A Network Congestion Control Approach to Airport Departure Management

Harshad Khadilkar and Hamsa Balakrishnan

***Abstract*— This paper presents a novel approach to managing the aircraft taxi-out process at airports, by posing the problem in a network congestion control framework. We develop a network model for a generic airport and then validate it using surface surveillance data from Boston Logan International Airport. A set of stochastic processes that constitute the link travel times are proposed, followed by a discussion of the theoretical maximum network throughput. Finally, we propose a control algorithm that balances network congestion with performance, while maintaining stability. We show through simulation that the algorithm is capable of regulating total traffic in the network to a desired level.**

I. Introduction

A. Motivation

The reduction of taxi-out times at airports has the potential to substantially reduce flight delays and fuel consumption on the airport surface, and to improve the air quality in surrounding communities. The taxiway and runway systems at airports determine its maximum possible *departure throughput*, or rate of aircraft departures. Current air traffic control policy allows aircraft to push from their gates as soon as they are ready, and enter the taxiway system. As this *pushback rate* approaches the maximum throughput, large queues are formed at the departure runway, in addition to conflicts happening at other points on the surface. While there have been several studies that aim to relieve surface congestion [1], [2], [3], [4], [5], [6], they typically either fail to consider the inherent stochasticity of aircraft movement and pilot behavior, or are intrinsically numerical in nature. This work presents a formulation that explicitly accounts for stochastic taxi-out behavior, while providing insight into the system via an analytical treatment of airport performance.

B. Literature Review

Improving the efficiency of the departure process at airports has been a problem of interest for some considerable time [1]. Most traditional approaches to this problem involve the use of optimal scheduling algorithms [2], [3] or queuing theory [4], [5], [6]. Optimization methods typically assume that aircraft move at constant velocities at all times, and follow time-based taxi instructions exactly. This is not particularly realistic, considering the current state of technology at airports. Queuing theory approaches involve either the use of simplifying assumptions to obtain analytical results, or else require numerical solution. On the other hand, posing the problem in a network congestion control framework has the advantage of being able to accommodate complex link travel time distributions. At the same time, it allows us to address the issue of network stability and performance through analytical approaches. In addition, new runway safety systems at many major airports provide us with the capability to support such models with empirical evidence.

The system considered here is similar to those encountered in wireless networks [7], [8], TCP congestion control [9], [10] as well as the design of manufacturing systems [11], [12], [13]. One key difference is that, while the implementation of TCP requires decentralized control, centralized control is possible for congestion management at airports. In addition, the information available is global in nature. Similar network models have been previously proposed for urban transportation problems, and there exists literature that deals with solving optimal route problems in this context [14], [15]. However, these models are primarily suited to networks with a large number of sources, sinks and links, while the airport network contains only a few of each. On the other hand, the airport model requires more detailed models of link travel times than those found in urban transportation studies. Also, the emphasis is more on optimal time of entry into the network than on optimal routing. In this context, there have been aggregate rate-control approaches that aim to stabilize surface traffic at a specific level [16]. These algorithms are easy to implement in practice, but do not realize the full potential gains that may be available from congestion management strategies, in terms of fuel savings and airport performance. For example, the sampling and control intervals for such algorithms tend to be large, of the order of 15 minutes. The solution approach proposed in this paper will address some of the issues highlighted here. The objective is to suggest a time of *pushback* from the gate for each aircraft preparing to depart, such that congestion on the airport surface is minimized while limiting the adverse effect on throughput.

C. Overview of Surface Surveillance Data Source

Airport Surface Detection Equipment, Model-X (ASDE-X) is primarily a safety tool designed to mitigate the risk of runway collisions [17]. It incorporates real-time tracking of aircraft on the surface to detect potential conflicts. There is potential, however, to use the data generated by it for surface operations analysis and modeling of aircraft behavior. Reported parameters in ASDE-X include each aircraft's

Harshad Khadilkar is a doctoral candidate in the Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA 02139, USA harshadk@mit.edu

Hamsa Balakrishnan is an Assistant Professor in the Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA 02139, USA hamsa@mit.edu

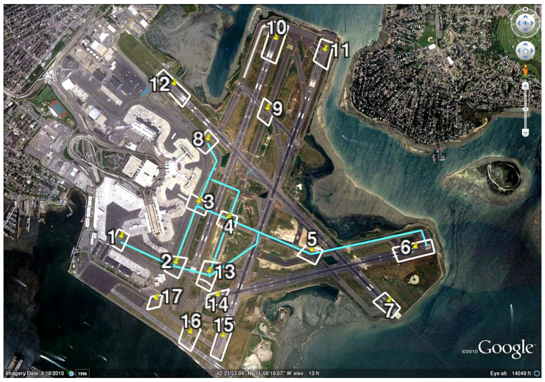

Fig. 1. Layout of the airport surface at Boston Logan. Nodes in the network model are marked with white boxes. The configuration-specific network for departures from Runway 27 has been highlighted in light blue.

