighbourhood. Results demonstrated these schedules to be more amenable to rescheduling. Jensen also considered an alternative co-evolutionary approach for testing schedules against a *worst case* scenario [11]. Worst case schedules are evolved alongside break down scenarios for a job shop scheduling problem. This approach was found to evolve more flexible schedules. Branke and Mattfield [3] considered the use of a secondary *flexibility* measure when evolving schedules. This measure involved an *anticipation* measure of likely future changes for a job shop scheduling problem whereby avoiding early machine idle time was considered beneficial to schedules in a dynamic environment. Finally, Al-Hinai and ElMekkawy [1] used a hybrid GA to generate robust and stable schedules for the flexible job shop scheduling problem using a secondary objective to test six different robustness measure of proposed schedules based on predicted operations.

III. Dynamic Optimisation of Security Lane Schedules

A. The Security Lane Optimisation Problem

Passengers travelling by air are required to pass through stringent security checks such as hand baggage searches and passing through metal detectors etc. with a number of available security lanes for processing passengers. Security checks are staff intensive and cannot be compromised as maintaining security is paramount. One aspect of security that is open to optimisation is the schedules of opening these security lanes. Clearly, minimising passenger waiting times at security reduces passenger dissatisfaction. Thus, opening all security lanes will achieve this but to the expense of the airport but alternatively, closing lanes will increase passenger waiting times and hence increase passenger dissatisfaction. Therefore, it can be considered that the problem is multi-objective in nature, minimising waiting times and minimising security operations are mutually exclusive objectives. However, passenger demand will ebb and flow and therefore the problem becomes the design of a schedule that ensures low passenger waiting times at peak times and lower security lane opening hours at times of low passenger demand. Figure 1 demonstrates the ebb and flow of passenger demand during a 24h period with a supplied generalised forecast of passenger arrivals at an airport for four exemplar problems and an example of actual passenger flow data as a comparison.

B. Evolutionary Optimisation of Security Lane Schedules

The optimisation objective of the security lane problem is to simultaneously reduce passenger waiting times whilst also minimising the degree to which security lanes are open hence reducing costs. The two objectives are mutually exclusive hence the problem is multi-objective. The main objective is to reduce the worst passenger waiting time as defined by:

$$\text{minimise} \qquad f_1 = \max_{i \in \{1,\ldots,m\}} (W_i), \tag{1}$$

where W_i is the waiting time experienced by the i^{th} passenger at the security queue and m is the number of passengers that arrive over the time period.

The secondary objective is to minimise the degree of time to which security lanes are open during the stated time period, defined as follows:

$$\text{minimise} \qquad f_2 = \sum_{i=1}^{i \leq n} S_i, \tag{2}$$

where S_i is the time for which the i^{th} security lane shift lasts and n is the number of shifts within the schedule.

Essentially, the key objective is to minimise the maximum passenger waiting time experienced by any passenger across the whole time period. Therefore, the multi-objective problem

(a) (b) (c) (d)

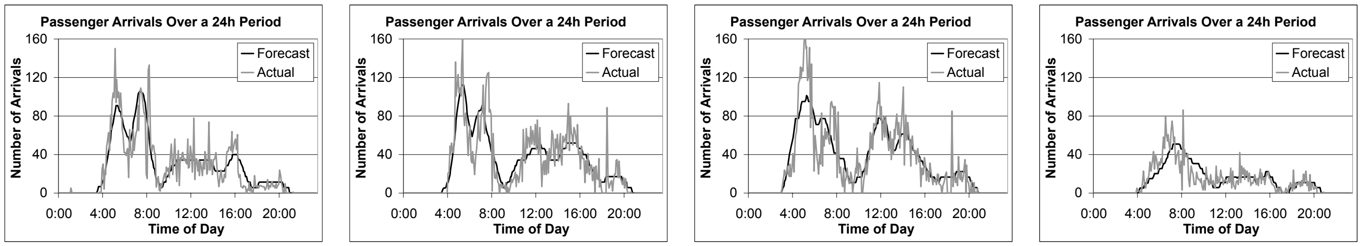

Fig. 1. The forecast arrivals of passengers at security and a set of actual passenger arrivals over a 24 hour period for the exemplar problems labelled (a) F_PAXflow_2425, (b) F_PAXflow_2428, (c) F_PAXflow_2501 and (d) F_PAXflow_21113 respectively.

can be simplified to finding the lowest maximum waiting time experienced by a passenger with the fewest hours of security lane operation. To derive the optimal security lane operational schedule an evolutionary approach can be used by deploying a GA whereby a candidate schedule is represented as a set of shifts defined by a start and finish time with a granularity of five minutes with shifts restricted to being between two and four hours in length and each gene represents a shift. Since a set of shifts constituting a schedule can be variable in nature, a variable GA approach is used [7]. Two point crossover swaps subsets of shifts between two candidate solutions with these subsets being of differing size. Mutation consists of either swapping a subset of shifts with a random replacement set or a low probability bitwise mutation of starting and finishing times of shifts. In terms of fitness selection, a candidate schedule with a lower maximum passenger waiting time is considered the fitter. If the times are identical then the schedule with the lower degree of lane operation is considered the fitter. A simulation based approach is used to measure passenger waiting times. Passengers are simulated arriving at security defined by the passenger flow forecast and enter a queue operating in a First In First Out (FIFO) manner. Open security lanes take passengers from this queue and process them which is defined as randomly taking between 15 and 21 seconds per passenger. Ten simulations are used to account passenger processing variance.

