Developing Improved Algorithms for QuickPass Systems

Moorea L. Brega
Alejandro L. Cantarero
Corry L. Lee
University of Colorado at Boulder
Boulder, CO

Advisor: Bengt Fornberg

Summary

We model the arrivals at a "main attraction" of an amusement park by a Poisson process with constant rate; in a more advanced model, we vary the arrival rates throughout the day. The park is open 10 h/day, with a "peak arrival time" between 2.5 and 6 h of the park opening.

We model how a group arriving at the attraction decides whether to enter the normal queue or to obtain a QuickPass (QP)—a pass to return later for a shorter wait. Their decision is governed by their desire to ride, the length of the normal queue, and the return time for the QP.

We explore several models for assigning QPs. The basic model, which gives absolute priority to the QP line, is problematic, since it can bring the normal line to a halt. Our more advanced models avoid this problem by using a "boarding ratio"—either fixed or dynamically varying—for how many from each line to load onto the ride. We use polynomial regression to predict the behavior of the queues, and we determine a dynamic ratio that minimizes the wait times when the number of QPs that can be assigned per time interval is fixed. Finally, we combine these algorithms and determine dynamically both the boarding ratio and the number of QPs to issue throughout the day.

Our advanced models tend to be extrememly robust to small pertubations in starting parameters, as well as to moderate variations in the number of arrivals. We avoid the problems attributed to the current QP system, as long as the ride is not "slammed' with substantially more guests than its capacity. In addition, our system cannot print shorter return times than have previously been issued. Averaging over a two-month period, the total wait times, the

The UMAP Journal 25 (3) (2004) 319–336.

number of people in each queue when the park closes, and the number of QPs issued are consistent across all models.

Introduction

In amusement parks, guests spend a great deal of time waiting in line, especially for popular rides. To reduce the time in line and hence increase overall enjoyment, a "QuickPass" system has been implemented at various locations. Rather than waiting in the normal queue for a ride, a guest can choose to enter a virtual queue during a one-hour time window later in the day.

Our model takes into account various factors, including the length of the normal queue, the number of people with a QP for the ride, and the percentage of the ride capacity that the QPs can commandeer.

Disney's "FastPass" System

We base some of our system design on the "FastPass" system implemented by Disney Theme parks for their most popular rides. A guest who approaches a ride sees the projected wait time in the normal queue, as well as the current FastPass time window; if the guest chooses a FastPass, the system prints a ticket for that time window, which tells the guest when they can enter the FastPass line. Wait times in the FastPass queue tend to range from 5 to 10 min [R.Y.I. Enterprises 2004]. FastPasses are set to commandeer 40–90% of the given ride's capacity [Jayne 2003]. A guest is allowed to get a FastPass every 45 min to 2 h, depending on how busy the park is. At popular attractions, FastPasses are often sold out before noon on busy days [Jayne 2003].

Simplifying Assumptions

- At all times, we know the number of people in the amusement park (determined using turnstiles to count the entries and departures).
- At all times, we know the number of people in both the normal and the QP lines.
- Groups arriving together act together (e.g., all wait in the normal line or all obtain QPs).
- People who obtain QPs always return during their allotted time and enter the QP queue.

Queue Flows and Wait Times

The flow rates (people/min) for the queues are determined as follows:

$$f_{\text{in}}^{\text{NL}} = \frac{\sum_{i=1}^{L} N_{\text{arrival}}^{\text{NL}}(t+i\Delta t)}{\Delta t_{\text{flow}}}, \qquad f_{out}^{\text{NL}} = \frac{\sum_{i=1}^{L} N_{\text{exit}}^{\text{NL}}(t+i\Delta t)}{\Delta t_{\text{flow}}},$$

$$f_{\text{in}}^{\text{QP}} = \frac{\sum_{i=1}^{L} N_{\text{arrival}}^{\text{QP}}(t+i\Delta t)}{\Delta t_{\text{flow}}}, \qquad f_{out}^{\text{QP}} = \frac{\sum_{i=1}^{L} N_{\text{exit}}^{\text{QP}}(t+i\Delta t)}{\Delta t_{\text{flow}}},$$

where

- $N_{\text{arrival}}^{\text{NL}}(t)$ is the number of people entering the normal queue at time t,
- $N_{\text{arrival}}^{\text{QP}}(t)$ is the number of people entering the QP queue at time t,
- $N_{\text{exit}}(t)$ is the number of people leaving each queue at time t,
- L is a fixed constant defining the size of the interval over which we wish to compute the flow, and
- Δt_{flow} is the total time over which the sum is computed, $L\Delta t - t$.