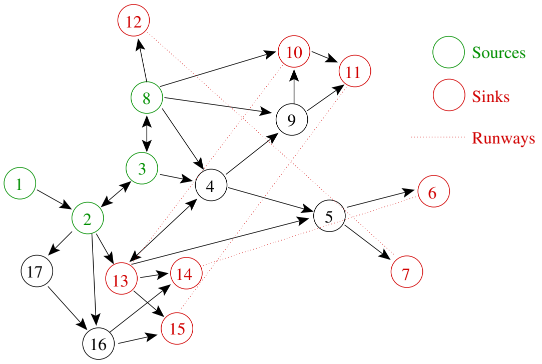

Fig. 2. Abstraction of the airport surface as the union of all configuration-specific networks, with link directionality marked. Green nodes are sources and red ones are terminal nodes.

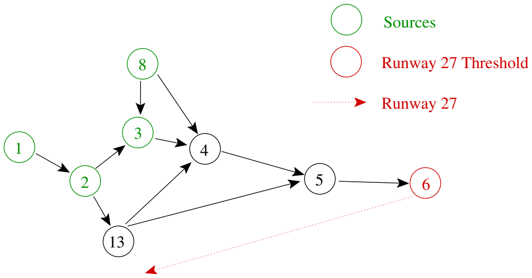

Fig. 3. Network layout for departures from Runway 27. Note that there are only a few choices for the taxi path from each source.

position, velocity, altitude and heading. The update rate is once per second for each individual flight track. For this study, we used 45 days of ASDE-X data from May and June 2011, at Boston Logan International Airport. The data set consisted of 24,636 departing aircraft, and was split into a training data set and a test data set. Raw surface tracks were processed using a multi-modal unscented Kalman filter developed in prior work [18].

II. Modeling of Taxi-out Times

In this section, we describe the model for taxi times on each link of the network, and the methods used for parameter estimation. For the sake of clarity, we first define certain terms relating to surface operations at airports. A *gate* is a parking bay for aircraft, attached to the airport terminal. This is where passengers board the aircraft. *Pushback* is the process of pushing an aircraft back from the gate, in preparation for taxi to the runway. *Pushback delay* is an instruction given to an aircraft, delaying the start of its pushback process. The aircraft is supposed to wait at the gate until it is given permission to pushback.

A. Modeling Framework

Fig. 1 shows the set of runways and taxiways on the airport surface at Boston Logan, that are represented in the network model. The taxiways form the links of the network, and their major intersections are marked as the nodes. The taxi-out phase for an aircraft is defined to be from the time an aircraft leaves the gate to the time it starts its takeoff roll from the runway threshold. Therefore, the source nodes in the network are the ones adjoining the gates, while the sink nodes are the runway thresholds. An abstraction of the resulting model is shown in Fig. 2. Note that the figure shows the union of the networks for all possible airport configurations (allocation of runways to landings and takeoffs). In practice, only one configuration is active at a time, and each aircraft has a unique source and sink node.

Fig. 3 shows the effective network layout when the airport is performing departures from Runway 27. In any specific configuration such as this one, aircraft maintain a flow from the gates to the runway, and generally do not taxi in cyclic paths. Consequently, the configuration-specific networks are directed acyclic graphs with random link travel times. Since arrivals taxiing-in at Boston Logan airport tend not to interact with departures taxiing-out, these are not specifically addressed in the proposed model. However, the stochasticity introduced by the occasional interactions between arrivals and departures is captured by the probability distribution of link travel times.

B. Model Selection and Parameter Estimation

Aircraft on the surface taxi at fairly constant velocities, occasionally stopping because of other aircraft crossing their path, or when about to cross an active runway. Notionally, the taxi-out process can thus be classified into two modes: *unimpeded taxi*, and *stationary*. We characterized these modes by generating empirical distributions of the number of stops on each link, the time spent stationary during each instance of a stop, and the unimpeded travel time distributions. The procedure is depicted in Fig. 4.

Following this, a set of theoretical distributions was selected to explain the empirical data. The Kullback-Leibler (KL) divergence from the true distribution to the model [19] was chosen as a measure of goodness of fit of a candidate model to the data. By evaluating the KL divergence over a fixed, discrete set of evaluation points, it was possible to compare the performance of different candidate models. For each candidate family of models, the KL divergence from the empirical distribution to its projection onto the family was calculated. Minimizing this ‘best’ value of KL divergence

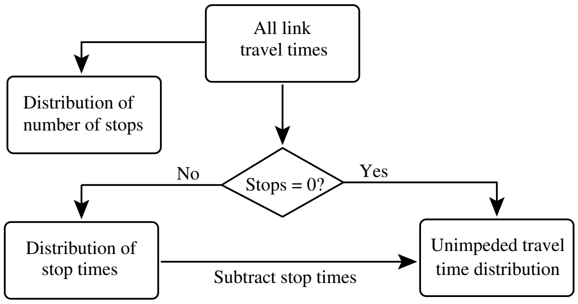
All link travel times
Distribution of number of stops
Stops = 0?
No
Yes
Distribution of stop times
Subtract stop times
Unimpeded travel time distribution

Fig. 4. Flowchart for measuring empirical distribution of link travel times.

across different candidate model families produced the final model for travel times on each link. The estimation process is described in more detail in prior work [20].

