Janet: A Better Airport Queue-Sorting Model

Derek Gerlach
Mark Christopher Haase
Charles H. Martin
Ohio State University
Columbus, Ohio 43210

Advisor: T. Ralley

Summary

When the number of aircraft requesting pushback (departure) exceeds the airport runway capacity, the airport must employ some rationale for deciding in what order aircraft should take off.

Our problem is to develop and test a queue-sorting method that assigns takeoff priorities based on airlines' and passengers' satisfaction, as an alternative to first-come-first-served.

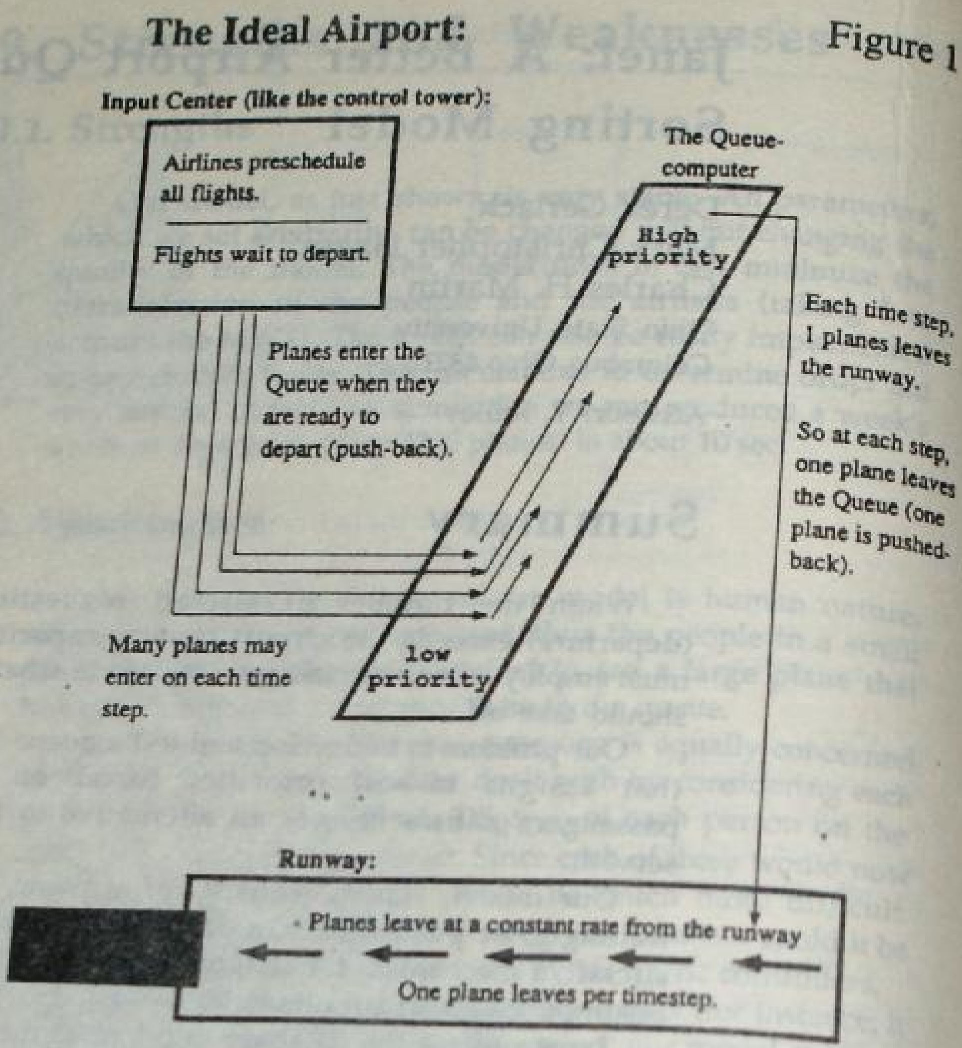
Our model, "Janet," consists of a queue (list) of aircraft waiting to be pushed back, a database of information for each aircraft, a mechanism for assigning takeoff priorities, and some assumptions about airports.