Note that L can vary over the day and the spacing Δt is not necessarily uniform.

We use the flow rates to estimate the waiting times in the two queues. Because the flow rates can change suddenly, we use linear regression on the previous two flow values and the current flow value to help smooth the data.

Basic Model

We begin with a basic model describing the important aspects of the "primary attraction." We are interested in:

- the frequency of arrivals at the attraction,
- the number of people in each group that arrives at the attraction,
- the lengths of the normal and QP queues when the group arrives,
- how many groups obtain a QP, and
- the current state of the QP system (e.g., can it assign any more QPs today, or is it "sold out").

Figure 1 is a flow-chart of the logic in the basic model. We first simulate the number of groups arriving at the attraction using a Poisson process with constant rate λ.

We use a Poisson random variable with a mean of 3.6 to simulate the size of a group; the sampled value is added to the minimum group size of 1 person,

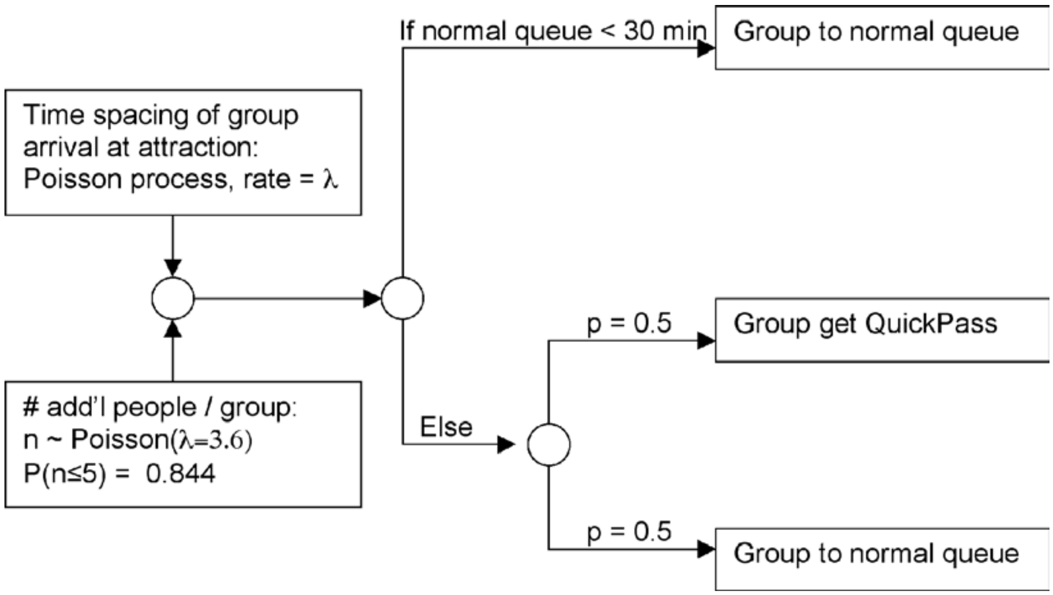


Figure 1. Flowchart for the primary processes in the basic model.

resulting in a mean group size of 4.6 people. This choice of rate gives 84% probability that groups have between 1 and 6 people.

If the wait-time of the normal queue is less than 30 min, the group enters the normal queue; if it is longer, they get a QP 50% of the time (unless the QP system is sold out, in which case the group enters the normal queue).

The return time for the QP is $\max\left(t_{\text{NL}}, t_{\text{syst}}\right)$, where t_{NL} is the predicted wait-time in the normal queue (based on the current number of people in both the queues and the ride capacity) and t_{syst} is the internal time of the QP system. The internal time is determined as follows:

1. When the QP system first turns on, the system time (and the start time for the first QP issued) is set to $t_C + t_{\text{NL}}$, where t_C is the current clock time (say, 1 h after the park opens).

