JOURNAL OF AIRCRAFT
Vol. 38, No. 6, November–December 2001

Runway Operations Planning and Control: Sequencing and Scheduling

Ioannis Anagnostakis* and John-Paul Clarke†
Massachusetts Institute of Technology, Cambridge, Massachusetts 02139
and
Dietmar Böhme‡ and Uwe Völckers§
DLR, German Aerospace Research Center, 38108 Brunswick, Germany

The commercial aviation community has experienced a significant increase in the demand for air transportation over the past two decades. Despite the advent of new air traffic control (ATC) technologies, this increase in demand for air transportation has not been matched by an increase in capacity. Thus, because capacity/demand mismatches affect ATC airspace and ground operations, there has been a significant increase in delays during departure operations at major European and U.S. airports. The associated environmental impact and economic inefficiencies generate a growing need for the reduction of such delays. A framework and solution methodology developed at the Massachusetts Institute of Technology and the German Aerospace Research Establishment (DLR), to develop automated decision-aiding systems to assist air traffic controllers in handling departure traffic and mitigating the adverse effects of ground congestion and delays is presented. A possible formulation of the runway operations planning problem is presented with objectives and constraints.

I. Introduction

SIGNIFICANT delays are commonly observed in the ground departure flow at major European and U.S. airports. These delays are the result of a mismatch between the demand for air transportation and airport capacity. Whereas the demand for air transportation continues to grow at a significant rate, the capacity of most airports in the air transportation system has been growing at a slower rate, and at many airports, there has been no increase in capacity. Much of the constraints to capacity enhancement are based on environmental considerations, specifically noise and emissions. Thus, if the capacity of airports is to be increased to reduce delays, then the operations at these airports must become more efficient in terms of the number of aircraft handled per unit of resource and the environmental impact per aircraft.

Experience with the surface movement advisor program at Atlanta's airport[1] and conclusions from current NASA-sponsored research on causes of departure delay[2–5] all suggest that an automation aid to help optimize and control the departure flow would benefit both controllers and aircraft operators. An automated decision support system, which includes planning and control algorithms, is thus likely to be a significant component of the solution to this problem.[2,6–11] In fact, the primary objective of the surface management system research prototype being developed by NASA is to contribute to the understanding and solution of various problems existing within the surface domain of airports in the National Airspace System.

Airport departure management includes several control tasks, that is, pushback, engine start time, taxiway entry, runway assignment, and takeoff clearances. Air traffic controllers typically perform these tasks under conditions of high workload and time critical decision demand. With implementation of a decision-aiding system, it will be possible to explore automatically a very large number of possible future departure schedules. In addition, a decision-aiding system will hopefully contribute to the reduction of existing uncertainties by exercising tighter sequencing and scheduling control on each portion of the taxi-out process.

This paper documents the framework and methodology developed at the Massachusetts Institute of Technology (MIT) and the DLR, German Aerospace Research Establishment, to develop automated decision-aiding systems to assist air traffic controllers in handling and controlling departure traffic and mitigating the adverse effects of ground congestion and delays.

The paper is organized as follows. Section II describes the operational context for runway operations planning (ROP), discusses runway usage, and presents the basic operational requirements that any decision-support system has to meet. Section III gives a global framework for departure management and presents a high-level architecture. Section IV provides a description of runway occupancy planning (occupancy will be defined later). The planning task is formulated, and its solution as an optimization task is demonstrated. In addition, this section presents initial thoughts on solving ROP problems in a dynamic context to account for the inherent information uncertainty and incompleteness and the dynamic nature of airport operations, but no rigorous solution is pursued. In Sec. V, a short summary is given, together with topics for future work in this area.

II. Operational Context

A. ROP Within the Air Traffic Management System

In this section, a generic structural representation of the air traffic management (ATM) process is presented. Such a structure is useful in the effort to design and build decision-aiding systems for air traffic controllers because it describes the links and dependencies between the three main traffic elements of ATM: arrivals, departures, and ground traffic. Figure 1 shows a representation of the ATM process. As Fig. 1 shows, ROP is a subset of the runway operations management (ROM) process, which in turn is a subset of the overall process of ATM.

Received 19 January 2001; revision received 25 July 2001; accepted for publication 25 July 2001.

*Research Assistant, Department of Aeronautics and Astronautics, 77 Massachusetts Avenue, Room 35-217; yianag@mit.edu.

†Assistant Professor, Department of Aeronautics and Astronautics, 77 Massachusetts Avenue, Room 33-314; johnpaul@mit.edu. Senior Member AIAA.

‡Senior Researcher, Institut für Flugführung, Lilienthalplatz 7; dietmar.boehme@dlr.de.

§Director, Institut für Flugführung, Lilienthalplatz 7; uwe.voelckers@dlr.de.

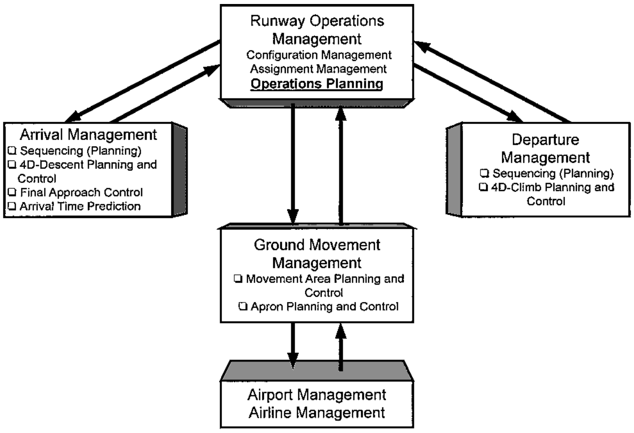

Fig. 1 Runway operations planning context.

The main tasks of ROM are as follows.

1) Runway configuration management involves the strategic planning of runway configurations to be used; operational procedures to be deployed within each configuration, such as the accelerated departure procedure at Boston Logan[12]; and configuration change times due to weather conditions, expected traffic fluctuation, and other regulations and constraints.

2) Runway allocation management determines the runway to be used by each aircraft.

3) ROP is the tactical planning of runway time allocation to arriving, departing, and crossing aircraft (Sec. II.B).

If an airport is operating under a certain prespecified configuration and the runway allocation decision for each flight has been made, the focus of the ROM process is on the latter tactical task.

In ATM operations of a single airport, ROM is the connecting link between arriving, departing, and ground traffic. Therefore, ROP, as the tactical planning component of ROM, is directly affected by arrival, departure, and ground flow constraints. At the same time, ROP produces information that can be fed back to ground, arrival, and departure management systems.

B. Runway Usage

The runway, as a resource shared by all aircraft, is a significant flow bottleneck.[12] Based on observations at Boston Logan tower,[5] the runway queue was found to be very sensitive to any factor that either limits the effective capacity of the airport or increases the demand for runway service.

During regular operations, runways are used for landings, takeoffs, and for crossings of arrivals taxiing to their gates (stands) and departures taxiing to their assigned runway. In addition, depending on the airport layout and the active runway configuration, there may be other interdependencies such as those between converging, intersecting, or closely spaced parallel runways.