Janet utilizes the database to calculate the "importance" of every aircraft in the queue at each time step. The aircraft of maximum importance at each time step is removed from the queue and launched. The importance function, I, naturally separates into three components:

$$I = I_1 + I_2 + I_3.$$

I_1 is a function only of the number of people in the plane, and how long they have to wait for their plane to be pushed back. I_2 depends on the number of people waiting to make a connection and the remaining time before they miss it. I_3 depends only on how late the plan is. One can show that the first-come-first-served model is a special case of Janet with $I = I_3$.

To test the model, we wrote a computer program simulating both the first-come-first-served model and Janet. The program



Our Problem: to re-design the sorting on the Queue as to better satisfy the passengers and the airlines.

measured the number of hours wasted by people sitting in the queue and the number of people who missed their connections. We showed that with a proper choice of the constants, Janet decreased both wasted hours and missed connections, outperforming first-come-first-served.

Problem and Objective

Uncongested airports direct aircraft to runways on a first-come-first-served basis, depending on when pilots request departure clearance (i.e., a pushback request). But when takeoff traffic exceeds an airport's capacity, the airport must implement priority-based takeoff control procedures that optimize airlines' and passengers' satisfaction. Our problem consists of two parts: first, to develop a mathematical model to quantify airlines' and passengers' satisfaction, as a means for assigning takeoff priorities; and second, to evaluate this model, using a computer simulation.

Assumptions

The following assumptions are used in the construction of a model of an ideal airport (see Figure 1). Within the framework of this ideal airport, we develop and test an alternative priority sorting method.

1. We interpret a pushback in the following manner: as soon as an aircraft is ready to leave the gate, the pilot requests the aircraft be pushed onto the runway.

2. When an aircraft is ready to be pushed back (at the time of scheduled pushback), it is added to a list called the queue. An aircraft's priority depends on its position in the queue. Because an airport has a finite capacity for holding aircraft, the queue will be assumed to have a finite length.

3. The tower accesses an online database with the following information:

   a) the time an aircraft is scheduled for pushback, t_{sp}
   b) the time it actually pushes back, t_{ap}
   c) the number of passengers on board, N

d) the number of passenger scheduled to make a connection at the next stop, N_c, as well as the time of the connection, t_c (for simplicity, assumed to be the same for all passengers scheduled to make a connection)

e) the scheduled time of arrival at the destination, t_{sa}.

Although aircraft capacity ranges from 100 to 400 persons (in steps of 50), individual aircraft capacities are not listed in the database.

4. The queue samples time discretely, with time interval equal to the amount of time required to launch an aircraft once it is cleared for takeoff. (The length of this interval depends on the airport in question). The scheduled pushback time (t_{sa}), the time an aircraft actually pushes back (t_{ap}), the scheduled arrival time (t_a), and the scheduled connection time (t_c) all fall on these discrete intervals.

5. The airport has an amazingly efficient runway plan: An aircraft ordered to push back from any gate can be quickly launched. When an aircraft is removed from the queue, it is pushed onto a runway and is launched by the next time interval. Therefore, once an aircraft is removed from the queue, it is thereafter considered to be airborne.

6. One aircraft leaves the queue in each time interval, and therefore one aircraft takes off in each time interval. Hence, our ideal airport has only one runway.

7. Airline flight schedules do not account for the time that an aircraft may spend in the queue. (We also assume that all aircraft fly at a constant rate and therefore cannot adjust for being late by speeding up.) That is, airlines form their flight schedules assuming an aircraft will be launched when it requests pushback. Therefore, the amount of time an aircraft spends in the queue corresponds to how late the aircraft will be.

8. Because we are interested only in the sorting mechanism of the queue, we construct an ideal airport consisting of only three parts: an input center, the queue control, and an output device (the runway) (Figure 1).

