Transportation Research Part C 91 (2018) 296–316

Contents lists available at ScienceDirect

Transportation Research Part C

journal homepage: www.elsevier.com/locate/trc

Preference-based evolutionary algorithm for airport surface operations

Michal Weiszer[a], Jun Chen[a,*], Paul Stewart[b], Xuejun Zhang[c]

[a] School of Engineering and Materials Science, Queen Mary University of London, UK
[b] Institute for Innovation in Sustainable Engineering, Derby, UK
[c] School of Electronic and Information Engineering, Beihang University and National Key Laboratory of CNS/ATM, Beijing, PR China

A R T I C L E I N F O

Keywords:
Airport ground operations
Runway scheduling
Multiobjective optimisation
Preference search

A B S T R A C T

In addition to time efficiency, minimisation of fuel consumption and related emissions has started to be considered by research on optimisation of airport surface operations as more airports face severe congestion and tightening environmental regulations. Objectives are related to economic cost which can be used as preferences to search for a region of cost efficient and Pareto optimal solutions. A multi-objective evolutionary optimisation framework with preferences is proposed in this paper to solve a complex optimisation problem integrating runway scheduling and airport ground movement problem. The evolutionary search algorithm uses modified crowding distance in the replacement procedure to take into account cost of delay and fuel price. Furthermore, uncertainty inherent in prices is reflected by expressing preferences as an interval. Preference information is used to control the extent of region of interest, which has a beneficial effect on algorithm performance. As a result, the search algorithm can achieve faster convergence and potentially better solutions. A filtering procedure is further proposed to select an evenly distributed subset of Pareto optimal solutions in order to reduce its size and help the decision maker. The computational results with data from major international hub airports show the efficiency of the proposed approach.

1. Introduction

Twice as many passengers are predicted to be carried by air traffic in 2030 compared to 2013 (ICAO, 2014). With this continuous growth and no actions taken, congestion will become a serious problem for many airports together with a significant environmental impact. As a result, a lot of attention has been attracted towards research on airport operations on the surface (Atkin et al., 2010; Chen et al., 2016a,b; Marín and Codina, 2008; Lesire, 2010; Clare and Richards, 2011; Deau et al., 2009; Ravizza et al., 2013; Roling and Visser, 2008) and near airspace (Bianco et al., 2006; Samà et al., 2013).

Recently, the Active Routing (AR) approach for airport ground movement has been introduced (Chen et al., 2016a,b; Weiszer et al., 2015a; Weiszer et al., 2015b) with the aim of providing near-optimal nondominated speed profiles and routes for taxiing aircraft. AR enables the routing and scheduling of taxiing aircraft, which was previously based on distance, emphasising time efficiency, to be optimised with regard to richer information embedded within speed profiles. These include the taxiing times, the corresponding fuel consumption, and the associated economic implications, i.e. cost of taxi time and fuel (Chen et al., 2016b). Results in Ravizza et al. (2013), Chen et al. (2016b) demonstrated a significant trade-off between taxi time and fuel consumption using different speed profiles and routes, which facilitates multi-objective decision making (e.g. selecting the taxi time efficient solutions in

* Corresponding author.
E-mail addresses: m.weiszer@qmul.ac.uk (M. Weiszer), jun.chen@qmul.ac.uk (J. Chen), p.stewart1@derby.ac.uk (P. Stewart), zhxj@buaa.edu.cn (X. Zhang).

https://doi.org/10.1016/j.trc.2018.04.008
Received 26 September 2017; Received in revised form 30 March 2018; Accepted 9 April 2018
Available online 21 April 2018
0968-090X/

the peak period and fuel efficient ones in the off-peak period as in Chen et al. (2016b)). The real-time application of the AR framework can be achieved using a pre-computed database of nondominated speed profiles for key building blocks (i.e. straight taxiway segments) of the airport layout (Weiszer et al., 2015a). The database acts as a middleware to effectively separate the speed profile generation module from the routing and scheduling module. Furthermore, the airport ground movement problem has been shown in Weiszer et al. (2015b) to have an impact on another critical surface operation, runway scheduling.

Due to the multi-objective nature of speed profile generation, routing and scheduling in the AR framework implies an existence of Pareto optimal solutions for different conflicting objectives. This gives rise to the following issues: (1) which Pareto optimal solutions should be selected and stored in the database when the size of the Pareto optimal set is extensively large; (2) which routing and scheduling solution should be selected and implemented for the airport ground movement and the runway scheduling problem.