There are certain fundamental properties of each of the various types of runway operations, which delineate the difference in nature between landing/takeoff operations and crossing operations. For example, depending on the commanded taxi path, not all arrivals and departures cross an active runway, and therefore, not all generate a request for a crossing occupancy interval. [The term runway occupancy interval (or, in short, occupancy) is defined as the time interval during which a single aircraft occupies a runway.] On the other hand, each arrival/departure generates a request for exactly one landing/takeoff occupancy interval. Unlike landing and takeoff operations, the nature of crossing operations offers the possibility to delay crossing clearances until a group of aircraft has accumulated to one or more crossing points and then clear aircraft from that group to cross in quick succession. Note that there is usually an upper limit on how many aircraft can be grouped at the same crossing point, due to the limited time between two runway operations. In addition, due to limited taxiway space, excessively long aircraft crossing queues can block taxiways and, consequently, the adjacent runways. In such a situation, landing and/or departure operations may have to be delayed until all crossings are cleared and runways are available again.

C. Basic Operational Requirements

In any busy airport, planning the allocation of runway time to the various types of runway operations can be a challenging task. Planning and control (execution of plans) of ground operations are the two main tactical tasks performed by the system. Although these tasks are conceptually different, one cannot overlook the interrelations and links between them. Such links are usually neither obvious nor easily observable because, in many cases, one controller mentally performs both the planning and control tasks at the same time. In addition, at an airport, planning and control involve many parties and must be performed in a distributed fashion. Consequently, a successful design of an automated ROP system that will be under airport-wide use must provide for the necessary integration between the different needs, controls, and inputs of all parties involved in ground operations. Airlines, airport authorities, airport service operators, passengers, and the air traffic controllers in the tower and the terminal radar approach control, all introduce their own constraints and inputs in the system due to the special needs and varying objectives that each of them has. In such an environment, it is difficult for an automated system to take into account all objectives and incorporate very dynamic changes of the system state (traffic situation). The design of a decision-aiding tool deployed to assist air traffic controllers with such a task must also account for operational requirements, which mainly arise due to the presence of humans operating a complex automated system and executing decisions based on the outputs of this system. Thus, the ROP plans generated must be presented to controllers as operational guidelines and not as inviolable laws, for them to have ample planning and control flexibility to modify the solutions for factors and events not visible to the ROP tool.

III. Future Departure Management Context

The two most important elements of an optimization problem are its objectives and constraints. Thus, if the ROP problem is to be formulated as an optimization problem, a set of objectives and constraints must be determined. These objectives and constraints are discussed next.

A. Objectives

Airport operations are in general governed by the operational and financial interests of airport users (airlines, passengers) and ATM service providers (airport authority, air traffic controllers). Whereas the general public will also have economic interests in the airport, surrounding communities have legitimate concerns about the environmental impact by aircraft engine emissions and noise. Specifically, communities want the environmental impact of aircraft to be reduced over time or at least maintained at current levels. Thus, the objectives and interests of the different stakeholders may not be mutually supporting. In fact, in many instances these interests may be in conflict. For example, the necessary airport throughput capacity enhancement mandated by traffic demand forecasts competes with environmental objectives. Of course, each of the different parties mentioned, attach different preference to each of the general objectives. Despite the differences, all stakeholders agree that the level of safety in air transportation must at least be maintained if not enhanced and that any new technical system or procedure developed for ATM must not in any way increase air traffic controllers' workload.

Given these principles, the main objectives of ROM have been identified as follows.

1) Enhance resource efficiency (gates, ramp, taxiways, runways) by maximizing throughput for each available runway configuration by servicing requests for runway time as quickly as possible and balancing the load between all active runways, minimizing taxi times, and minimizing pushback and clearance delays.

2) Reduce the environmental impact by controlling engine emissions during taxiing and considering noise regulations and constraints when determining airport configuration plans, for the benefits of surrounding communities.

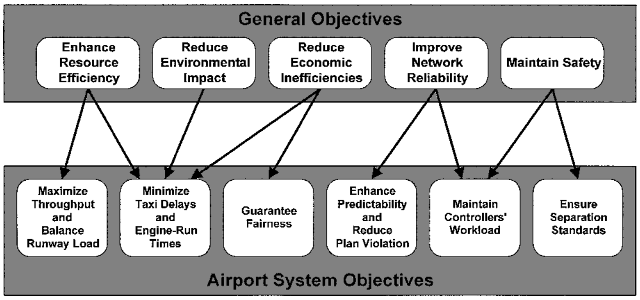

Fig. 2 Objective layers of runway operations management.

3) Reduce economic inefficiencies by minimizing engine-run times (related to taxi times) and saving the airlines from unnecessary fuel burn and the associated costs, guaranteeing fair treatment for all airport users (different airlines or flights of the same carrier), and coordinating with the arrival stream to preserve arrival priorities and minimize airborne holding (additional environmental benefits).

4) Improve reliability of the air traffic control (ATC) network that connects airports by decreasing the likelihood and the degree of violation of a runway operations plan that has been designed under certain network considerations and is now described by a set of constraints, for example, en-route time slots, and/or schedules and enhancing the predictability of operations, such as pushbacks, engine start-ups, ramp holds, and taxi and takeoff delays.

5) Maintain safety by complying with wake vortex separations or miles-in-trail restrictions and maintaining controllers' workload at acceptable levels without reducing their vigilance.

Each system objective can be mapped to an optimization criterion. Because the air traffic controllers will likely have final authority in determining the most important optimization criteria to be considered at each point in time, these criteria must be presented to controllers in a way that has intuitive meaning to them. Two common examples of intuitive criteria are the airport departure or total operations rate (throughput) and the average taxi time. Once the high-level objectives have been defined, a transformation is necessary to translate objectives to specific mathematical objective functions, which will be used to evaluate the system behavior with respect to one or more of the criteria in hand. For this very purpose, it is helpful to conceptually separate objectives into two layers: 1) general and 2) airport system objectives. Satisfying the latter is the means to fulfill the former. The relations indicated by arrows in Fig. 2 may be interpreted as a first "transformation" of objectives.

B. Constraints

The formulation of the ROP problem would not be complete without considering the most important constraints that affect runway operations. Because violation of specific constraints will have different impact on the airport and ATC system, two types of constraints can be distinguished: 1) hard constraints that are inviolable and must be satisfied by all generated solutions and 2) weak constraints that can be violated but the less the violation the better the solution quality.

Wake vortex separations between successive aircraft is an example of a hard constraint. Based on a penalty function method for handling constraints, irrespective of the mathematical formulation of the objective function, if a generated plan violates wake vortex restrictions it must drive the value of the objective function to infinity, so that departure plans that violate hard constraints will be ruled out from the solution.

A weak constraint allows more planning flexibility to the optimizer but still adds cost to the objective function when violated. An example is the scheduled takeoff times (or slots). If a flight is optimally planned to take off within a certain time slot and that slot needs to be slightly exceeded, the resulting takeoff plan is still feasible but not optimal any more. The degree of violation determines how far from optimality the system is driven.

Thus, when determining system constraints for an ROP system, it is preferable to use weak constraints whenever possible because too many hard constraints will result in a system that is overconstrained and a null solution set.