The *input center* submits to the queue at each time interval a list of aircraft desiring pushback. The input center of a real airport might consist of the airlines, who schedule flights, and a physical area for the aircraft to sit and wait for departure. The *queue control* holds the list of aircraft waiting to be pushed back, and also sorts the aircraft on a priority basis. The runway acts as an *output device* for the queue. As stated above, once an aircraft is removed from the queue, it is considered immediately airborne; therefore, there is no waiting line on the runway.

3. Motivation for the Design

Different airports have different objectives. For instance, while Moscow International Airport might very well place higher priority on flights making connections, Mayberry Airport might place higher priority on flights that are completely booked. Our model must be flexible enough to incorporate the takeoff priority policies of individual airports.

Given the information on the online database, we identified four variables that have an effect on the urgency for an aircraft to depart:

1. How late the aircraft will be arriving at its next destination,
2. The amount of layover before a connection is missed,
3. The amount of time that an aircraft spends in the queue,
4. The number of passengers on board and the number who need to make the connecting flight.

In order to design a queue management system that maximizes airline and passenger satisfaction, we must first examine what constitutes airline and passenger *dissatisfaction*.

4. Effects of Delaying Departure

Why would delaying an aircraft's departure dissatisfy the passengers?

1. Delaying a flight wastes time. Wasted hours are proportional to the number of passengers on board and to the amount of time the aircraft spends in the queue.

2. The schedules of the passengers (and their loved ones or business associates) are thrown off. Passenger dissatisfaction based on inconvenience is proportional to the number of passengers on board and to the amount of time the aircraft spends in the queue.

3. A passenger may miss a connecting flight. Passenger dissatisfaction based on a missed connection is proportional to the number of people making the connection, as well as to the amount of layover time the aircraft has left until all passengers will surely miss the connection.

Why would delaying an aircraft's departure dissatisfy the airline?

4. When expensive aircraft sit on the ground rather than provide transportation to customers, they are not making any money. The airline needs the aircraft for another flight at the next stop. Dissatisfaction in this respect is proportional to the time spent in the queue, independent of the number of passengers on board.

5. Anything that upsets the customer will surely upset the airline. Hence, all comments in reference to dissatisfaction of the passenger also apply to the dissatisfaction of the airlines.

5. The Mathematics of Janet

In order to decide which aircraft should be removed from the queue, we assign an *importance* to each aircraft by means of an importance function. The queue will select an aircraft of highest importance and remove it.

First, we observe that the importance of an aircraft must reflect the degree to which the aircraft contributes to passenger and airline dissatisfaction. The five factors contributing to airline and passenger dissatisfaction are discussed above. For simplicity, we assume that our importance function can be separated into components, corresponding to these factors.

The importance of wasted hours incurred by a delayed departure is denoted by I_1:

$$I_1 = a_1 N(t - t_{sp})^{b_1}$$

where a_1, b_1 are constants and $(t - t_{sp})$ is the time an aircraft spends in the queue. Thus, I_1 depends on both the number of passengers on board and the amount of time the aircraft spends in the queue.

We have assumed that the amount of time spent in the queue is equal to the delay of the aircraft's arrival. Therefore, the passenger dissatisfaction incurred by a late arrival depends on the same variables as the number of passenger-hours spent in the queue. Due to this interrelation, we will not add a new term to our importance function, but rather adjust a_1 and b_1 to reflect this additional contribution.

The second term of our importance function reflects the urgency for an aircraft to make its next connection. We have assumed that this should increase with the number of people making connections. Moreover, it should also depend on the amount of layover at the next stop before a connection is impossible. The function is:

$$I_2 = \begin{cases} a_2 N_c^{b_3} [(t_c - t_{sa}) / [a_6 + (t_c - t_{sa}) - (t - t_{sp})]]^{b_4} & \text{if } t < t_c - t_{sa} \\ 0 \quad \text{otherwise,} & \end{cases}$$

where the a's and b's are constants. Note that I_2 depends on the number of people making the connection, the length of the layover, and the amount of time the aircraft spends in the queue.