2. From this point on, each time someone arrives at the attraction, we check:

   (a) If the QP system has reached the maximum number of QPs issued for a given start time, we increment the system time by a fixed value (e.g., 5 min).

   (b) If $t_C + t_{\text{NL}} \leq t_{\text{syst}}$, we issue a QP with $t_{\text{start}} = t_{\text{syst}}$; otherwise

   (c) if $t_{\text{syst}} < t_C + t_{\text{NL}}$ then we issue a QP with $t_{\text{start}} = t_C + t_{\text{NL}}$, and update the system time to $t_{\text{syst}} = t_C + t_{\text{NL}}$.

3. Once $t_{\text{syst}} \geq T - 1.25$ h, where T is the length of time that the park is open, we no longer issue QPs.

This system avoids the problem of the current system, where if the length of the normal line fluctuates drastically, a QP can be assigned for a time, say

4 h away, and a short while later for a time only 1 h away. By resetting the system time to $t_C + t_{\text{NL}}$ if this number is greater than the current system time, we guarantee that subsequent QPs always print a start time later than (or the same as) previous QPs.

The QP is issued to each guest with a time window from the specified t_{start} to that time plus one hour. We assume that all guests who obtain a QP return during their allotted time; their return is simulated using a uniform distribution over the hour for which the ticket is valid.

Once the current group enters the normal queue or obtains a QP, we check to see if any of the attraction's cars have left since the last group arrived and update the number of people in each queue.

Improvements on the Basic Model

An Improved Decision Algorithm

We now include a decision algorithm that enables a group to make a choice based on three factors:

- their desire to ride the main attraction ($d \in [0, 1]$),
- the length of the normal queue (L_{NL}), and
- the return-time for the QP (L_{QP}).

We define

$$N \equiv L_{\text{NL}} - \mu_{\text{NL}}, \qquad Q \equiv 0.3\, L_{\text{QP}} - \mu_{\text{QP}}, \tag{1}$$

where

$$\mu_{\text{NL}} \equiv \min\bigl(f(d), L_{\text{NL}}\bigr), \qquad \mu_{\text{QP}} \equiv \min\bigl(f(1-d), L_{\text{QP}}\bigr).$$

The function $f(d)$ translates the group's desire to ride the attraction into the length of time that they are willing to wait in line. The function $f(1-d)$ determines how much later in the day the group would be willing to return for the QP queue. Using $0.3\, L_{\text{QP}}$ instead of L_{QP} takes into account that people are more willing to wait in a "virtual" queue (where they can spend time riding other rides) than physically in line.

We define f as a quadratic function passing through the points $(0, 0)$, $(1, T)$, and $(-1, T)$, where T is the length of time that the park is open. Thus, a person with zero desire would be willing to wait for no time, and a desire level of 1 would indicate that the group would wait all day for this ride. As a more realistic example, a desire level $d = 0.6$ indicates that the group is willing to wait in the normal line for 3.6 h; they would prefer the QP option if the return time were less than 0.48 h.

The value for d, the group's level of desire to ride, is taken from a normal distribution $N(\mu = 0.5, \sigma = 0.1938)$, with 99.2% of its area contained in $[0, 1]$. We compute N and Q from **(1)**. The quantity with the minimum value determines whether the group enters the normal line or opts for the QP.

Arrival Time at the Attraction

Amusement-park-goers know that the best times to ride the big attractions are early in the morning and late in the evening, when there are fewer people and the queues are shortest. In the basic model, we simulate the interarrival time between groups as an exponential random variable with constant rate λ. In a more realistic model, λ should vary with the time of day—people arrive less frequently near the beginning and near the end of the day. We define this rate for the Poisson process as the piecewise continuous function:

$$\lambda = \begin{cases} \left(\dfrac{\dfrac{1}{M} - \dfrac{1}{m}}{a} \right) t + \dfrac{1}{m_B}, & 0 \le t < a; \\[2ex] \dfrac{1}{M}, & a \le t < b; \\[2ex] \left(\dfrac{\dfrac{1}{M} - \dfrac{1}{m_E}}{b - T} \right) t - \left(\dfrac{\dfrac{1}{M} - \dfrac{1}{m_E}}{b - T} \right) b + \dfrac{1}{M}, & b \le t < T - c; \\[2ex] \dfrac{1}{m_E}, & T - c \le t < T, \end{cases} \tag{2}$$