Based on field observations, the following categories of constraints were identified.

1) General operational constraints are obeyed by ATC operators to handle traffic with safety, such as the wake vortex separation restrictions.

2) Specific operational constraints are introduced by individual controller or pilot requests and may be airport-specific, such as estimated departure clearance time (EDCT) slots in ground delay programs, or standard instrument departure (SID) separations. Also, pilot/airline inputs can be considered specific operational constraints, such as when a pilot requests use of a specific runway or an airline requests priority handling for specific flights or last minute schedule adjustments to accommodate connecting passengers.

3) Physical constraints exist in the system due to the airport topology and primarily affect aircraft taxi times and runway crossings. For example, if at a given airport it takes at least 10 min for an aircraft to taxi at regular taxi speed from its gate to the assigned runway, any generated plan that requires this operation to occur in less than 10 min is unacceptable. Such constraints can be incorporated in the planning tool by having the optimizer exchange information and extract data from an accurate operational and traffic model, which possibly models the queuing behavior of different aircraft queues forming on the airport surface.

Naturally, all types of restrictions and system constraints have to be mathematically formulated to be easily included in the implementation of the ROP optimizer, that is, constraint translation into departure–departure, arrival–departure, or departure–arrival separations.

C. High-Level Architecture

The design of a high-level architecture for airport departure management should be based on a thorough analysis of the airport system and understanding of the constraints in current airport operational procedures. To this end, a significant set of field observations performed by MIT researchers[5,12] is summarized here. These observations were also the basis on which the general system architecture was developed.[2] The virtual queue manager is highlighted as the most important architectural component.

1. Field Observation Results

Field observations and data analysis at Logan and other major U.S. airports, such as Chicago O'Hare, Atlanta Hartsfield (ATL), and Dallas–Fort Worth (DFW) identified various constraints in the flow of departing aircraft. These contribute to the formation of a network of departure queues and to the low predictability associated with departures. The airport system components that were identified to contribute to the formation of surface queues are 1) the runway system, 2) the gates complex, 3) the taxiway system, and 4) the ramp area (where it exists). When the aircraft queuing process and the control process are observed, the following control functions were identified, which occur at specific control points: 1) pushback clearance (only for jets) or taxi clearance (for jets and smaller aircraft); 2) clearance to enter the taxiway system; 3) runway and taxi-path assignment; 4) sequencing of aircraft takeoffs, that is, merging of aircraft into the same takeoff queue or mixing between aircraft from multiple queues, for example, one jet aircraft and one turboprop aircraft queue, at the same takeoff runway; and 5) takeoff release of each aircraft.

It was observed that a control function could be exercised at different times and locations. Controllers have to grant specific clearances to transition an aircraft from one state to another or can give other commands to pilots at several time points or airport locations. For example, aircraft sequencing can be performed at the gate (pushback control), at the taxiway entry points as aircraft are released into the taxiway system, and up to the physical point beyond which the aircraft have to commit to a particular takeoff queue. Once the aircraft are physically present at the runway end, the takeoff sequence is hard to modify. Therefore, notionally, a control point is defined as the last opportunity that the controllers have to apply a particular

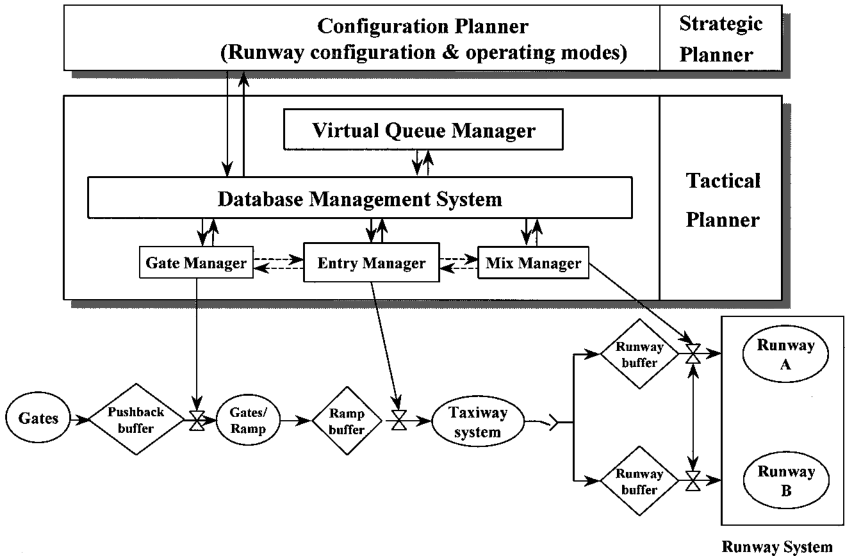

Fig. 3 Proposed architecture for runway operations management.

control function to the departure queues. It can be a physical point on the airport surface or it can be a point in time during the departure process, when the aircraft transitions from one state to another. The main control points associated with the control functions outlined earlier are 1) the gate, 2) the point of entry from the gate or ramp into the airport taxiway system, 3) the point of commitment to a specific queue (temporal or spatial), and 4) the point of entry to an active runway (exit from a takeoff queue).

2. Overview of the Proposed Architecture

Figure 3 illustrates the two principal parts that the architecture consists of, 1) a strategic planner, which is essentially a configuration planner[2] and 2) a tactical planner with an approximately 15–30 min time horizon for tactical planning of runway operations under a specific runway configuration.

The tactical core of the system is separated into four distinct components.[2] The most critical tactical component introduced in the system is the virtual queue manager. The remaining three tactical system components (from the gates, following the departure flow to the runway takeoff queues) are 1) the gate manager, 2) the taxiway entry manager, and 3) the mix manager, introduced to manage the arrival/departure mix onto active runways.

In a generic framework that can be applied to any airport, all components are envisioned to communicate and exchange data with each other directly or by way of a common data and information management system called the database management system. The latter is designed to ensure that all components have access to the same consistent information. It interacts with several specialized databases containing necessary airport specific data, such as 1) airport topology; 2) ATC procedures, regulations, and restrictions; 3) aircraft performance data, as well as dynamically generated data (flight plans and aircraft identification information); and 4) ATC constraints that are dynamically introduced, for example, flight priorities and ground delay programs.

3. Virtual Queue Manager

The virtual queue manager proactively manages the airport's virtual queue so that system objectives are met, and therefore, airport resources (runways, taxiways and gates) are efficiently utilized. A virtual queue can be defined as a notional waiting line of departing aircraft arranged, at any instant of time, according to the order in which they are expected to take off. It consists of two parts: 1) A physical part involves aircraft that are or will shortly be physically present at a certain location on the airport surface, with no further chance for resequencing; therefore, these aircraft have a fixed (frozen) position in the virtual queue. 2) A virtual part involves aircraft that are scheduled to occupy a particular position in the sequence of aircraft that will take off, but are not physically present in the takeoff queue yet. Position assignments in this virtual part of the virtual queue are very much subject to revision.

In other words, the virtual queue can be seen as an extension of the notion of a physical queue that depicts the final takeoff sequence of all scheduled departures as the runway operations planner has planned it up to the current point in time.