However, as evidenced in Fig. 1, passenger arrivals will often not reflect the predicted forecast with bad weather or road traffic accidents causing changes to passenger arrivals. With schedules optimised to the forecast this will likely cause significant queues with security lanes not being open, these schedules essentially *over-fit* the forecast and by minimising security lane opening hours there is no spare capacity. To address this issue a dynamic re-optimisaton approach can be used to improve these optimised schedules by modifying the shifts. In fact, resource managers often alter schedules to suit demand known as real-time shift updating [10]. However, there are constraints with this policy in that shifts due to their human component may only have their start time brought forward or pushed back by up to an hour and similarly for the finish time with shifts restricted to being between two and four hours in length. To dynamically modify security lane schedules a re-optimisation is performed every hour using the same aforementioned evolutionary approach. Forecast passenger flow is used for simulated future arrivals and actual passenger events are represented purely by the current passengers in the queue which could be much larger than expected. Full details of the approach can be found in [5].

TABLE I
GA PARAMETERS USED THROUGHOUT UNLESS OTHERWISE STATED

Population Size	100
Max Generations	2,000
Tournament Size	7
Crossover Probability	0.9
Mutation Probability	0.1
Primary Fitness Measure	Minimisation of max. passenger waiting time
Secondary Fitness Measure	Minimisation of total lane opening time

To establish the effectiveness of an evolutionary approach to the design of security lane schedules for both static and dynamic policies experiments are conducted for the four exemplar problems with results averaged over 25 random runs. Initial schedules are evolved using the forecast passenger flow information. These are then tested against the actual passenger flow information. Moreover, the dynamic re-optimisation of these schedules is also tested against actual passenger flow events. The parameters used by the evolutionary optimisation are shown in Table I.

The results in terms of maximum passenger waiting times, average passenger waiting times and total security lane opening hours from the initial evolved static schedules and the dynamically re-optimised schedules throughout the given time period are shown in Table II. It is clear to see that there are lengthy maximum passenger waiting times of over several hours for the static schedules for actual passenger arrival events. Dynamic re-optimisation of these schedules results in significant reductions in these maximum waiting times. This demonstrates how the static schedules have *over-fit* the forecast in terms of the pattern of lane opening hours matching projected peaks in passenger demand. Deviations from this forecast results in significant passenger delays. Clearly, the dynamic approach is highly effective in mitigating for the *over-fitting* issue but it can be considered that there is a limit to the degree to which it can improve passenger experiences as a result of the *constrained* nature of modifying schedules. As previously discussed, scheduled shift start and finish times can only be modified by up to plus or minus an hour with shifts remaining no longer than four hours in length.

TABLE II
THE MAXIMUM WAITING TIMES, THE AVERAGE WAITING TIMES AND THE SCHEDULED TOTAL LANE OPENING TIME USING ACTUAL PASSENGER ARRIVALS FOR A RANGE OF AVAILABLE LANES FOR THE OPTIMAL STATIC AND THE DYNAMICALLY RE-OPTIMISED SCHEDULES. RESULTS AVERAGED OVER 25 EVOLVED SCHEDULES AND 10 SIMULATIONS WITH VARYING PASSENGER PROCESSING TIMES.