C. Model for Taxi Times

The travel time over a link l is modeled as,

$$t_l = t_{u,l} + \sum_{i=1}^{N_{s,l}} t_{s,l,i}, \tag{1}$$

- $t_{u,l} > 0$ is the unimpeded travel time over the link l, an erlang random variable with order n_l and rate λ_l,
- $N_{s,l} \in \{0, 1, 2, ...\}$ is the number of stops on the link, modeled as a geometric random variable with parameter $p_{k,l} \in [0, 1]$, where k is the current level of traffic on the surface and l is the current link,
- $t_{s,l,i} > 0$ is the stationary time corresponding to the i^{th} stop on link l, modeled as an exponential random variable with rate $\mu_l > 0$. Each $t_{s,l,i}$ is assumed independent and identically distributed (*i.i.d.*) for a given link l.

If the number of stops is $N_{s,l} = 0$, then $t_l = t_{u,l}$. Further, each instance of travel time on a link is independent of all other instances, whether on the same link or on other links, conditioned on the level of surface traffic. It is possible to calculate the probability density function for t_l in Eqn. (1), but the expressions are extremely complicated. On the other hand, a simple analytical approximation that agrees well with the exact distribution of t_l can be derived. In the interest of brevity, this aspect is not addressed in this paper. Numerical calculations, with parameters tuned as explained previously, are also tractable. A comparison between one such predicted link taxi time distribution, and an empirical distribution from an independent data set, is shown in Fig. 5. We have observed similar matches for all other frequently used links in the network.

D. Parameter Variation with Surface Traffic

It is well-known that taxi-out times at airports increase with traffic on the surface. To account for this effect, the parameters of the model need to vary as well. Let us define the *surface traffic level*, k, to be the total number of departing aircraft on the surface, that have pushed back from their gates but have not taken off yet. Empirical evidence shows that the unimpeded travel time parameters n_l and λ_l, as well as the stop time parameter μ_l, remain invariant with changes in k.

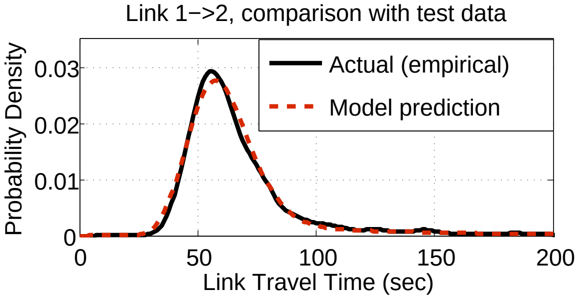

Fig. 5. Comparison of probability density of taxi times on link 1 → 2. Parameters are estimated separately for each link, and the model is then compared to independent test data (in black).

The additional taxi-out time due to congestion is accounted for by an increase in the stopping probability on each link, $p_{k,l}$. Note that this fact implicitly accounts for the 'departure queue' that forms at the runway, such as on link 5 → 7 in Fig. 1. Increased $p_{k,l}$ will lead to an increase in the number of stops, and each additional stop will add, on average, a penalty of $\frac{1}{\mu_l}$ to the queuing and taxi-out times. The average travel time for each link l can be calculated by taking the expectation of both sides of Eqn. (1).

$$\mathbb{E}[t_l \mid k] = \frac{n_l}{\lambda_l} + \mathbb{E}[N_{s,l}|k]\frac{1}{\mu_l} = \frac{n_l}{\lambda_l} + \frac{p_{k,l}}{1-p_{k,l}}\frac{1}{\mu_l}. \tag{2}$$

We now assume that each additional aircraft on the surface adds a fixed time penalty to the expectation in Eqn. (2). Denoting this penalty as a fraction X_l of the expected time per stop $\frac{1}{\mu_l}$, it may be observed that for an aircraft that pushes from its gate at a traffic level of $(k+1)$ instead of k,

$$\mathbb{E}[t_l \mid k+1] = \frac{n_l}{\lambda_l} + \frac{p_{k,l}}{1-p_{k,l}}\frac{1}{\mu_l} + \frac{X_l}{\mu_l}. \tag{3}$$

Comparing Eqn. (3) with Eqn. (2) evaluated at $(k+1)$, the following relation is seen:

$$\frac{p_{k+1,l}}{1-p_{k+1,l}} = \frac{p_{k,l}}{1-p_{k,l}} + X_l.$$

This property describes a telescoping series given by,

$$\frac{p_{k,l}}{1-p_{k,l}} = \frac{p_{0,l}}{1-p_{0,l}} + kX_l,$$
$$\implies p_{k,l} = \frac{p_{0,l} + kX_l(1-p_{0,l})}{1+kX_l(1-p_{0,l})}. \tag{4}$$

Eqn. (4) has some desirable properties from the perspective of the system under consideration:

- $p_{k,l} \in [0, 1] \; \forall \; k$, if $p_{0,l} \in [0, 1]$.
- $p_{k,l}$ is non-decreasing in k, which means that increased congestion increases the probability of stopping.
- $p_{k,l}$ approaches 1 as k increases, but does not equal 1 unless $p_{0,l} = 1$. This is important, because $p_{k,l} = 1$ would make the expectation in Eqn. (2) infinite.
- For each aircraft on the surface, the expected number of stops on the current link $(l_i, \; i \in \{1, ..., k\})$ is $\frac{p_{k,l_i}}{1-p_{k,l_i}}$.