As shown in Fig. 3, it is hypothesized to reside in the system hierarchy at one level above the other tactical elements. It performs the necessary task of coordinating among them and interacts separately with each of them and with the strategic configuration planner. Acting as a central processing function that incorporates all of the requests from various physical queues in the system, it relays back to them information about generated runway operations' plans and the required control actions to implement those plans.

Because of the high workload, it is very hard in most cases for air traffic controllers to determine mentally the appropriate timing and sequence of departures, while at the same time keeping in mind all constraints and satisfying all system objectives. In the interest of optimizing departure operations, tower controllers may need to determine possible aircraft swaps between aircraft that are within the same stage of their taxi process, for example, both taxiing or both at the gate, or even between aircraft that are at different taxiing stages. The virtual queue may assist them in such a task as well as point out some of the optimal sequences that the controllers may not realize under heavy workload. Note that the design of a human–machine interface (HMI) between controllers and such a decision-aiding system is of critical importance. A well-designed HMI can limit the head-down time for controllers and can allow them to select the amount and type of information that will be presented to them inasmuch as the latter may affect the level of control that they will be able to exercise with the presented information in hand.

IV. Runway Occupancy Planning

Section IV.A contains a description of the formulation of the ROP problem as an optimization problem, and Sec. IV.B describes ways to incorporate several objectives and constraints in a complete problem statement. In Sec. IV.C, the use of tree search solution methods to solve Eq. (5) is discussed.

In the analysis presented, crossing operations are not included in the planning process. However, information about the planned occupancies for departures together with the predicted occupancies for arrivals can help the controllers decide when crossing operations can be performed in a way that maximizes the current airport throughput.

A. Formulation of the Planning Problem

In general, a planning problem may not need to be solved by optimization methods. It is however, advantageous to formulate it as an optimization problem, because even if an optimal solution cannot be found for any reason, such a formulation can contribute to a better understanding of the problem structure.

At first, the planning task is considered as an optimization task based only on the departure sequence and the following sets are defined: D is the set of indices of the n_D departures for which the optimum sequence s^* will be found, A is the set of indices of the n_A arrivals expected to interfere with the departure stream, and C is the set of all known hard-constraints.

Equation (1)

$$s^* = \arg \min_{s \in S(D,A,C)} Q(s) \tag{1}$$

describes the optimization problem, where $Q(s)$ is a function used to quantify the quality of the sequence s, and $S(D, A, C)$ is the set of all feasible sequences of departures merged with arrivals. When it is assumed that the arrival sequence cannot be changed during planning, an upper bound for the size (cardinality) of set S is given by

$$n_S = card(S) \leq (n_A + n_D)!/n_A! \tag{2}$$

For departure and arrival traffic at the level that is usually observed at large busy airports, n_D and n_A can assume high values. In such a case, Eq. (2) shows that, even if only the sequence of departures is controlled, the number of possible sequences still becomes very large. In reality, because aircraft can only be resequenced within a small time window, the number of possible sequences can be further reduced. Two other challenges exist. First, a method to construct a measurement function $Q(s)$, which would measure the suitability of every possible sequence with respect to departure planning objectives, is not currently known. Second, in calculating $S(A, D, C)$, it is very hard to distinguish between feasible and infeasible sequences because the set S contains sequences with merged arrivals and departures. Based solely on sequences and without the presence of times, it is not clear whether it will be possible for an arrival to touch down or a departure to take off as described in a certain sequence of S. When it is known that many of the hard constraints are either directly based on time, for example, EDCT, or can be expressed in terms of time, for example, miles in trail, and that time is also included in the objectives either explicitly, for example, taxi time minimization, or implicitly, for example, throughput maximization, a time-based approach must be considered.

A general time-based formulation of the planning problem is now presented. For the following analysis it is sufficient to map every takeoff operation to exactly one time. These times could be the takeoff (wheels-off) times or the starting times of the occupancy intervals, as those were defined in Sec. II.B, assuming that the (average) size of these occupancy intervals is known. For simplicity, the term takeoff times will be used regardless of their exact definition. Given a vector $\boldsymbol{t}$, which contains all takeoff times of the departures in the set D, the time-based planning task is then described as

$$\boldsymbol{t}^* = \arg \min_{\boldsymbol{t} \in T(C)} Q(\boldsymbol{t}) \tag{3}$$

where $\boldsymbol{t}^*$ is the optimizing time vector, $Q(\boldsymbol{t})$ is the time-based evaluation function that reflects the planning objectives, and $T(C)$ is an n_D-dimensional search space restricted by the set C of hard constraints. Because the constraint set C contains the minimum separation requirements between takeoff occupancies as well as information on expected arrival occupancies, the search space $T(C)$ is, in general, not only a nonconvex set, but also possibly a disjoint space (Fig. 4).

In Fig. 4 the minimum separation between both takeoff times is 120 s ($c_1 : |t_1 - t_2| \leq 120$), the range of (predicted) runway occupancy by an arrival is from 120 to 165 s ($c_2 : \boldsymbol{t}_{A1} = \langle 120 \quad 165 \rangle$) and departure 1 cannot reach the runway earlier than 100 s for certain reasons ($c_3 : t_1 \geq 100$). The vector $\boldsymbol{t} = [t_1 \quad t_2]$ includes the starting times of the departure occupancies.

Besides the difficulties caused by the properties of $T(C)$, it is very likely that the function $Q(\boldsymbol{t})$ is a multimodal one with several local minimums. Consequently, the optimization problem is usually too complex to be solved within reasonable computation time.

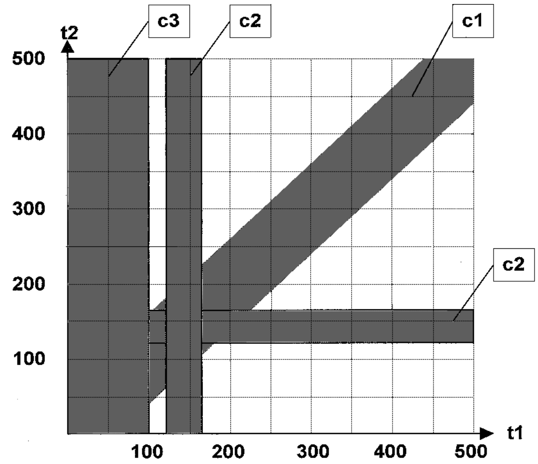

Fig. 4 An example of a two-dimensional, non-convex, disjoint search space T(C) (white area).