State-of-the-art approaches for the multi-objective optimisation problem considered in this study, e.g. those in Clare and Richards (2011), Frankovich and Bertsimas (2013), are proved to be computationally demanding for larger and complex instances. Therefore, a legitimate approach in this case is multi-objective evolutionary optimisation algorithms, as they are suitable for complex optimisation problems and have the ability to find multiple near-Pareto optimal solutions in a single run compared to the classical optimisation methods (Deb et al., 2002). Traditionally, multi-objective evolutionary optimisation algorithms have emphasised on the search for a complete Pareto optimal set. It is often the case that a decision maker (DM) is expected to select a preferred solution from the obtained Pareto optimal set, i.e. a posteriori, according to his/her preferences. However, the complete Pareto optimal set may be difficult to approximate and an unconverged Pareto front does not allow the DM to find an ideal solution to his/her preferences (Deb and Kumar, 2007). As a consequence, a solution with higher taxi time and fuel consumption may be chosen by the DM. On the contrary, if DM's preferences are considered before the search, i.e. a priori, the optimisation algorithm can concentrate on guiding the search to a preferred region of interest (RoI), making the search more computationally efficient with faster convergence (Branke and Deb, 2005; Karahan and Köksalan, 2010). The search by the optimisation algorithm can be also steered in an interactive manner, in which the DM progressively articulates the preferences during the search.

Scalarizing functions (Miettinen and Mäkelä, 2002) involving some additional parameters corresponding to DM's preference are often used to transform a multi-objective problem into a single-objective one. However, this approach may be counterproductive (Deb et al., 2006). The DM cannot investigate other optimal or near-optimal solutions and their properties corresponding to the preference information if only a single solution is found during the search. Moreover, in practice, the preferences are often only vague as relative weighing of the priorities is usually approximate. Therefore, the preferences are better to be utilised to search for a RoI rather than a single solution in order to take into account such uncertainty.

Research on incorporating preferences into evolutionary algorithms has been active in the last two decades. For a recent review, see (Purshouse et al., 2014). There are several ways of expressing preferences (Coello, 2000). In addition to use weights, reference points (Deb et al., 2006; Deb and Jain, 2014), aspiration levels or goal vectors to represent the desired values of objectives, the DM can also specify a utility function, preference (López-Jaimes and Coello, 2014) or outranking relation (Jaszkiewicz and Słowiński, 1999). Weights or trade-off information (i.e. how many units in one objective is at most worth a unit improvement in another objective) are often used to express preferences. Examples include an evolutionary algorithm in Branke et al. (2001) with maximally acceptable trade-off rate between objectives, a weight distribution function in Friedrich et al. (2011) and a modified Nondominated Sorting Genetic Algorithm-II (NSGA-II) (Deb et al., 2002) with reference direction (weights) (Branke and Deb, 2005; Deb and Kumar, 2007). The dominance relation is modified according to the distance to the reference point in Ben Said et al. (2010) or aspiration level satisfaction in Molina et al. (2009). An achievement scalarizing function taking into account reference point is used to prefer some solutions closer to the RoI in Thiele et al. (2009), López-Jaimes and Coello (2014), Deb and Kumar (2007).

Although the abovementioned approaches assume inherently approximate preferences and search for the RoI instead of a single solution, sometimes more information about the preference is available. For example, an interval provides more information (upper and lower bound) about the underlying uncertainty in the preference compared to the single value of the reference point, weight, etc. The uncertainty in preferences should be linked to the size of the RoI, with the RoI adjusted accordingly. Usually, a user defined parameter is introduced to control the extent of the RoI (Branke and Deb, 2005; Deb et al., 2006; Deb and Jain, 2014). From a practical point of view, setting up this parameter is not intuitive and the DM can control the extent of the RoI only approximately. As a result, improper parameter setting will lead to either a too wide RoI, wasting computational resources, or a too narrow RoI, not including preferred solutions. In more recent development, Tchebycheff weights that minimise the weighted Tchebycheff distance from the ideal point in Karahan and Köksalan (2010) and objective function values in a co-evolutionary algorithm (Wang et al., 2015) can be expressed as an interval. Also, a brushing technique (Wang et al., 2015) enables the DM to conveniently specify a range of preferred objective function values by drawing in the objective space. However, the specification of these ranges in Karahan and Köksalan (2010), Wang et al. (2015) is left completely to the DM.

In addition, a scalarizing function (i.e. weighted aggregation) and its corresponding parameters (i.e. weights) can express approximate preference. Scalarizing functions have been used in decomposition evolutionary algorithms (Zhang and Li, 2007) to convert a multi-objective problem into a set of single-objective subproblems. By varying the parameters of the scalarizing function, different solutions are obtained during the search and combined to provide a Pareto set. The DM can select multiple parameters according to his/her approximate preference, or the parameters can be set by the algorithm as in Mohammadi et al. (2012) to find solutions close to a reference point. However, the decomposition based approach has the following disadvantages: (1) if a weighted sum is used as a scalarizing function, the decomposition based algorithm cannot reach solutions on a non-convex Pareto front; (2) if a reference point is used, then it leads to a problem of controlling the extent of the RoI as described earlier; and (3) selecting evenly