The expected total number of stops by aircraft currently on the surface while they are still on their current link, is therefore:

$$\mathbb{E}[N_{tot}] = \sum_{i=1}^{k} \frac{p_{k,l_i}}{1-p_{k,l_i}} = \sum_{i=1}^{k} \left(\frac{p_{0,l_i}}{1-p_{0,l_i}} + kX_{l_i} \right).$$

Note that $\mathbb{E}[N_{tot}]$ increases quadratically with k (the summation makes the first term linear in k, and the second term quadratic). To estimate the number of conflicts on the surface, we (i) multiply $\mathbb{E}[N_{tot}]$ by a factor of 0.5, assuming that a conflict occurs between two aircraft, causing them both to stop, and (ii) multiply by a constant factor representing the average number of links traversed by each aircraft. After this manipulation, we see that the dominant behavior is still quadratic, and is therefore in close agreement with prior empirical studies [21].

Finally, it is seen that the variation of $p_{k,l}$ as given by Eqn. (4) is in close agreement with empirical data, where the probability of stopping was independently measured by looking at the fraction of aircraft that stopped on a link, as a function of the level of traffic on the surface when the aircraft entered the link. Fig. 6 shows a sample result for the link 5→7.

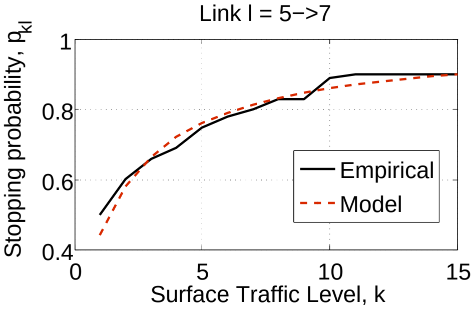

Fig. 6. Variation of stopping probability on link 5→7 with the level of surface traffic. The model curve corresponds to Eqn. (4) with suitably tuned parameters.

III. Network Performance

The theoretical limits of network performance for the airport surface are derived here, using the model for the taxi-out process with its parameters estimated as described in the previous section. A natural measure of performance in this case is the maximum network throughput. The following discussion will focus upon the derivation of these theoretical limits and an aircraft pushback time algorithm that aims to balance surface congestion with the throughput shortfall from the theoretical maximum.

A. Maximum Link Throughput

Before analyzing a generic network, we derive the maximum throughput of a network containing a single link l, assuming it can accommodate an infinite number of aircraft. Furthermore, an initial assumption is that the link is operating in deterministic steady state, with the taxi time of each aircraft being equal to its expectation value. Aircraft are distributed regularly along its length, each *departure from* the link occurs after a fixed time interval, and each *arrival to* the link happens at the same. Thus the number of aircraft on the link always stays the same. If ζ_l is the arrival rate to the link, under this assumption, a new arrival occurs every $\left(\frac{1}{\zeta_l}\right)$ seconds. From Eqn. (3), note that the (deterministic) taxi time on the link is,

$$\mathbb{E}[t_l|k] = \frac{n_l}{\lambda_l} + \frac{p_{0,l}}{(1-p_{0,l})\mu_l} + \frac{kX_l}{\mu_l} = \eta_l + \frac{kX_l}{\mu_l}, \qquad (5)$$

where η_l is a constant comprised of the first two terms. During this time interval in which an aircraft travels over the link, the k aircraft ahead of it depart from the link. Therefore, the inter-departure time $\Delta t_{k,l}$ is,

$$\Delta t_{k,l} = \frac{\mathbb{E}[t_l|k]}{k} = \frac{\eta_l}{k} + \frac{X_l}{\mu_l}. \qquad (6)$$

In steady state, k is the result of equating the inter-arrival interval $\left(\frac{1}{\zeta_l}\right)$ to $\Delta t_{k,l}$. Consequently, the minimum inter-arrival interval that can be sustained by the link is $\left(\frac{1}{\zeta_l}\right)_{\text{min}} = \frac{X_l}{\mu_l}$, achieved as $k \to \infty$. The maximum sustained throughput of link l is defined as the inverse of this value,

$$\sigma_l \triangleq (\zeta_l)_{\text{max}} = \frac{\mu_l}{X_l}. \qquad (7)$$

In the stochastic case, the average inter-departure times will be governed by the expectation in Eqn. (5). Therefore, the result from Eqn. (7) still holds. Relaxing the assumption of infinite link capacity is also quite simple. Assuming the maximum capacity of the link to be k_{max}, the minimum sustained inter-departure time, as derived from Eqn. (6), will be $\left(\frac{1}{\zeta_l}\right)_{\text{min}} = \frac{\eta_l}{k_{\text{max}}} + \frac{X_l}{\mu_l}$. The maximum throughput will be the inverse of this quantity. In further analysis in this paper, only infinite-capacity links are considered, with the knowledge that all results may be extended for finite-capacity links. If the input rate to a link l is less than σ_l, the link will be referred to as being *stable*.