Because the sequence-based approach is insufficient and the time-based approach is too complex, a hybrid formulation is introduced here, which, under certain conditions, can provide a basis for solving the problem. Hence, for any sequence s, a time-based optimization task is considered where the solution space $T[C, U(s)]$ is not only defined by the hard constraints C, but also by a set of inequalities each of which governs a pair of takeoff times, for example,

$$U(s) = \{t_i \leq t_j \leftarrow i \underset{s}{\prec} j \; \forall i, j \in s\} \tag{4}$$

Therefore, based on the evaluation function $Q(\boldsymbol{t})$, the optimal takeoff time vector $\boldsymbol{t}^*$ for a specific sequence s is calculated as

$$\boldsymbol{t}^* = \boldsymbol{t}^*(s) = \arg \min_{\boldsymbol{t} \in T[C,U(s)]} Q(\boldsymbol{t}) \tag{5}$$

and the optimal sequence s^* results from minimizing the values $Q(\boldsymbol{t}^*)$ among all feasible sequences in the set $S(D, C)$:

$$s^* = \arg \min_{s \in S(D,C)} Q[\boldsymbol{t}^*(s)] \tag{6}$$

Therefore, the equation

$$\boldsymbol{t}^{**} = \boldsymbol{t}^*(s^*) = \arg \min_{s \in S(D,C)} \min_{\boldsymbol{t} \in T[C,U(s)]} Q(\boldsymbol{t}) \tag{7}$$

yields the global optimum vector of takeoff times. The first minimization in Eq. (7) can be done by a discrete search over a number of sequences not exceeding $n!$. However, it is not certain that each necessary minimization of the planning function [Eq. (5)] can be solved easier than the original problem (3).

B. Objective Functions

1. *Objective Functions and System Objectives*

In this section, some methods are outlined for transforming system objectives to objective functions, and the objective functions, $q_i(t)$ are introduced. Each of these functions serves as an evaluation means either for specific system objective or for the degree of violation of weak constraints (Sec. IV.B.2). Furthermore, for reasons explained in Sec. IV.C.1, each objective function must have the properties

$$q_i(\boldsymbol{t}) \geq 0, \qquad q_i(\boldsymbol{t}) = \sum_{j=1}^{n} q_{i,j}(t_j) \tag{8}$$

Additionally, each q must be defined in such a way that, the lower the value of $q_i(\boldsymbol{t})$ is, the more the solution satisfies a certain system objective or the less a certain weak-constraint is violated.

The specific definitions of the objective functions are highly dependent on the intuition of the system developer. To illustrate this, a few objective functions are presented, each of which is linked to one of the system objectives of throughput, taxi times, fairness, and workload. The objective functions defined subsequently should be treated purely as examples that could be modified or replaced by other definitions.

Throughput. Because departure throughput can be defined as the quotient of the number of takeoffs n and the time necessary to perform these takeoff operations, it is related to the takeoff time of the last aircraft in the optimal sequence s^*. Hence, the definition $q(\boldsymbol{t}) = \max(\boldsymbol{t})$ naturally follows. However, in that case, only the last takeoff time is considered, without the values of the other elements of the takeoff time vector $\boldsymbol{t}$. Therefore, the function

$$q(\boldsymbol{t}) = \sum_{j=1}^{n} \tau_j^p, \qquad p > 1 \tag{9}$$

where

$$\tau_j = t_j - t_0$$

is the relative time and t_0 is the present time, seems to be more suitable.

Taxi Times. If no advance taxi planning is performed, minimizing taxi-out times is equivalent to minimizing taxi delays. An objective function is now defined on the basis of delays that occur at the end of the taxiing process:

$$q(\boldsymbol{t}) = \sum_{i=1}^{n} \delta_{T,j}(t_j)^p, \qquad p > 1 \tag{10}$$

where

$$\delta_{T,j}(t_j) = t_j - t_{\mathrm{RTO},j}$$

is the taxi delay for aircraft j and $t_{\mathrm{RTO},j}$ is either the time the departure became ready to takeoff (RTO), if $t_{\mathrm{RTO},j} \leq t_0$, or the time the departure is expected to become RTO if aircraft j has pushed back from its gate ($t_{\mathrm{PBC},j}$) but is still taxiing, that is, $t_{\mathrm{RTO},j} > t_0$ and $t_{\mathrm{PBC},j} < t_0$, or the time the departure aircraft j is planned to become RTO, if $t_{\mathrm{RTO},j} > t_0$ and $t_{\mathrm{PBC},j} \geq t_0$, that is, $\delta_{T,j}(t_j) = 0$.

Fairness. Fairness requires that unequal treatment of different airport users should be avoided as much as possible. For example, unfair treatment occurs when a single departure takes an amount of delay significantly higher than other flights operated at or around the same time. The difference (delay) between the time $t_{\mathrm{RPB},j}$ the departure j calls ready for pushback (RPB) and the time $t_{\mathrm{PPB},j}$ it is planned to pushback (PPB) from the gate is of special interest because this delay does not affect the taxi-out times. This motivates the use of the following objective function with a progressive evaluation of such delays:

$$q(\boldsymbol{t}) = \sum_{i=1}^{n} \delta_{\mathrm{PB},j}(t_j)^p, \qquad p > 0 \tag{11}$$

where

$$\delta_{\mathrm{PB},j}(t_j) = \begin{cases} 0, & t_{\mathrm{PPB},j} < t_0 \\ t_{\mathrm{PPB},j}(t_j) - t_{\mathrm{RPB},j}, & \text{otherwise} \end{cases}$$

is the pushback delay for departure j. Note that, if the pushback clearance has already been granted, that is, $t_{\mathrm{PPB},j} < t_0$, then the pushback delay is not a function of the planned takeoff time.

Workload. A possible indicator for controller workload is the number of departures N that have to be controlled by all controllers or n_k, the number controlled by a specific controller position. These numbers are functions of time and depend on the schedule of granted and planned clearances and on the movement of aircraft. Future clearances are planned based on takeoff times generated by the system. Note that the controllers are not expected to be able to implement departure plans exactly on the planned time (Sec. II.C). However, because the future behavior of the controllers is unknown, all measurement functions and ROP consider the future in general as what would happen if controllers strictly adhered on the times planned by the ROP system.

Controllers' workload is also affected by the dynamics of ROP because every change of a plan (which may occur quite frequently) requires mental and/or physical effort. To account for this, an objective function can be introduced that quantifies workload on the basis of the dissimilarity $d(\boldsymbol{t}_{-1}^{**}, \boldsymbol{t})$ between two consecutive plans, where $\boldsymbol{t}_{-1}^{**}$ is the previous optimal plan. In the very dynamic environment that an ROP system operates, the two successive iterations that generate the consecutive plans $\boldsymbol{t}_{-1}^{**}$ and $\boldsymbol{t}$ are likely to occur within a few seconds of each other.

2. *Objective Functions and Weak Constraints*

Weak constraints are often expressed in terms of time. They usually translate downstream constraints, such as departure slots generated by Eurocontrol's Central Flow Management Unit (CFMU)[¶] in Europe or by special flow control programs such as departure spacing programs and ground delay programs in the United States into time-based constraints on aircraft in the departure queue. Such constraints may also be imposed by the predetermined daily departure schedules of airlines.