	Max.	Maximum Wait (in minutes)		Average Wait (in minutes)		Shift Time (in hours)	
Problem	Lanes	Static	Dynamic	Static	Dynamic	Static	Dynamic
F_PAXflow_2425	4	161.79 ± 5.84	107.81 ± 19.43	27.32 ± 2.97	15.47 ± 3.79	38.14 ± 1.07	42.88 ± 1.55
	5	143.61 ± 8.05	114.12 ± 17.48	17.58 ± 3.08	12.18 ± 2.87	42.04 ± 2.06	45.97 ± 2.72
	6	143.25 ± 8.95	117.20 ± 15.64	15.75 ± 3.17	10.82 ± 2.40	46.74 ± 2.48	50.67 ± 2.72
	7	137.24 ± 27.23	106.31 ± 26.96	12.82 ± 3.53	8.34 ± 3.01	60.06 ± 2.40	65.38 ± 2.46
	8	139.67 ± 19.21	111.25 ± 20.81	13.22 ± 3.60	8.60 ± 2.81	64.08 ± 3.17	70.48 ± 3.68
F_PAXflow_2428	4	135.92 ± 70.40	47.70 ± 5.51	32.91 ± 10.40	13.44 ± 2.66	41.02 ± 0.78	44.03 ± 1.61
	5	62.50 ± 12.51	41.44 ± 7.56	13.67 ± 3.57	7.88 ± 1.40	44.48 ± 1.68	47.90 ± 2.06
	6	27.20 ± 1.33	25.90 ± 3.13	5.91 ± 0.60	4.97 ± 0.63	51.68 ± 2.34	53.80 ± 2.58
	7	15.35 ± 0.19	13.77 ± 2.04	2.72 ± 0.13	2.55 ± 0.18	64.10 ± 2.70	66.18 ± 2.89
	8	15.10 ± 0.71	13.38 ± 2.28	2.44 ± 0.07	2.26 ± 0.14	69.88 ± 2.47	72.67 ± 3.22
F_PAXflow_2501	4	191.79 ± 51.55	79.70 ± 26.59	57.78 ± 10.78	20.48 ± 8.13	45.12 ± 1.00	48.30 ± 1.70
	5	46.99 ± 4.85	44.00 ± 0.57	13.56 ± 3.15	8.02 ± 0.82	48.82 ± 1.81	52.70 ± 1.72
	6	25.71 ± 0.34	25.71 ± 0.34	3.78 ± 0.07	3.64 ± 0.08	65.22 ± 1.70	67.70 ± 1.87
	7	13.54 ± 0.15	13.54 ± 0.15	2.76 ± 0.11	2.57 ± 0.10	73.34 ± 3.23	76.70 ± 3.57
	8	12.17 ± 1.07	10.40 ± 1.13	2.20 ± 0.06	2.08 ± 0.07	83.34 ± 3.07	87.66 ± 3.60
F_PAXflow_21113	4	8.16 ± 0.15	8.16 ± 0.15	2.05 ± 0.05	1.95 ± 0.06	35.20 ± 1.45	36.87 ± 1.88
	5	7.48 ± 0.74	7.48 ± 0.74	1.54 ± 0.04	1.48 ± 0.04	44.64 ± 1.78	47.38 ± 2.07
	6	7.19 ± 0.91	7.19 ± 0.91	1.30 ± 0.04	1.26 ± 0.04	53.56 ± 2.21	56.21 ± 2.69
	7	6.95 ± 1.20	6.95 ± 1.20	1.19 ± 0.04	1.13 ± 0.05	60.08 ± 2.83	63.05 ± 3.35
	8	6.09 ± 0.83	6.09 ± 0.83	1.08 ± 0.04	1.02 ± 0.04	66.82 ± 4.66	70.91 ± 5.07

Therefore, if peaks in passenger demand differ significantly from the forecast and the schedule *over-fits* this forecast then evolutionary dynamic re-optimisation of schedules will likely only partially mitigate for this problem.

IV. MEASURING THE FLEXIBILITY OF SCHEDULES

It can be considered that a highly optimised initial schedule of security lane shifts will likely *over-fit* a forecast of expected passenger arrivals due to the secondary objective of minimising the total number of shift hours. With differing passenger arrivals the schedule will perform badly and due to the *constrained* nature of schedules it can be theorised that evolutionary dynamic re-optimisation will not achieve an optimal *schedule*. Therefore, to improve the performance of the re-optimisation process it would be beneficial to have an initial schedule that is more amenable to modification. Essentially, when evolving the initial schedules, a candidate schedule needs, in addition to its primary objectives, to be assessed in terms of its ability to deal with unforeseen passenger arrivals or its *flexibility*. Indeed, Branke and Mattfeld also postulated that in dynamic, changing environments the *flexibility* of schedules should be of consideration in the evolutionary process [3].

A simple example can demonstrate the concept. Consider an initial schedule with four shifts all starting at the same time and lasting the same amount of time. This could be considered ineffective as each of these shifts can only be modified within the same time zone. An improved shift layout might be to slightly offset the shifts, effectively stagger them such that as a whole the four shifts have a greater degree of coverage if required. An additional point to consider is the length of shifts. A few shifts of four hours in length, the maximum allowed, provides less potential options for re-optimisation than a greater number of shifts of only two or three hours in length as four hour shifts cannot be increased in length, only the start time of the shift can be modified.

Therefore, some additional *flexibility* measures will be introduced to try and reflect some of the aforementioned issues with the initially evolved schedules in order to make them more amenable to evolutionary dynamic re-optimisation.

A. Measuring the Maximum Number of Lane Hours

One method of describing the flexibility or amenability of a schedule to evolutionary dynamic re-optimisation is to sum the total number of hours that the security lanes can be operated throughout the time period including both the current statically defined shifts and the dynamic aspects. Essentially, at each five minute interval in the schedule (as defined by the granularity) count the number of the available security lanes that could be opened by a shift in the current schedule including a maximum dynamic modification. Thus, if at the current time interval a shift by having its start time brought forward by up to an hour can then be opened within this time period then a security lane can be considered as being capable of being opened. The number of lanes that can be opened in a time period is obviously restricted to the maximum number of available lanes. This flexibility measure will be henceforth referred to as *MaxLaneCoverage* and this third optimisation objective can be described as:

$$\text{maximise} \qquad f_3 = \sum_{t=1}^{t \leq T} N_t, \tag{3}$$

where N_t is the total number of lanes that can be opened at the five minute interval time period t in the schedule with T the number of five minute intervals within the time period.