where

- M is the expected number of groups per minute at the peak of the day,
- m_B is the expected number of groups per minute at the start of the day,
- m_E is the expected number at the end of the day,
- a and b are the beginning and end of the peak arrival time,
- c is the time at which the arrival rate assumes a constant value of $1/m_E$, and
- T is the number of hours that the park is open.

In our standard simulation, we take $M = 5$, $m_B = 1.5$, $m_E = 0.1$, $a = 2.5$ h, $b = 6$ h, $c = 1$ h, and $T = 10$ h.

Dynamically-Calculated Number of QuickPasses

In the simple model, the wait time for the QP queue is kept low by placing all QP guests on the ride before anyone from the normal queue. As a result, the normal queue may build up to a wait time of several hours. A better system is to fix α, the ratio of the number of people allowed to board from the normal queue to the capacity of the ride; $\alpha = 0$ reverts to the simple model of boarding all QP guests before anyone from the normal queue.

To ensure that the wait time of the normal line remains reasonable, we need an educated guess of how many QPs to give out for future time intervals. **Figure 2** shows the logic. The machine uses past and current information about the flow into and out of each queue to determine a projected wait time. The number of QPs for a time interval is determined by trying to keep the wait times of the queue below 20 min (QP) and 2.5 h (normal).

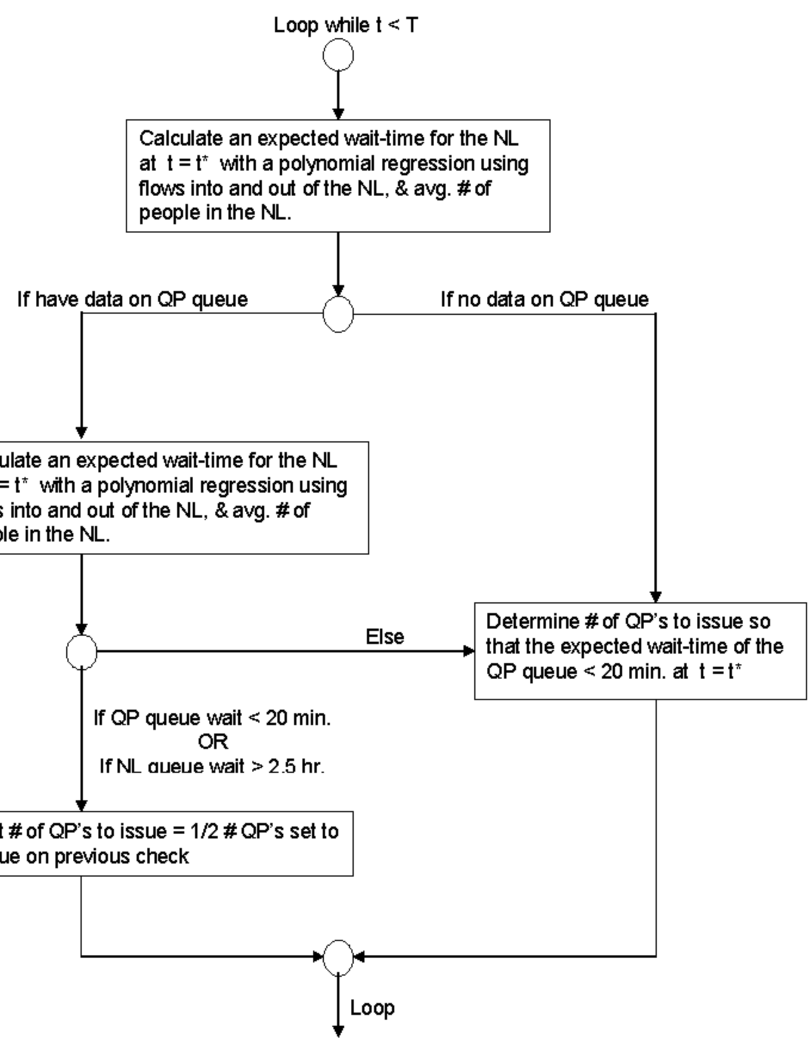


Figure 2. A flowchart of the QuickPass system that dynamically varies the number of QuickPasses offered for any one time interval.