The next term of the importance function reflects the airline dissatisfaction caused by the aircraft's idling in the queue. Because this is independent of the number of passengers on board, it must be considered independent of I_1:

$$I_3 = a_3 (t - t_{sp})^{b_5},$$

where a_3 and b_5 are constants. This importance contribution depends only on the amount of time an aircraft spends in the queue.

Because an unhappy passenger is lost business, anything that dissatisfies the passengers also dissatisfies the airlines. Therefore, we should not take into account separately airline dissatisfaction due to passenger dissatisfaction, since the overall importance value already includes passenger dissatisfaction. By changing the constants, these terms can be simply absorbed into the existing importance function.

We conclude that there are only three terms affecting dissatisfaction: the time passengers spend waiting (I_1), the urgency for an aircraft to make its connection (I_2), and the time an aircraft has been waiting for departure (I_3). The overall importance of an aircraft can be calculated by summing these three components:

$$I = I_1 + I_2 + I_3.$$

6. Analyzing the Model

The analysis of the model Janet consists of two parts. First, the terms in the importance function I are examined in detail. Second, a computer program is constructed to simulate the ideal airport. We present the results of a comparison between Janet and the first-come-first-served model.

When an aircraft enters the queue, its importance function is calculated at time t. Each term of I contains undetermined constants. By changing these constants, we can customize the importance function to suit particular airline and passenger priorities.

It is extremely important to note that *within a given time interval* one aircraft may request pushback before another. In the event that two aircraft enter the queue in the same time interval with the exact same parameters, the importances assigned to them will be equal. Our queue-sorting method will then resort to the original method and launch the aircraft on a first-come-first-served basis.

6.1. Behavior of I_1 and I_3

What happens to the importance of a plane due to its I_1 factor? Suppose two aircraft enter the queue, but aircraft A has 400 people, and aircraft B has 300 people, so that $I_1(A) > I_1(B)$.

Janet uses the I_1 factor to push back aircraft A first. Increasing the exponent b_1 will change the shape of the curve over time. In the case b_1=3, the plot flattens out when $(t - t_{sp})$ is small. An airport can alter b_1 to allow for flexibility in layover times and flight speeds (which we ignored in our model). The constant a_1 simply acts as a scaling factor. For example, in the computer simulation, the time $(t - t_{sp})$ is of order 10, and N is of order 10^2(at most). Therefore, a_1 was chosen to be of order 10^{-3}. This type of argument is used to scale all of the components of I.

I_3 has the same form as I_1, except it is not weighted by N. Notice that as time spent in the queue, $(t - t_{sp})$, increases, I_3 increases (and quite rapidly if b_5 is large). This feature insures that no plane ever gets stuck in the queue!

6.2. Behavior of I_2

I_2 tries to ensure that the passengers making a connection do not miss it. Increasing b_3 places more importance on the number of people waiting. Increasing b_4 places more emphasis on missing the connection time:

$t_c - t_{sa}$ = layover time before next connection

$t - t_{sa}$ = time spent in the queue

So when $(t - t_{sa})$ increases, I_2 increases, but it is proportional to the layover time as well. If I_2 depended solely on $(t - t_{sa})$, then no priority would be given to the layover time, and planes having great amounts of layover time (time to spare) might be pushed back before planes having little layover time.

That is, suppose aircraft A and B enter the queue on the same time step with the exact same parameters, except that $t_c(A) > t_c(B)$, and A requested pushback before B. Although A has a larger layover time than B, A would be pushed out before B because it came first in the list: $(t_c - t_{sa})(A) > (t_c - t_{sa})(B)$.

Also, notice that if an aircraft has no hope of making its connection, I_2 is zero.

Table 1

Plane Data

No.	tsp	tsa	tc	N	Nc
1	0	300	340	200	150
2	0	300	340	150	100
3	0	300	340	400	2
4	10	100	120	100	0
5	10	130	170	200	23
6	10	200	500	400	300
7	10	270	290	300	173
8	20	800	810	100	90
9	20	800	810	100	5
10	20	400	640	95	33

All times are in minutes.

The simulation runs from t = 0.0 minutes until all of the planes have left the Queue.