B. Measuring the Scope of a Schedule

An alternative measure to testing the flexibility of a given schedule is to consider minimising the number of five minute intervals within a given schedule whereby there is no lane coverage at this time even when all the shifts in the initial static schedule are dynamically modified to their full extent. The reasoning behind this measure is that long delays can often happen because passengers unexpectedly arrive at a time when there are no security lanes open meaning they have to simply wait in the queue until a security lane finally opens. This flexibility measure will be henceforth referred to as *Unopenable* and this objective can be described as:

$$\text{minimise} \qquad f_3 = \sum_{t=1}^{t \leq T} C_t, \tag{4}$$

where T is the number of five minute intervals within the given time period and C_t denotes if no security lanes can be opened at time interval t described as:

$$C_t = \begin{cases} 1 & \text{if no lane can be opened at time } t \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

However this flexibility measure does not consider situations whereby only a single security lane can be opened as opposed to up to eight lanes. Therefore, a second methodology of measuring the degree of security lanes that cannot be opened at a given time period is proposed. In this instance, the number of the available security lanes that cannot possibly be opened taking into consideration the dynamic modification of shifts are counted at each five minute interval in a schedule. Moreover, in order to more greatly penalise schedules that have a high occurrence of time periods whereby most lanes cannot be opened even with dynamic modification to the schedule, the number of unopenable lanes at a given time period is raised to a power of two. This flexibility measure will be referred to henceforth as *UnopenableLanes* and this optimisation objective can be described as:

$$\text{minimise} \qquad f_3 = \sum_{t=1}^{t \leq T} U_t^2, \tag{6}$$

where U_t is the number of available security lanes that cannot possibly be opened at the given time period and T is the number of five minute time intervals in the given schedule.

C. Measuring the Average Shift Length in a Schedule

A final methodology of measuring the flexibility of an evolved initial schedule is to simply consider the number of shifts in the schedule or the average shift length. Consider that a schedule could consist of say twenty two hour shifts or ten four hours shifts. However, recall that shifts must be within two and four hours in length and their start and finish times can be modified by up to one hour. Therefore, in effect the first schedule with more shifts can actually double the amount of shift coverage whereas the second can not increase the shift coverage at all but merely move the currently scheduled shifts. Moreover, a greater number of shifts in a schedule provides more options for the reconfiguration of a schedule when exposed to evolutionary dynamic re-optimisation. Consequently, a final flexibility measure is considered whereby the aim is to reduce the average length of shifts within a given schedule. This flexibility measure will be henceforth referred to as *AverageShiftLength* and this optimisation objective can be described as:

$$\text{minimise} \qquad f_3 = \frac{\sum_{i=1}^{i \leq n} S_i}{n}, \tag{7}$$

where S_i is the time for which the i^{th} security lane shift lasts and n is the number of shifts within the schedule.

V. Comparing the Effectiveness of the Flexibility Measures

To gauge the effectiveness of the aforementioned four proposed flexibility measures the experiments with evolutionary dynamic re-optimisation of airport security lane schedules will be repeated. However, the fitness function for evolving the initial static schedules for the opening of security lanes will now consist of three optimisation objectives. The first being the minimisation of the maximum passenger waiting time at security as recorded by the simulation. The second objective being the minimisation of the number of hours that the security lanes are open. The third objective will consist of one of the four flexibility measures as described in Section IV. As previously when evolving security lane schedules, when comparing two candidate schedules to establish the fitter of the two, the schedule with the lowest maximum passenger waiting time is considered the fitter as this is the priority objective. However, if the two schedules have the same maximum passenger wait then the candidate schedule with the lower number of security lane opening hours is considered the fitter. However, if both schedules have the same maximum waiting time and scheduled opening hours then the schedule with the better flexibility measure will be considered the fitter.

As previously, experiments will be conducted by evolving 25 differing schedules and then dynamically re-optimising them. Ten simulations will be performed to establish the fitness of candidate schedules. The results for each flexibility measure in terms of the maximum passenger waiting times for schedules exposed to differing actual passenger arrivals at security and dynamically re-optimised are shown in Table III.

From these results, the first aspect to note is that the average maximum passenger waiting times recorded for each flexibility measure across all problems and ranges of security lane availability are broadly similar. However, there are minor improvements over the results in Table II whereby no flexibility measure is used. Thus, the advantages from using a *flexibility* measure to improve the evolved initial schedules amenability to dynamic re-optimisation are only subtle. This though is to be expected as the number of hours security lanes

TABLE III
THE MAXIMUM PASSENGER WAITING TIMES FOR DYNAMICALLY RE-OPTIMISED SCHEDULES THAT WERE INITIALLY EVOLVED USING A RANGE OF *flexibility* MEASURES. RESULTS AVERAGED OVER 25 EVOLVED SCHEDULES AND 10 SIMULATIONS WITH VARYING PASSENGER PROCESSING TIMES. BEST RESULTS HIGHLIGHTED IN BOLD.