Every 15 min, the QP machine calculates the average number of people entering and leaving each line during that time interval. Then, using a polynomial regression through the last three available flow rates, it calculates estimate of the flow rates at 15 min past the start of the QP return time window, $t_f = t_r + 15$ min. We approximate the flow rate from the current time t_c to t_f as the average of the current flow f_c and the flow determined by polynomial regression: $\hat{f} = \frac{1}{2}(f_p + f_c)$. An approximation to the number of people in the queue at time t_f is

$$N_f \approx \bar{N} - \hat{f}_{\text{out}} t_f + \hat{f}_{\text{in}} t_f = \bar{N} - \tfrac{1}{2}(f_{p,\text{out}} + f_{c,\text{out}})t_f + \tfrac{1}{2}(f_{p,\text{in}} + f_{c,\text{in}})t_f, \qquad \textbf{(3)}$$

where $\bar{N}$ is the estimated number of people currently in the queue. The number of people in the queue depends only on the current and projected flow rates for that queue and the estimated number of people currently in the queue. If **(3)** produces a negative number (indicating an unrealistic approximation of the flow rate), we use instead the current flow rate,

$$N_f = \bar{N}\,\frac{f_{c,\text{in}}}{f_{c,\text{out}}}.$$

The projected wait time t_{NL} for the normal line is given by the projected number of people in the queue, N_f^{NL}, divided by the estimated future outflow rate:

$$t_{\text{NL}} = \frac{N_f^{\text{NL}}}{f_{p,\text{out}}^{\text{NL}}}.$$

The wait time for the QP queue is computed in a similar manner. If the wait times for the normal queue and the QP queue are below their maximum acceptable wait times and the flow for the QP line is zero, the number of QPs issued is determined using

$$n = 4(1-\alpha) f_{\text{out}_{\text{NL}}} \min\left(20 \text{ min},\ \frac{t_{\text{NL}}}{3}\right).$$

If the flow for the QP queue is nonzero, we use

$$n = 4 f_{\text{out}_{\text{QP}}} \min\left(20 \text{ min},\ \frac{t_{\text{NL}}}{3}\right).$$

This calculation is done every 15 min or whenever all the QPs for the time interval have been issued.

Dynamic Ride-Loading Ratio

In the previous model, we fix the number of guests from each queue who can enter the ride and dynamically vary the number of QPs issued for each time interval. Here, we consider the opposite idea: The QP system issues a fixed number of tickets for every time interval, and the parameter α (the number of people from the normal queue divided by the total capacity of the ride) varies. **Figure 3** shows the logic chart for this system.