Mathematical functions can be used to quantify the degree of constraint violation. However, when several constraints have to be taken into account, the definition of the corresponding function must be adjusted for the acceptable degree of violation. For example, a departure can be delayed 30 min with respect to its scheduled departure time, and that may be acceptable given the current traffic situation at the originating airport. However, this amount of delay may not be acceptable when the reference time is the end of a CFMU slot. Even a slight slot violation may need approval, whereas larger delays require a new slot negotiation altogether. When this is considered, an evaluation function for the constraint violations can be

$$q(\boldsymbol{t}) = \sum_{k=1}^{n} c(\tau_j)\delta_j(t_j, t_{c,j})^p, \qquad p > 0 \tag{12}$$

where $t_{c,j}$ is the reference time (equal to $+\infty$ when there is no such constraint for departure j), $p > 0$ is a parameter with an appropriate value, and

$$\delta_j(t_j, t_{c,j}) = \begin{cases} 0, & t_j \leq t_{c,j} \\ t_j - t_{c,j}, & t_j > t_{c,j} \end{cases} \tag{13}$$

is the magnitude of the constraint violation.

3. *Objective Functions and Planning Criteria*

In the preceding examples, objective functions served as evaluators for system objectives as well as for constraint violations. The planning task naturally transforms into a vector optimization problem because various functions are minimized, each of which corresponds to different system objectives or constraints. In addition, some of the system objectives and the corresponding objective functions contradict each other, which means that the minimization of only one function would increase the value of other functions. For example, considering fairness as the only important objective and minimizing the function of Eq. (11), would enforce a first-come–first-serve strategy, which, in general, is neither optimal with respect to the throughput nor minimizes the taxi-out times. Therefore, some sort of tradeoff between different objective functions is required.

Although several objectives were stated (Secs. III.A and IV. B.1), the problem can be solved as the minimization of a single objective, which is a liner combination of all objectives. Therefore, a planning function $Q(\boldsymbol{t})$ is introduced as a linear combination of individual objective functions $q_i(\boldsymbol{t})$, that is,

$$Q(\boldsymbol{t}) = \boldsymbol{a}^T \boldsymbol{q}(\boldsymbol{t}) \tag{14}$$

where

$$\boldsymbol{q}(\boldsymbol{t}) = [q_1(\boldsymbol{t}) \cdots q_r(\boldsymbol{t})]^T \tag{15}$$

[¶]The CFMU website is URL: http://www.cfmu.eurocontrol.be/doc/cfmu_doc/index.htm.

is a vector of r selected objective functions and

$$\boldsymbol{a}^T = [a_1 \cdots a_r] \tag{16}$$

is a corresponding vector of weight factors for which the following equation holds:

$$a_i \geq 0, \ \forall i \in N_r = \{1, 2, \ldots, r\}, \qquad \sum_{i=1}^{r} a = 1 \tag{17}$$

The optimal solution now becomes a function of the weight vector $\boldsymbol{a}$,

$$\boldsymbol{t}^*(\boldsymbol{a}) = \arg \min_{t \in T(C)} Q(\boldsymbol{a}, \boldsymbol{t}) \tag{18}$$

$$s^*(\boldsymbol{a}) = \arg \min_{s \in S(D,C)} Q[\boldsymbol{a}, \boldsymbol{t}^*(s)] \tag{19}$$

This method not only allows the incorporation of different optimization aspects in the same function, but also offers flexibility and adaptability to ROP.

C. Solution of the Planning Problem

1. *Sequential Determination of Takeoff Times*

The examples of objectives presented in the preceding section reveal that the vast majority of optimization aspects (system objectives or weak constraints) share a common property: As takeoff times t_j increase, the system is driven away from satisfying objectives or complying with constraints. This property matches the general expectation that successful runway operations management should allocate resources in such a way that takeoffs are handled as quickly as possible. Therefore, the corresponding objective functions should be monotonically increasing, regardless of the specific definition of each of them.

If only such monotonic objective functions are considered, then $Q(\boldsymbol{t}) = \boldsymbol{a}^T \boldsymbol{q}(\boldsymbol{t})$ will be monotonically increasing, too,

$$Q(\boldsymbol{a}, \boldsymbol{t}_k) \leq Q(\boldsymbol{a}, \boldsymbol{t}_l) \qquad \text{if } \boldsymbol{t}_k < \boldsymbol{t}_l \tag{20}$$

which provides the key to solve the yet open optimization problem of the ROP task. As already shown in Sec. IV.B, the optimization task described by Eq. (7) consists of two subtasks: 1) determination of optimum takeoff times for a given sequence according to Eq. (5) and 2) determination of the best sequence on the basis of the corresponding optimal takeoff times (see Sec. IV.C.2).

When it is known that $Q(\boldsymbol{t})$ satisfies Eq. (20), it is possible to determine sequentially the takeoff times as the earliest feasible release times subject to the set of inequalities $U(s)$ in Eq. (4). Sequential determination of takeoff times transforms the original n-dimensional minimization task [Eq. (3)] into n one-dimensional optimization tasks, which are much easier to solve. For simplicity, and without loss of generality, it is assumed that the vector $\boldsymbol{t}$ of takeoff times can be arranged in the order of a certain sequence s, so that t_j is the takeoff time of the jth departure in s. (Note that for the remainder, the sequence index is omitted when the symbol refers to only one sequence.) Whether departure j can start its takeoff roll at time $t_j \geq t_0$, that is, t_j is a feasible time or not, depends on hard constraints imposed by past or present (time t_0) landings and takeoffs; future (predicted) landings; the $j-1$ takeoffs that are currently planned to occur in the future before t_j, that is,

$$t_i^*(s) \leq t_j, \qquad \forall i < j$$

and the expected (modeled) time, necessary for departure j to reach RTO status.

At this point it is necessary to develop a model that calculates the earliest feasible departure time for the next departure subject to the hard constraints and also propagates the impact of hard constraints on each flight as time advances and the aircraft proceed from their gates to the takeoff point. Time-based operational constraints are usually defined between successive departures (A, B), based on well-known ordered attributes of a feature set, such as weight class (attributes are heavy, medium, and light), engine types (single propeller, twin propeller, and all other), assigned runway (number 1, number 2, etc.), assigned SID (route 1, route 2, etc.), flight destinations (DFW, ATL, etc.), etc. Therefore, these constraints can be expressed in terms of constraint matrices of the form

$$C_c = \begin{bmatrix} c_{c,1,1} & L & c_{c,1,r_c} \\ M & O & M \\ c_{c,r_c,1} & L & c_{c,r_c,r_c} \end{bmatrix} \tag{21}$$

where element $c_{c,k,l}$ is the required minimum time separation when aircraft A has attribute k and aircraft B has attribute l of feature c. A common example of such a matrix is the wake vortex separation matrix, which is usually presented in terms of mile separations, but can also be transformed to a matrix containing minimum separation times.