B. Maximum Network Throughput

Once the maximum throughput values for each link are known, it is straightforward to derive the maximum network throughput. Assume that the maximum throughput values, $\sigma_l = \frac{\mu_l}{X_l}$, for each link l are known. Then the maximum throughput of a generic directed, acyclic graph can be found using the mincut/maxflow theorem [22], [23]. This theorem states that the maximum flow rate through a network is equal to the maximum flow rate through the most constrained cut across the network. A *cut* with respect to two terminals is defined to be "a set of branches such that when deleted from the network, the network falls into two or more unconnected parts with the two terminals in different parts" [23]. In a single-link network, there is only one possible cut and the maximum throughput is thus trivially equal to σ_l. The method for deriving the throughput for more complicated networks is given in detail in [22] and [23].

IV. Network Control

A. Stability Considerations

As shown above, it is possible to derive the maximum input rates to a generic network, including the one in Fig. 1. If the input rate to any link is less than its maximum throughput, the link is said to be in a *stable* condition. Note that in general, simply ensuring that each individual link remains stable does not ensure stability of the entire network [11], [12]. However, it is known that for the special case of directed acyclic graphs, such stability is guaranteed.

The proposed implementation protocol in this work is to suggest *pushback times* to aircraft that are ready to depart from their gates. In essence, the control variables are the times of entry of aircraft into the network. Consequently, the control algorithm needs to assign a pushback time to each aircraft as it calls ready, while ensuring the stability of the network.

B. Formulation of Control Strategy

A control algorithm that aims to maintain a steady level of traffic on the airport surface, is described here. It has been shown [16] that regulating traffic on the airport surface to a well-chosen level results in fuel savings. Therefore, this objective is a logical starting point for any new control strategy. Consider a scenario where an aircraft calls ready to pushback at time $t = 0$. Now consider a First-Come-First-Served (FCFS) algorithm that aims to minimize a weighted linear cost function for the aircraft, composed of *pushback delay* t_p and *expected taxi-out time* t_T (equal to $\mathbb{E}[t_l|k]$ from Eqn.(2)). That is, the cost function is $\mathcal{C} = a \cdot t_p + t_T$, with a being a constant weighting factor. The control value t_p that minimizes expected cost is,

$$\begin{aligned} t_p^* &= \arg\min_{t_p \geq 0} \; \mathbb{E}\left[a\, t_p + t_T\right] \\ &= \arg\min_{t_p \geq 0} \; \left(a\, t_p + \sum_{l=l_1}^{l_r} \left[\frac{n_l}{\lambda_l} + \frac{p_{k_p,l}}{1 - p_{k_p,l}} \frac{1}{\mu_l} \right] \right) \\ \Rightarrow t_p^* &= \arg\min_{t_p \geq 0} \; \left(a\, t_p + \sum_{l=l_1}^{l_r} \left[\eta_l + \frac{k_p X_l}{\mu_l} \right] \right), \qquad (8) \end{aligned}$$

where the last two steps follow from Eqn. (2) and the definition of η_l. Eqn. (8) assumes that the aircraft's route follows links $l = l_1, l_2, ..., l_r$, and that $p_{k_p,l}$ is the stopping probability on link l, when the projected surface traffic level at t_p is k_p. The expected taxi time is thus also a function of t_p. The projected traffic level, k_p, can be calculated based on the expected times between successive departures, as given in Eqn. (6). Since entry into the network is assumed FCFS, the projected traffic level decreases as t_p increases (there can be no additional aircraft entering the network while the current aircraft is waiting). Also, Eqn. (6) shows that the expected time between departures increases as k_p decreases. Therefore, at some value of k_p, the increase in expected cost due to the first term in Eqn. (8) is going to outweigh the decrease in cost due to a smaller k_p in the second term. The actual value of this 'target' k_p is controlled by the weight a.

C. Incorporation of Target Traffic Level

To develop a control strategy from an analytical perspective, some simplifying assumptions about the departure process need to be made. The following derivation will be helpful in providing intuition about the departure process. Furthermore, simulations carried out using the full-scale model with no simplifying assumptions (described in the next section) show that the ensuing control strategy is still valid.