Analysis of 5 runs of janet vs the 1st-come 1st-serve model

trial	wasted manhours	No. missed connections	No. people missing conn.	planes that missed ...	changed parameter
1st come 1st serve	1102.5	4	268	3,7,8,9	
1	877.5	5	291	4,5,7,8,9	
2	902.5	3	95	4,8,9	b4 = 2.0
3	894.17	4	268	4,7,8,9	b4 = 2.0, b3 = 0.5
4	752.5	4	195	2,4,8,9	b4 = 2.0, b2 = 2.0
5	752.5	4	195	2,4,8,9	b4 = 2.0, b2 = 1.5

Importance fcn   I = I1 + I2 + I3

```
I1 = a1(t-tsp)**b1 * N**b2
I2 = a2*(Nc)**b3 * ((tc-tsa)/(tc-tsa-t+tsp+a6))**b4
     * u(t-tsp)u(tsp+tc-tsa-t)    u is a step fcn
I3 = a3*(t-tsp)**b5
```

a1 = 0.003 a6 = 1.0
a2 = 0.005
a3 = 1.0

b1 = 1.0 b4 = 1.3
b2 = 2.0 b5 = 1.0
b3 = 3.0

7. The Computer Simulation

In order to evaluate the performance of Janet, a computer simulation was written in Fortran for a Sun 3/50 microcomputer.

On each time step, the program reads the aircraft data and places aircraft into an array representing the queue. Then the program calculates the importance for each aircraft, chooses one with greatest value, and removes the corresponding aircraft from the queue. The program can simulate either a first-come-first-served model or the Janet model.

To compare the performance of Janet with first-come-first-served, the program also calculates at each time step the number of people on the ground (the number of wasted passenger-hours), the number of connections missed, and the number of people who missed their connections.

Five runs of Janet and one run of first-come-first-served were tested on a sample data set. The results are in Table 1.

On the first run of Janet, 291 people missed their connections, and 5 flights were missed. On the other hand, wasted passenger-hours went down from 1102.5 to 877.5. But when b_4 was increased to 2, the number of planes missing connections went down and the passenger-hours slightly increased.

Notice what happened in runs 4 and 5. When b_4 increased, the importance of the flight time increased. Aircraft 2 and 3 have the same layover time between connections; but aircraft 2 has 400 people, and aircraft 3 only 150. Since $b_3 = 1$, aircraft 3—despite having 100 people needing to make connections—was unable to precede aircraft 2.

We conclude from the experimental runs that our model works better than first-come-first-served, but careful attention must be given to the constants. If a real airport were to use this model, or even if more computer runs were desired, the relative importance of the number of passenger-hours wasted versus the number of people who missed connections must be taken into account. A self-optimizing program could be developed and tested on a series of random data sets in order to determine optimal values for the constants.

8. Strengths and Weaknesses

We have shown how the Janet model is effective on a sample data set using a computer simulation. The model was tested in the framework of an ideal airport but should be easily adaptable to real cases.

Since the only experience any of us have with airports is flying in a plane, we do not have any verified knowledge as to soundness of our assumptions. We did show, however, that adjusting the parameters can account for layover time. Also, we do not suggest that the importance functions I_n are absolute, but rather show how functions like these can simulate the response we desire in the model. Given the extra factors, extra functions can be added as well.

We did not see any reasonable way to treat the model in an analytical form, to show how the function I could be maximized for all planes while minimizing wasted passenger-hours and missed connections. Nevertheless, airports have easy access to computers; and these computations require only a few seconds. The parameters could be optimized using real airline data.

In conclusion, in a computer simulation the Janet queue-sorting model outperformed the first-come-first-served model in satisfying both customers and passenger. Janet is flexible enough for a variety of airports to use, is simple enough to program, and yields to improvement in the form of a self-optimization approach. The model has not been tested on real data, and the ideal airport assumptions may prove inadequate for some cases; but the model does not rely so heavily on the airport assumptions and thus should be adaptable to most situations.