Let us introduce a corresponding state vector for each constraint matrix,

$$\boldsymbol{x}_c(t') = [x_{c,1}(t') \cdots x_{c,r_c}(t')] \tag{22}$$

This vector contains earliest takeoff times corresponding to the values of attribute c. The takeoff time t_j for departure j is constrained by

$$t_j \geq x_{c,k}(t') = \check{x}_c(t') \tag{23}$$

when it has value k of attribute c. If there are m different hard constraints for the takeoff time, the most restrictive one is considered, and, therefore,

$$t_j \geq \max\{\check{x}_1(t'), \ldots, \check{x}_c(t'), \ldots, \check{x}_m(t')\} = \hat{x}(t') \tag{24}$$

The takeoff time $t_j = \hat{x}(t')$ is suitable only if the occupancy interval $[\hat{x}(t'), \hat{x}(t') + \Delta_j]$ does not overlap with the predicted runway occupancy interval of any of those arrivals expected to use a runway interdependent with the takeoff runway of departure j. If t_j is not suitable, the minimum conflict-free takeoff time $t_j > \hat{x}(t')$ has to be determined within the next arrival gap that is large enough to fit this takeoff operation within it. When the earliest conflict-free takeoff time t_j is calculated, the new state vectors $\boldsymbol{x}_c(t_j)$ can be determined by

$$\boldsymbol{x}_c(t_j) = \max \begin{bmatrix} \boldsymbol{x}_c(t') \\ t_j + \boldsymbol{c}_{c,k} \end{bmatrix} \tag{25}$$

where $\boldsymbol{c}_{c,l}$ is the kth row vector of $\boldsymbol{C}_c$ and departure j has value k of attribute c. In Eq. (25), the max-operator is performed along the column vectors. The scalar vector addition denotes an addition of the scalar to every element of the vector.

2. *Determination of the Optimum Sequence*

For the selection of the best sequence

$$s^*(\boldsymbol{a}) = \arg \min_{s \in S(D,C)} Q[\boldsymbol{a}, \boldsymbol{t}^*(s)] = \arg \min_{s \in S(D,C)} Q^*(\boldsymbol{a}, s) \tag{26}$$

a tree-search method can be used in which only a subset of all feasible sequences has to be examined.

In the search tree in Fig. 5, every node except the root carries information about a possible takeoff operation of exactly one aircraft. The extension beyond a certain search node $[k-1]$ means that, for all remaining departures, that is, for all departures that do not belong to the sequence $s_{[k-1]}$ (which corresponds to the path from the root to node $[k-1]$, the possibility of being the next takeoff is investigated. The node $[k-1]$ that is currently expanded is called father and the nodes resulting from this expansion are called sons. A node, which has not yet been expanded, is called a leaf. Whenever a son $[k]$ is created, it is examined whether the new sequence $s_{[k]}$ violates any sequence constraint or not. If there are no hard constraint violations, for each corresponding aircraft j, the minimum takeoff time

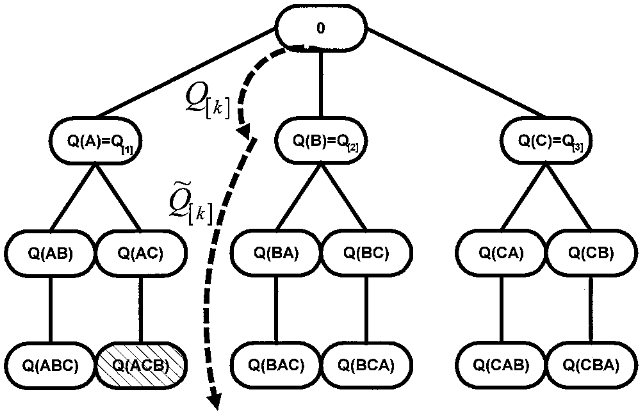

Fig. 5 Illustration of the search tree.

$t_j = t_{[k]}$ has to be determined with the help of the state vector set $\boldsymbol{X}_{[k-1]}$ of the father node $[k-1]$

$$\boldsymbol{X}_{[k-1]} = X\left(t_{[k-1]}\right) = \left\{x_1\left(t_{[k-1]}\right), \ldots, x_c\left(t_{[k-1]}\right), \ldots, x_m\left(t_{[k-1]}\right)\right\} \tag{27}$$

and the contribution $Q_j^*(a)$ to the planning criteria has to be calculated [Eq. (29)].

Because every objective function is a sum over all aircraft of the value that each of them contributes to the specific evaluation [Eq. (8)], the planning criteria can be formulated as

$$Q(\boldsymbol{a}, \boldsymbol{t}) = \boldsymbol{a}^T \sum_{j=1}^{n} \boldsymbol{q}(t_j) = \sum_{j=1}^{n} Q_j(\boldsymbol{a}, t_j) \tag{28}$$

To emphasize that the summation is performed for a certain sequence s and the corresponding optimum departure times $\boldsymbol{t}^*(s)$, Eq. (28) is written

$$Q[\boldsymbol{a}, \boldsymbol{t}^*(s)] = \sum_{s \Rightarrow j} Q\left[\boldsymbol{a}, t_j^*(s)\right] = \sum_{s \Rightarrow j} Q_j^*(\boldsymbol{a}) = Q^*(s, \boldsymbol{a}) \tag{29}$$

where

$$\sum_{s \Rightarrow j}$$

denotes a summation over all aircraft in the order they are described by sequence s. Let now $s_i = (s_{i1}, s_{i2}, \ldots, s_{in}) \in S(D, A, C)$ be a feasible sequence and $s_{ir} = (s_{i1}, s_{i2}, \ldots, s_{ir})$ $r < n$ a subset (subsequence) of s_i, such that

$$Q^*(s_{iu}, \boldsymbol{a}) \le Q^*(s_{ir}, \boldsymbol{a}) \le Q^*(s_i, \boldsymbol{a}) \qquad \forall i, \forall u < r \tag{30}$$

holds because Eq. (8) ensures that $Q(\boldsymbol{a}, t_j) \ge 0$. For example, if $D = \{a, b, c\}$, and also $s_i = (b, c, a)$, then, it is $Q[(b)] \le Q[(b, c)] \le Q[(b, c, a)]$, and obviously when Eq. (30) is applied to the search tree (Fig. 5) along any tree path, the values of $Q^*(\cdot)$ corresponding to the tree nodes traversed are always increasing.

This property facilitates the possible application of a branch and bound algorithm. This is because, during the search process, the tree can be pruned below a search node $[k]$, if for the corresponding value $Q_{[k]}^*$ Eq. (31) holds:

$$Q_{[k]}^* \ge Q^*(\tilde{s}^*) \tag{31}$$

where $Q^*(\tilde{s}^*)$ is the minimum value of the planning criteria for all sequences that have already been investigated and, therefore, $\tilde{s}^*$ is the best sequence that has been found so far. On the other hand, a possible application of an A^* algorithm requires an estimator function $\tilde{Q}(s_{ir})$, which estimates the remaining costs for any possible subsequence s_{ir}. This function has the properties

$$0 \le \tilde{Q}(s_{ir}) \le \min\left\{Q^*(\bar{s}_{ir,1}), Q^*(\bar{s}_{ir,2}), \ldots\right\}$$
$$\forall s_i \in S(D, A, C), \qquad \forall r < n, \qquad \forall \bar{s}_{ir,j} \in \bar{S}_{ir}(s_{ir}, D, A, C) \tag{32}$$

where $\bar{S}_{ir}(s_{ir}, D, A, C) \subset S(D, A, C)$ is the set of all sequences with identical subsequence s_{ir}. With such a function $\tilde{Q}(s_{ir})$ in hand, the tree can be pruned below a search node $[k]$, if for the corresponding value $Q_{[k]}^*$ holds

$$Q_{[k]}^* + \tilde{Q}(s_{ir}) \ge Q^*(\tilde{s}^*) \tag{33}$$

when s_{ir} denotes the sequence from the root node-to-node k.