For simplicity, first consider the single-link network. For moderately large values of surface traffic k, it may be assumed that departures from the link occur as independent exponential processes, one for each aircraft. If there are k aircraft on the link, there are k racing exponential processes. Using the memoryless property [24], the expected departure time relative to the present time, of each aircraft is derived from Eqn. (2) to be equal to $\eta_l + \frac{kX_l}{\mu_l}$. Consequently, the rate of each process is the inverse of this quantity, and the net departure rate for k independent exponential processes is,

$$R_{kl} = k \cdot \frac{1}{\eta_l + \frac{kX_l}{\mu_l}} \text{ aircraft per unit time.}$$

Now since each departure from the link corresponds to an expected taxi time reduction of $\frac{X_l}{\mu_l}$ for the aircraft being assigned pushback delay, the instantaneous rate of decrease of expected taxi time is given by,

$$-\frac{d}{dt} t_T = R_{kl} \, \frac{X_l}{\mu_l}.$$

It is possible to derive a simple expression for the weight a in Eqn. (8), based on the assumptions made regarding the departure process from the network. In order to target a certain level of traffic (say $k = k_{\text{ctrl}}$), set the value of the weight such that the rate of reduction of expected taxi time is equal to the rate of increase of the term $a\, t_p$, when $k = k_{\text{ctrl}}$. Since the term $a\, t_p$ increases at a constant rate a,

$$\begin{aligned} -\frac{d}{dt} t_T &= a \\ \Rightarrow a = R_{k_{\text{ctrl}} l} \, \frac{X_l}{\mu_l} &= \frac{k_{\text{ctrl}} \frac{X_l}{\mu_l}}{\eta_l + k_{\text{ctrl}} \frac{X_l}{\mu_l}} \qquad (9) \end{aligned}$$

Eqn. (9) relates the target traffic level k_{ctrl} to the weighting factor a, and vice-versa.

D. Derivation of Optimal Control Values

Since there is no entry of new aircraft into the link until the current aircraft pushes back, R_{kl} is the rate of decrease of the projected traffic level k_p. This makes it possible to find an expression for the optimal control value t_p required to target a specific value of k_p. An implicit assumption is that the exponential nature of the departure process for each aircraft is maintained as this projected value evolves. If $k_p(t_p)$ is the projected traffic level after time t_p,

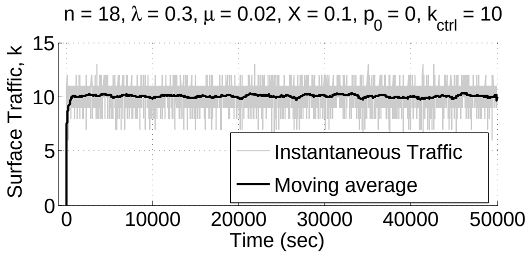

Fig. 7. Simulation of control algorithm for single link, with $k_{\mathrm{ctrl}} = 10$. The traffic level k can be seen to stabilize to k_{ctrl} after an initial transient.

$$\frac{d}{dt}k_p(t_p) = -R_{k_pl} = -\frac{k_p}{\eta_l + k_p\frac{X_l}{\mu_l}}$$

$$\Rightarrow t_p = \eta_l \ln\frac{k_p(0)}{k_p(t_p)} + (k_p(0) - k_p(t_p))\frac{X_l}{\mu_l}. \qquad (10)$$

Since $k_p(0)$ is a known quantity (the current traffic level), the optimal control for each $k = k_p(0)$ is defined by substituting $k_p(t_p) = k_{\mathrm{ctrl}}$ in Eqn. (10). If the optimal value is negative, t_p is assigned a value of zero, in order to obey the constraint $t_p \geq 0$. Note that this happens iff $k_p(0) < k_{\mathrm{ctrl}}$, which means that the control strategy calls for immediate pushback if the traffic level is below the target value. If the current traffic level is above k_{ctrl}, the pushback delay becomes progressively larger with $k_p(0)$. Another point to note is that for every value of $k = k_p(0)$, there is a unique control t_p^* that is commanded.

V. Simulation of Control Strategy

In this section, the model developed in Sec. II-C is used to simulate the taxi-out process, with the algorithm developed in Sec. IV-D controlling entry of aircraft into the network.

A. Single-Link Case

Since the optimal control strategy as derived above relies on a number of approximations, it is necessary to validate it using independent simulations. In Figs. 7 and 8, simulation results are shown for a single-link network. It is assumed that there is infinite demand (aircraft waiting to push back), and that the current traffic level is known at all times. In addition, aircraft are released according to a First-Come-First-Served (FCFS) policy. From Fig. 7, the average steady-state traffic level is seen to stabilize to k_{ctrl}. Fig. 8 shows that the average taxi times are in close agreement with Eqn. (2). Note that in this example, the choice of k_{ctrl} adds on average, $\frac{\eta_l}{k_{\mathrm{ctrl}}} = 6$ seconds to the inter-departure times (as predicted by Eqns. (6) and (7)). This throughput shortfall can be reduced to a value arbitrarily close to zero, by increasing k_{ctrl}.