Problem	Max. Lanes	Maximum Passenger Waiting Time for Re-Optimised Schedules (in minutes)			
		MaxLaneCoverage	*Unopenable*	*UnopenableLanes*	*AverageShiftLength*
F_PAXflow _2425	4	117.22 ± 15.60	112.42 ± 21.74	**105.03** ± 23.46	109.42 ± 23.46
	5	110.99 ± 18.72	114.12 ± 17.49	112.49 ± 18.27	**110.96** ± 18.75
	6	115.61 ± 16.75	**106.21** ± 19.60	115.64 ± 16.69	110.94 ± 18.78
	7	110.85 ± 18.91	109.28 ± 19.30	107.70 ± 19.55	**106.14** ± 19.67
	8	110.84 ± 18.91	110.85 ± 18.91	110.86 ± 18.90	**104.96** ± 21.64
F_PAXflow _2428	4	49.76 ± 4.05	46.95 ± 4.03	46.67 ± 3.51	**46.29** ± 3.45
	5	46.63 ± 9.60	42.03 ± 7.61	**39.82** ± 6.22	41.83 ± 6.17
	6	28.09 ± 0.78	26.35 ± 2.38	**25.49** ± 3.25	26.16 ± 2.81
	7	15.22 ± 1.08	14.13 ± 1.81	**13.69** ± 1.86	13.98 ± 1.86
	8	15.35 ± 0.67	13.98 ± 1.59	**13.49** ± 2.23	13.56 ± 2.18
F_PAXflow _2501	4	74.91 ± 1.78	75.01 ± 1.74	**74.83** ± 1.56	75.83 ± 9.79
	5	**43.96** ± 0.41	43.98 ± 0.44	**43.96** ± 0.41	43.97 ± 0.43
	6	25.75 ± 0.31	25.75 ± 0.31	**25.71** ± 0.34	25.73 ± 0.33
	7	13.54 ± 0.15	13.54 ± 0.15	13.54 ± 0.15	13.54 ± 0.15
	8	**10.06** ± 1.05	10.43 ± 1.51	10.26 ± 1.00	10.29 ± 1.22
F_PAXflow _21113	4	8.43 ± 0.09	8.15 ± 0.17	8.16 ± 0.15	**8.12** ± 0.31
	5	7.66 ± 0.71	**7.47** ± 0.76	7.54 ± 0.74	7.70 ± 0.72
	6	7.55 ± 0.89	7.05 ± 0.87	7.21 ± 0.88	**6.92** ± 0.81
	7	7.99 ± 0.82	7.29 ± 1.14	**6.90** ± 1.33	**6.90** ± 1.33
	8	6.33 ± 0.71	**5.88** ± 0.96	6.09 ± 0.83	6.02 ± 0.88

are open will not change to a great degree and maximum passenger waiting time is most likely dominated by points in peak demand whereby all lanes need to be open.

The best improvements in the maximum passenger waiting times achieved by the *flexibility* measures are predominately from the *UnopenableLanes* measure which quadratically increases the penalty associated with the number of security lanes that cannot be opened. This measure seems to demonstrate that good throughput of passengers can reduce congestion and long delays and this can be achieved by reducing time periods whereby there are few security lanes available to be opened when taking into account dynamic modification. Furthermore, when comparing to the results in Table II, improvements in the maximum passenger waiting time are achieved in most cases. The other *flexibility* measure that shows promise is the *AverageShiftLength* measure. As previously theorised, reducing the average shift length increases the number of shifts that can be evolutionary dynamically re-optimised providing greater scope and by shifts being shorter, a greater increase in security lane opening time can be achieved to counter unexpected peaks in demand.

It should be stated that the improvements in the maximum passenger waiting times for the dynamically re-optimised schedules were achieved with no appreciable increase in security lane opening hours over those described in Table II whereby no flexibility measure is used.

A. Combining Multiple Flexibility Measures

From Table III it was clear that the two most effective flexibility measures in terms of reducing the maximum passenger waiting time in unforeseen actual passenger flow circumstances were the *UnopenableLanes* and *AverageShiftLength* measures. Therefore, it could be considered useful to combine both measures into the fitness function. Two methods will be considered. The first will extend the fitness function to consist of four objectives. As previously, the primary objective is the minimisation of the maximum passenger waiting times and the second the minimisation of the security lane opening hours. The third objective of minimising the *AverageShiftLength* is used to differentiate if the first two objective values are the same for two candidate schedules being compared. If all three objective values are the same between the two candidate schedules being compared then the fourth objective comes into effect, the minimisation of the *UnopenableLanes* flexibility measure.

The results from using both *flexibility* measures as fitness objectives are shown in Table IV. Comparing these results to those in Table II it can be observed that in most cases there has been a small reduction in the maximum passenger waiting times for both static and dynamically re-optimised schedules when using the combined flexibility measure. Indeed, slightly better reductions are observed for the average passenger waiting times of approximately 2-3% although in some cases as much as 10%. Moreover, there has only been a small increase in security lane opening time of a few minutes on average. However, it can be considered that the effect of the *flexibility* measures has been minimal.