When the planned runway occupancies have been calculated, pushback schedules can be produced by using statistical transfer models, which can capture the effect of factors that influence taxi times (pushback to takeoff), for example, traffic density, visibility conditions, runway configurations, etc. In general it is better to backcalculate several pushback schedules from one takeoff schedule each accounting for a different terminal area and for different ground traffic conditions.

V. Summary

This paper described the research efforts in the field of optimal planning of airport ground operations, with a particular focus on runway operations and even more specifically departures. In earlier work, field observations conducted at Boston Logan airport helped in identifying the different system parameters involved in the ROP problem and a conceptual architecture was devised[2] to describe the interactions and interdependence between these parameters and to provide a design basis for an ROP optimizer/decision-aiding tool for air traffic controllers. Here, a system structure was presented, which describes how ROP falls within the context of ROM and ATM in general. In addition, a possible formulation of the problem objectives and constraints was given for the static version of the ROP problem.

In the static version of the planning problem, planning is performed using all relevant information available at a certain point in time. However, ROP is actually a dynamic problem. In a dynamic airport environment, where humans are involved in operations planning, decision making, and control, there is usually significant uncertainty associated with any prediction and even with any observation of the current system state. Information relevant to ROP can change very rapidly. Information required by controllers, planners, and airline operators are not always available at the desired time or not available at all due to the lack of sufficient surveillance and information management systems, for example, information about the aircraft state before the pilot's call-ready for pushback. The randomness caused by system uncertainty and information incompleteness and the controllers' limited controllability on certain system variables call for dynamic planning to improve system performance. In the high-workload, time-stressed, decision-demanding environment of an airport control tower, an ROP system must be designed to perform event-driven optimization based on specific types of events that could trigger reoptimization (new solution of the static problem), such as pilot or controller inputs, for example, new call-ready for pushback or new ATC constraints. Such triggering events can occur frequently enough to cause stability problems between successive solutions. For that reason, the system must incorporate a clear definition of solution stability and the controllers, as users of the system, have to be trained to trade off between solution optimality and stability if necessary, by adjusting planning parameters, such as objective function weight factors.

Runway crossings must be considered as additional system constraints and be involved in the planning of runway operations because they also occupy runway time. Furthermore, thus far, only simple taxi time estimates are used, which were created after consulting experienced air traffic controllers at specific airports. A generic operational traffic/queuing model is necessary to represent the movement of departing aircraft from the gate to their assigned takeoff runways.

Such a model must take into account real-time airport congestion levels, as well as particular regular or irregular events that may affect taxi operations, such as taxiway blocking and unavailability or varying taxi paths (used arbitrarily by controllers) from the same gate to the same runway.

When the planning quality of the system is improved by designing a decision-aiding tool that allocates runway time for all types of ground operations (departures, arrivals, and crossings) among one or more runways, the next step is the enhancement of the system's control capabilities and the utilization of all flow control options. For example, apart from regulating takeoff releases and gate push-back clearances (gate holds), the system can offer the controllers additional options in managing the departure flow by controlling engine-start times; delaying taxiway entry clearances, even if the aircraft has been pushed back from its gate; modifying the aircraft's taxi path; and controlling of landing times and stretching gaps between aircraft in the arrival flow to absorb miles in trail restrictions imposed on departures.

Acknowledgment

This research was supported in part by NASA, under Grant NAG 2-1128.

References

[1]Lawson, D. R., "Surface Movement Advisor (SMA)," *Journal of Air Traffic Control*, Jan.–March 1998, pp. 25–27.

[2]Anagnostakis, I., Idris, H. R., Clarke, J.-P., Feron, E., Hansman, R. J., Odoni, A., and Hall, W. D., "A Conceptual Design of a Departure Planner Decision Aid," *Proceedings of the 3rd U.S.A./Europe Air Traffic Management Research and Development Seminar, Proceedings, Session E: Airport Ground Operations, Arrival and Departure Management*, Naples, Italy, 13–16 June 2000, FAA and Eurocontrol; also available at http://atm-seminar-2000.eurocontrol.fv [cited 6 Oct. 2000].

[3]Andersson, K., Carr, F., Feron, E., and Hall, W., "Analysis and Modeling of Ground Operations at Hub Airports," 3rd U.S.A./Europe Air Traffic Management Research and Development Seminar, June 2000.

[4]Pujet, N., and Feron, E., "Input–Output Modeling and Control of the Departure Process of Congested Airports," *Air Traffic Control Quarterly Journal*, Vol. 8, No. 1, 2000, pp. 1–32.

[5]Idris, H., Anagnostakis, I., Delcaire, B., Hansman, R. J., Clarke, J.-P., Feron, E., and Odoni, A., "Observations of Departure Processes at Logan Airport to Support the Development of Departure Planning Tools," *Air Traffic Control Quarterly Journal*, Special Issue in Air Traffic Management, Vol. 7, No. 4, 1999, pp. 229–257.

[6]Böhme, D., "Improved Airport Surface Traffic Management by Planning, Problems, Concepts and a Solution—TARMAC," *Advanced Technologies for Air Traffic Flow Management*, edited by H. Winter, Lecture Notes in Control and Information Sciences, Springer-Verlag, New York, 1994, pp. 191–224.

[7]Böhme, D., "Airport Capacity Enhancement by Planning of Optimal Runway Occupancies," *Proceeding of the 39. Internationales Wissenschaftliches Kolloquium der TU Ilmenau*, Vol. 3, ISSN 0943-7207, 24–30 Sept. 1994, pp. 250–256.

[8]Delcaire, B., and Feron, E., "Dealing with Airport Congestion: Development of Tactical Tools for the Departure Flows from a Large Airport," M.S. Thesis, Technology and Policy Program, Massachusetts Inst. of Technology, Cambridge, MA, June 1998.

[9]Feron, E., Hansman, R. J., Odoni, A., Cots, R. B., Delcaire, B., Feng, X., Hall, W. D., Idris, H., Muharremoglou, A., and Pujet, N., "The Departure Planner: A Conceptual Discussion," Massachusetts Inst. of Technology, available at URL: http://snipe.mit.edu/dp/documents.html [cited 17 April 1998].

[10]Völckers, U., and Böhme, D., "Dynamic Control of Ground Movements: State-of-the-Art Review and Perspectives," AGARD-R-825, 1997.

[11]Völckers, U., Brokof, U., Dippe, D., and Schubert, M., "Contributions of DLR to Air Traffic Capacity Enhancement within a Terminal Area," AGARD-CP-358, Machine Intelligence in Air Traffic Management Meeting, May 1993.

[12]Idris, H., Delcaire, B., Anagnostakis, I., Hall, W. D., Pujet, N., Feron, E., Hansman, R. J., Clarke, J-P., and Odoni, A., "Identification of Flow Constraint and Control Points in Departure Operations at Airport Systems," *Proceedings of the AIAA Guidance, Navigation, and Control Conference*, Vol. 2, AIAA, Reston, VA, Aug. 1998, pp. 947–956.