B. Complete BOS Network Example

In Fig. 9, results from a simulation based on the full network abstraction of Boston Logan airport are shown. All departures are assumed to happen from Runway 27 (marked as node 6 in Figs. 1 and 3). It is assumed that there is now a finite number of aircraft waiting for release. The pushback requests from these aircraft appear as a Poisson process with a rate based on the historical variation over a day, as derived from surface surveillance data. The x-axis in Fig. 9 shows the local time at the airport. Each curve in the figure plots the variation of simulated surface traffic over the day, for different values of k_{ctrl}. The curve for unrestricted entry into the network ($k_{\mathrm{ctrl}} = \infty$) clearly shows the morning and evening demand peaks, with high levels of surface congestion during these times. The case with $k_{\mathrm{ctrl}} = 15$ mitigates this effect to a great extent, by delaying aircraft at the gate during periods of excessive demand. Note that the peaks for this case are wider than the peaks for the unrestricted case, as the control algorithm clears out delayed aircraft. On the other hand, the case with $k_{\mathrm{ctrl}} = 10$ turns out to be too aggressive, with the algorithm not being able to clear the built-up demand until the end of the day. However, in both restricted cases, the control strategy is successfully able to limit surface traffic levels to the corresponding target values in times of high departure demand.

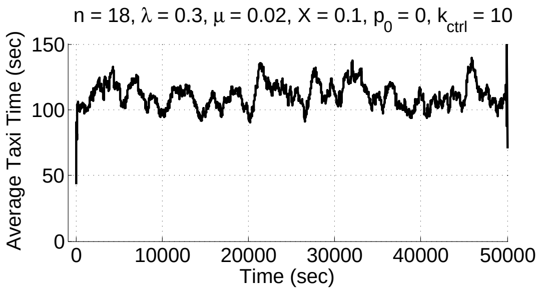

Fig. 8. Simulation of control algorithm for single link, with $k_{\mathrm{ctrl}} = 10$. The average taxi times are seen to agree with Eqn. (2).

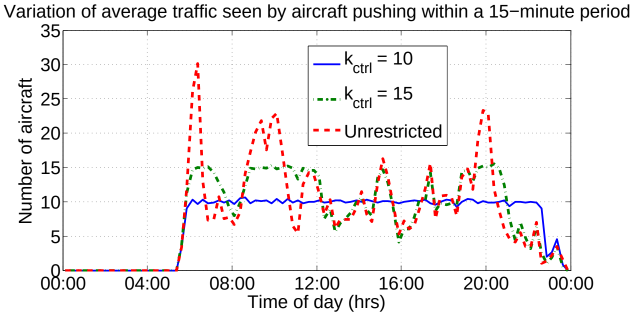

Fig. 9. Simulation of control algorithm for the Boston Logan network, for different values of k_{ctrl}. Departures are assumed to be happening from Runway 27. Taxi paths for each aircraft are assumed to be pre-defined.

VI. Conclusions

A. Discussion of Approach

The simulations above showed that the control algorithm described here is capable of maintaining a steady traffic level on the airport surface under realistic conditions. Since the approach is not limited by the large time constants of algorithms previously available in literature, it reduces the variability in traffic levels seen in these studies. However, there is potential for further improvement by building on these results. Specifically, the primary objective of congestion control methods is to reduce the fuel consumption and emissions on the surface. While maintaining steady, reasonable traffic levels does this to some extent, even greater savings may be possible by optimizing for these objectives directly. Therefore, there is incentive to develop control strategies that explicitly aim to minimize aircraft fuel consumption and emissions by considering the behavior of aircraft and their engines [25].

Finally, the applicability of any proposed strategy in the current air traffic control framework is an essential consideration. Implementation of the algorithm proposed in this paper would require a continuous knowledge of current traffic levels on the airport surface. Since real-time ASDE-X feeds are already available at most major airports within the US, this is a feasible requirement. Another issue that needs to be addressed is that of fairness, in terms of the order in which aircraft depart as compared to the order in which they call ready for pushback. Although the FCFS policy is perceived to be the most equitable by airlines and air traffic controllers, it may not result in the best possible performance, in terms of fuel savings and departure throughput. Therefore, a balance needs to be struck between the performance objectives and the perceptions of the stakeholders in the system. Control algorithms with a specific focus on handling these requirements are currently under development [26].

B. Summary and Future Work

This paper modeled the aircraft taxi-out process by representing the airport surface as a network and the total taxi time as the sum of the travel times along different links in the surface trajectory of an aircraft. A set of suitable random processes was proposed and validated to model the distribution of link travel times. A control algorithm that attempts to balance performance and congestion control objectives, while maintaining network stability was also proposed. Using the test cases of a simple, single-link model and the complete network for one runway configuration at Boston Logan, it was shown that the algorithm can successfully maintain the level of surface traffic at a specified average value.

References

[1] E. Feron, R. J. Hansman, A. R. Odoni, R. Cots, B. Delcaire, W. Hall, H. Idris, A. Muharremoglu, and N. Pujet, "The departure planner: A conceptual discussion," White paper, MIT, International Center for Air Transportation, December 1997.

[2] C. Brinton, J. Krozel, B. Capozzi, and S. Atkins, "Improved taxi prediction algorithms for the surface management system," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, Monterey, CA, August 2002.