A second alternative methodology of combining the *UnopenableLanes* and *AverageShiftLength* flexibility measures is to simply limit the evolutionary algorithm to allocating shifts at the minimum two hour length for the initial shift. This achieves the minimum average shift length but there is an issue

TABLE IV

THE MAXIMUM WAITING TIMES, THE AVERAGE WAITING TIMES AND THE SCHEDULED TOTAL LANE OPENING TIME USING ACTUAL PASSENGER ARRIVALS FOR A RANGE OF AVAILABLE LANES FOR STATIC AND DYNAMICALLY RE-OPTIMISED SCHEDULES USING BOTH THE *AverageShiftLength* AND *Unopenable* FLEXIBILITY MEASURES. RESULTS AVERAGED OVER 25 EVOLVED SCHEDULES AND 10 SIMULATIONS WITH VARYING PASSENGER PROCESSING TIMES.

Problem	Max. Lanes	Maximum Wait (in minutes) Static	Maximum Wait (in minutes) Dynamic	Average Wait (in minutes) Static	Average Wait (in minutes) Dynamic	Shift Time (in hours) Static	Shift Time (in hours) Dynamic
F_PAXflow_2425	4	154.03 ± 17.42	105.84 ± 19.67	27.07 ± 4.27	15.16 ± 3.15	38.28 ± 1.27	42.97 ± 1.77
	5	138.38 ± 15.85	111.31 ± 18.70	16.50 ± 3.88	11.54 ± 3.01	41.68 ± 2.11	46.22 ± 2.56
	6	140.44 ± 26.20	106.41 ± 26.01	14.83 ± 3.71	9.70 ± 3.14	47.03 ± 3.00	51.21 ± 3.29
	7	137.51 ± 29.76	99.65 ± 27.96	12.13 ± 3.46	7.52 ± 3.08	57.62 ± 2.68	63.56 ± 2.80
	8	142.23 ± 8.64	111.24 ± 18.80	12.46 ± 3.03	8.20 ± 2.67	64.58 ± 2.79	70.73 ± 3.21
F_PAXflow_2428	4	112.59 ± 66.71	47.06 ± 5.11	29.53 ± 10.33	12.37 ± 2.22	41.45 ± 1.34	45.28 ± 1.55
	5	63.50 ± 13.52	41.87 ± 7.60	13.30 ± 2.24	7.80 ± 1.36	45.00 ± 1.33	48.77 ± 1.57
	6	27.26 ± 0.76	25.94 ± 2.44	6.09 ± 0.47	4.90 ± 0.54	51.67 ± 2.32	54.26 ± 2.75
	7	15.00 ± 1.14	13.88 ± 1.89	2.71 ± 0.13	2.56 ± 0.16	63.94 ± 2.26	66.22 ± 2.76
	8	14.69 ± 1.03	13.26 ± 1.78	2.39 ± 0.10	2.21 ± 0.12	70.75 ± 2.80	73.65 ± 3.12
F_PAXflow_2501	4	197.19 ± 50.81	74.85 ± 1.46	59.32 ± 9.15	18.48 ± 2.31	44.93 ± 0.53	48.57 ± 1.17
	5	46.05 ± 3.91	43.96 ± 0.41	12.17 ± 2.71	7.76 ± 0.60	49.27 ± 1.57	52.95 ± 1.62
	6	25.72 ± 0.33	25.72 ± 0.33	3.78 ± 0.07	3.61 ± 0.10	65.20 ± 1.62	67.88 ± 1.85
	7	13.54 ± 0.15	13.54 ± 0.15	2.79 ± 0.09	2.57 ± 0.10	72.07 ± 2.81	75.66 ± 3.07
	8	12.09 ± 1.37	9.95 ± 0.95	2.22 ± 0.06	2.08 ± 0.06	82.80 ± 3.67	87.42 ± 3.98
F_PAXflow_21113	4	8.13 ± 0.19	8.13 ± 0.19	2.04 ± 0.19	1.93 ± 0.19	34.87 ± 1.15	36.56 ± 1.40
	5	7.60 ± 0.73	7.60 ± 0.73	1.57 ± 0.73	1.49 ± 0.73	44.15 ± 2.45	47.50 ± 2.93
	6	7.05 ± 0.85	7.05 ± 0.85	1.30 ± 0.04	1.24 ± 0.04	54.07 ± 2.59	57.42 ± 2.82
	7	6.83 ± 1.42	6.83 ± 1.42	1.18 ± 0.05	1.13 ± 0.05	60.61 ± 2.84	63.89 ± 3.27
	8	6.23 ± 0.74	6.23 ± 0.74	1.08 ± 0.03	1.01 ± 0.03	68.70 ± 3.23	72.95 ± 3.80

TABLE V

THE MAXIMUM WAITING TIMES, THE AVERAGE WAITING TIMES AND THE SCHEDULED TOTAL LANE OPENING TIME USING ACTUAL PASSENGER ARRIVALS FOR A RANGE OF AVAILABLE LANES FOR STATIC AND DYNAMICALLY RE-OPTIMISED SCHEDULES EVOLVED USING FIXED INITIAL SHIFT LENGTHS OF TWO HOURS AND THE *Unopenable* FLEXIBILITY MEASURE. RESULTS AVERAGED OVER 25 EVOLVED SCHEDULES AND 10 SIMULATIONS WITH VARYING PASSENGER PROCESSING TIMES.

Problem	Max. Lanes	Maximum Wait (in minutes) Static	Maximum Wait (in minutes) Dynamic	Average Wait (in minutes) Static	Average Wait (in minutes) Dynamic	Shift Time (in hours) Static	Shift Time (in hours) Dynamic
F_PAXflow_2425	4	129.67 ± 18.09	95.18 ± 16.79	20.13 ± 3.19	11.73 ± 2.37	41.20 ± 1.27	46.50 ± 1.78
	5	137.42 ± 21.45	95.62 ± 19.23	13.81 ± 3.13	8.32 ± 2.56	43.92 ± 1.92	50.74 ± 2.33
	6	145.50 ± 4.20	95.13 ± 16.82	12.48 ± 2.38	7.12 ± 2.51	49.76 ± 2.29	56.16 ± 2.76
	7	144.83 ± 5.79	91.85 ± 14.50	10.85 ± 2.24	5.77 ± 2.13	61.84 ± 2.47	70.02 ± 3.05
	8	138.23 ± 17.64	96.58 ± 17.76	10.53 ± 3.21	6.17 ± 2.55	67.68 ± 3.74	76.19 ± 4.26
F_PAXflow_2428	4	88.49 ± 44.90	45.01 ± 2.88	24.60 ± 6.97	10.56 ± 1.24	44.40 ± 0.98	49.96 ± 1.15
	5	52.02 ± 14.15	32.79 ± 6.21	11.08 ± 2.28	6.14 ± 0.81	47.60 ± 1.60	52.01 ± 1.95
	6	25.91 ± 2.05	21.07 ± 3.74	5.28 ± 0.77	3.80 ± 0.43	55.84 ± 1.96	59.78 ± 2.17
	7	13.56 ± 1.54	11.26 ± 1.82	2.49 ± 0.16	2.23 ± 0.14	68.88 ± 2.54	72.87 ± 3.08
	8	13.01 ± 1.56	9.84 ± 1.55	2.20 ± 0.10	1.97 ± 0.06	76.08 ± 2.23	81.09 ± 2.69
F_PAXflow_2501	4	105.28 ± 19.75	74.65 ± 0.57	40.15 ± 8.73	16.48 ± 1.33	48.56 ± 1.06	53.38 ± 1.42
	5	45.17 ± 1.36	43.96 ± 0.41	10.69 ± 1.41	6.96 ± 0.47	53.04 ± 1.80	57.72 ± 1.83
	6	25.59 ± 0.37	25.59 ± 0.37	3.62 ± 0.12	3.38 ± 0.13	69.20 ± 1.96	73.27 ± 2.26
	7	13.54 ± 0.15	13.54 ± 0.15	2.66 ± 0.08	2.40 ± 0.07	76.96 ± 1.71	82.20 ± 2.33
	8	11.67 ± 1.48	9.66 ± 0.80	2.13 ± 0.04	1.95 ± 0.05	86.96 ± 2.54	93.28 ± 2.78
F_PAXflow_21113	4	8.04 ± 0.23	8.06 ± 0.22	1.88 ± 0.06	1.75 ± 0.06	38.48 ± 1.63	41.94 ± 2.08
	5	7.17 ± 0.93	7.17 ± 0.93	1.46 ± 0.05	1.36 ± 0.06	48.32 ± 1.67	53.09 ± 2.07
	6	6.54 ± 1.44	6.54 ± 1.44	1.23 ± 0.05	1.15 ± 0.05	57.75 ± 2.55	62.90 ± 2.85
	7	5.85 ± 1.61	5.85 ± 1.61	1.12 ± 0.04	1.05 ± 0.04	63.52 ± 2.55	68.96 ± 2.92
	8	4.78 ± 0.78	4.78 ± 0.78	1.04 ± 0.04	0.96 ± 0.04	69.44 ± 3.52	75.39 ± 3.69

in this approach in that the optimised schedules will not be able to precisely fit security lane opening hours to the forecast of passenger arrivals. A security lane for instance cannot be opened for three hours, the evolved schedules will consist of a coarser granularity. In terms of the fitness function, the three objectives of minimising maximum passenger waiting time, minimising security lane opening hours and the *Unopenable-Lanes* flexibility measure will be used.

Results are shown in Table V whereby it can be clearly observed that the passenger waiting times are significantly lower using this approach for both static and dynamically re-optimised schedules. However, it can also be seen that there has been an increase in the security lane opening hours of approximately 10% which will naturally enable a reduction in passenger queuing times. Restricting shifts to being two hours in length for the initial schedules means that a schedule of shifts cannot precisely fit the forecast of passenger arrivals resulting in some extra shift hours. Effectively, this approach naturally reduces the *over-fitting* of the evolutionary optimisation process to the forecast of passenger arrivals. Therefore, this methodology must be considered by an airport if the 10% increase in security lane opening hours is worth the greater capacity to deal with unexpected passenger arrivals otherwise the former methodology is the better to use.

VI. Conclusions

This paper has investigated the consideration of *flexibility* when evolving schedules for a dynamic environment which also suffers from *constraints* restricting dynamic modifications. Evolutionary processes were applied to the design of schedules for the airport security lane problem whereby the goal is to both reduce passenger waiting times and security lane opening hours. However, evolved schedules were found to *over-fit* forecasts of passenger arrivals such that if arrivals deviate from this forecast long delays occur. Evolutionary dynamic re-optimisation was found to greatly mitigate for this issue but the effect can be limited due to the *constraints* in that scheduled shifts can be modified to a great extent.