[3] H. Balakrishnan and Y. Jung, "A framework for coordinated surface operations planning at Dallas-Fort Worth International Airport," in *AIAA Guidance, Navigation, and Control Conference*, Hilton Head, NC, August 2007.

[4] I. Simaiakis and H. Balakrishnan, "Queuing models of airport departure processes for emissions reduction," in *AIAA Guidance, Navigation and Control Conference and Exhibit*, Chicago, IL, August 2009.

[5] H. Idris, J.-P. Clarke, R. Bhuva, and L. Kang, "Queueing model for taxi-out time estimation," *Air Traffic Control Quarterly*, vol. 10, no. 1, pp. 1–22, January 2002.

[6] F. Carr, A. Evans, J.-P. Clarke, and E. Feron, "Modeling and control of airport queueing dynamics under severe flow restrictions," in *American Control Conference*, vol. 2, 2002, pp. 1314–1319.

[7] A. Eryilmaz and R. Srikant, "Fair resource allocation in wireless networks using queue-length-based scheduling and congestion control," in *INFOCOM 2005, 24th Annual Joint Conference of the IEEE Computer and Communications Societies, Proceedings IEEE*, vol. 3, March 2005, pp. 1794–1803.

[8] M. Neely, E. Modiano, and C.-P. Li, "Fairness and optimal stochastic control for heterogeneous networks," *Networking, Transactions on, IEEE/ACM*, vol. 16, no. 2, pp. 396–409, April 2008.

[9] F. Paganini, J. Doyle, and S. Low, "Scalable laws for stable network congestion control," in *Decision and Control, 2001. Proceedings of the 40th IEEE Conference on*, vol. 1, 2001, pp. 185–190 vol.1.

[10] S. Low, F. Paganini, J. Wang, S. Adlakha, and J. Doyle, "Dynamics of TCP/RED and a scalable control," in *INFOCOM 2002, Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies, Proceedings IEEE*, vol. 1, 2002, pp. 239–248.

[11] J. Perkins and P. Kumar, "Stable, distributed, real-time scheduling of flexible manufacturing/assembly/diassembly systems," *Automatic Control, IEEE Transactions on*, vol. 34, no. 2, pp. 139–148, February 1989.

[12] P. Kumar and T. Seidman, "Dynamic instabilities and stabilization methods in distributed real-time scheduling of manufacturing systems," *Automatic Control, IEEE Transactions on*, vol. 35, no. 3, pp. 289–298, March 1990.

[13] S. B. Gershwin, "Design and operation of manufacturing systems: the control-point policy," *IIE Transactions*, vol. 32, pp. 891–906, 2000.

[14] E. Nikolova, M. Brand, and D. Karger, "Optimal route planning under uncertainty," in *Automated Planning and Scheduling, International Conference on*, Lake District, England, June 2006.

[15] E. Nikolova and D. Karger, "Route planning under uncertainty: The Canadian traveller problem," in *Artificial Intelligence, Twenty-Third Conference on*, Chicago, IL, July 2008.

[16] I. Simaiakis, H. Khadilkar, H. Balakrishnan, T. G. Reynolds, R. J. Hansman, B. Reilly, and S. Urlass, "Demonstration of reduced airport congestion through pushback rate control," in *USA/Europe Air Traffic Management Research and Development Seminar*, Berlin, Germany, June 2011.

[17] *ASDE-X Brochure - ASDE-X 10/05.qxd*, Sensis Corporation, East Syracuse, NY, 2008.

[18] H. Khadilkar and H. Balakrishnan, "A multi-modal unscented Kalman filter for inference of aircraft position and taxi mode from surface surveillance data," in *Aviation Technology, Integration and Operations, AIAA Conference on*, Virginia Beach, VA, September 2011.

[19] S. Kullback and R. Leibler, "On information and sufficiency," *Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, 1951.

[20] H. Khadilkar, "Analysis and modeling of airport surface operations," Master's thesis, Massachusetts Institute of Technology, June 2011.

[21] A. Barnett, G. Paull, and J. Iadeluca, "Fatal US runway collisions over the next two decades," *Air Traffic Control Quarterly*, vol. 8, 2000.

[22] L. Ford and D. Fulkerson, "Maximal flow through a network," *Canadian Journal of Mathematics*, pp. 399–404, 1956.

[23] P. Elias, A. Feinstein, and C. Shannon, "A note on the maximum flow through a network," *Information Theory, IRE Transactions on*, vol. 2, no. 4, pp. 117–119, December 1956.

[24] R. Gallager, *Discrete Stochastic Processes*. Kluwer Academic Publishers, 1996, chapter 2: Poisson Processes.

[25] H. Khadilkar and H. Balakrishnan, "Estimation of aircraft taxi-out fuel burn using Flight Data Recorder archives," in *AIAA Guidance, Navigation, and Control Conference*, Portland, OR, August 2011.

[26] H. Khadilkar and H. Balakrishnan, "Control of aircraft pushbacks at an airport using a dynamic programming formulation," in *AIAA Guidance, Navigation, and Control Conference (submitted)*, Minneapolis, MN, August 2012.