Therefore, it was hypothesized that the potential *flexibility* or amenability of initially evolved schedules needs to be taken into consideration in order to aid the dynamic re-optimisation of schedules if required. Consequently, four differing *flexibility* measures were presented and tested. These were found to have a small positive effect on reducing the maximum passenger waiting times. Furthermore, it was considered that two best performing measures could be combined for maximum effect, the minimisation of the average shift length and the reduction of the number of security lanes that cannot be opened even when taking into account dynamic modifications. Moreover, limiting shifts in initial schedules to the minimum permissible length provided the best results by helping to reduce the extent of schedules *over-fitting* the forecast passenger arrivals.

Further work could consist of more sophisticated *flexibility* measures and analysis of sets of evolved initial schedules that perform well on the forecast of passenger arrivals but not for actual passenger arrivals when dynamically re-optimised. Schedules could also be tested directly to their amenability to dynamic re-optimisation for hypothetical differing passenger arrivals although this could be computationally expensive. Consideration could also be given to *robustness* by using multiple differing forecasts of passenger arrivals.

Acknowledgment

This work was supported by the Engineering and Physical Sciences Research Council (EPSRC) of U.K. under grant EP/K001310/1.

References

[1] N. Al-Hinai and T. ElMekkawy, "Robust and stable flexible job shop scheduling with random machine breakdowns using a hybrid genetic algorithm," *International Journal of Production Economics*, vol. 132, no. 2, pp. 279–291, 2011.

[2] G. E. Araujo and H. M. Repolho, "Optimizing the airport check-in counter allocation problem," *Journal of Transport Literature*, vol. 9, no. 4, pp. 15–19, 2015.

[3] J. Branke and D. C. Mattfeld, "Anticipation and flexibility in dynamic scheduling," *International Journal of Production Research*, vol. 43, no. 15, pp. 3103–3129, 2005.

[4] G. Bruno and A. Genovese, "A mathematical model for the optimization of the airport check-in service problem," *Electronic Notes in Discrete Mathematics*, vol. 36, pp. 703–710, 2010.

[5] D. M. Chitty, M. Gongora, and S. Yang, "Evolutionary dynamic optimisation of airport security lane schedules," in *2016 IEEE Symposium Series on Computational Intelligence (SSCI)*, Dec 2016, pp. 1–8.

[6] H. W. Chun and R. W. T. Mak, "Intelligent resource simulation for an airport check-in counter allocation system," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 29, no. 3, pp. 325–335, 1999.

[7] I. Harvey, "Species adaptation genetic algorithms: A basis for a continuing saga," in *Toward a Practice of Autonomous Systems: Proceedings of the First European Conference on Artificial Life*, 1992, pp. 346–354.

[8] J. H. Holland, *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence.* U Michigan Press, 1975.

[9] C.-I. Hsu, C.-C. Chao, and K.-Y. Shih, "Dynamic allocation of check-in facilities and dynamic assignment of passengers at air terminals," *Computers & Industrial Engineering*, vol. 63, no. 2, pp. 410–417, 2012.

[10] D. Hur, V. A. Mabert, and K. M. Bretthauer, "Real-time work schedule adjustment decisions: An investigation and evaluation," *Production and Operations Management*, vol. 13, no. 4, pp. 322–339, 2004.

[11] M. T. Jensen, "Finding worst-case flexible schedules using coevolution," in *Proceedings of the 3rd Annual Conference on Genetic and Evolutionary Computation*. Morgan Kaufmann Publishers Inc., 2001, pp. 1144–1151.

[12] M. T. Jensen, "Generating robust and flexible job shop schedules using genetic algorithms," *IEEE Transactions on evolutionary computation*, vol. 7, no. 3, pp. 275–288, 2003.

[13] M. A. M. Mota and C. Zuniga, "A simulation-evolutionary approach for the allocation of check-in desks in airport terminals," in *ATOS 2013, 4th International Air Transport and Operations Symposium*, 2013.

[14] M. Nandhini, K. Palanivel, and S. Oruganti, "Optimization of airport check-in service scheduling," 2012.

[15] M. Parlar and M. Sharafali, "Dynamic allocation of airline check-in counters: a queueing optimization approach," *Management Science*, vol. 54, no. 8, pp. 1410–1424, 2008.

[16] M. Parlar, B. Rodrigues, and M. Sharafali, "On the allocation of exclusive-use counters for airport check-in queues: static vs. dynamic policies," *Opsearch*, vol. 50, no. 3, pp. 433–453, 2013.

[17] A. A. Soukour, L. Devendeville, C. Lucet, and A. Moukrim, "A memetic algorithm for staff scheduling problem in airport security service," *Expert Systems with Applications*, vol. 40, no. 18, pp. 7504–7512, 2013.

[18] B. Wang and H. Chun, "Check-in counter allocation using genetic algorithm," in *Proc. Of the 7th Conf. on Artificial Intelligence and Expert Systems Applications*, 1995.