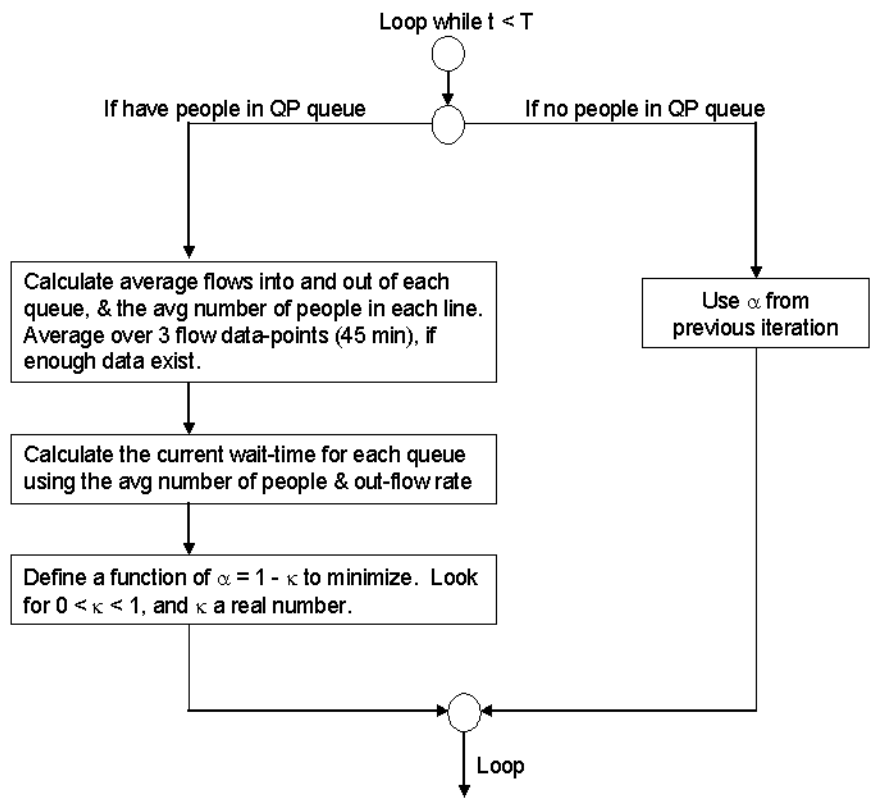


Figure 3. Regulating queue lengths by dynamically varying the ratio of people who board the ride from each queue.

We begin the day with an arbitrary α between 0 and 1. Once the QP queue forms, a new value is calculated by minimizing the dimensionless weighted sum of the wait times for each queue. First, the average wait time for each line is calculated by determining the average outflow rates for both lines and the average number of people in each queue during that time,

$$t_w = \frac{\bar{N}}{\bar{f}_{\text{out}}}.$$

Each waiting time is then normalized by the maximum acceptable waiting time, 20 min for the QP queue and 2.5 h for the normal queue, to create the weighting factors

$$\beta = \frac{t_w^{\text{QP}}}{1200}, \qquad \eta = \frac{t_w^{NP}}{9000},$$

with times in seconds. We then determine the value for $\alpha = 1-\kappa$ that minimizes the dimensionless wait time:

$$\text{dimensionless wait time} = \left(\beta \frac{f_{\text{in,QP}}}{144\kappa^2} + \eta \frac{f_{\text{in,NL}}}{144(1-\kappa)^2} \right)$$

by solving for the real root between 0 and 1 of the cubic polynomial

$$0 = (\mu + \gamma)\kappa^3 - 3\gamma\kappa^2 + 3\gamma\kappa - \gamma,$$

where

$$\gamma = \beta \bar{N}_{\text{QP}} f_{\text{in}}^{\text{QP}}, \qquad \mu = \eta \bar{N}_{\text{NL}} f_{\text{in}}^{\text{NL}}.$$

Dynamic Ps and Boarding Ratio

In this model, we simultaneously vary both the number of QPs per time interval and the boarding ratio α. Hence, we need only ensure that the initial parameter value for α is reasonable. If the initial α is too large, the algorithm will indicate that the system should issue no QPs at all; the algorithm to update α will then continue using the same α because no one has arrived in the QP queue. Hence, we expect this new model to be sensitive to the initial value of α.

Three-Tiered Queueing

The final improvement to the system adds a third queue, the Priority-OnePass (POP) queue, which is to the QP queue as the QP queue is to the regular line. This new queue will be shorter than the QP (now Priority II) queue, with a wait no longer than the arrival time between cars on the ride, but also with a return window of only 15 min that can be booked for as soon as 45 min from the current time. In essence, this new queue allows guests to "make an appointment" to ride the attraction. We assume that everyone who takes a POP or QP returns during their designated time window.

The POP queue has a maximum of 600 tickets for the day: 15 for every 15-min interval throughout the day. However, because the POP machine starts only after the wait time for the regular queue has exceeded half an hour, some POP tickets may not be issued.

The decision algorithm for the queues is relatively basic: If the return times for both queues are about the same, guests take the QP ticket 70% of the time because it allows more flexibility. The other cases are shown in detail in (**Figure 4**). This model can give first return times that are not in chronological order.

Once the group has decided to take a POP, their return time is chosen according to a $\chi^2(2)$ distribution, which favors return times closer to the current time while still allowing for times later in the day.

We did not fully implement this model; in comparing the models, we focus on the previously described systems.

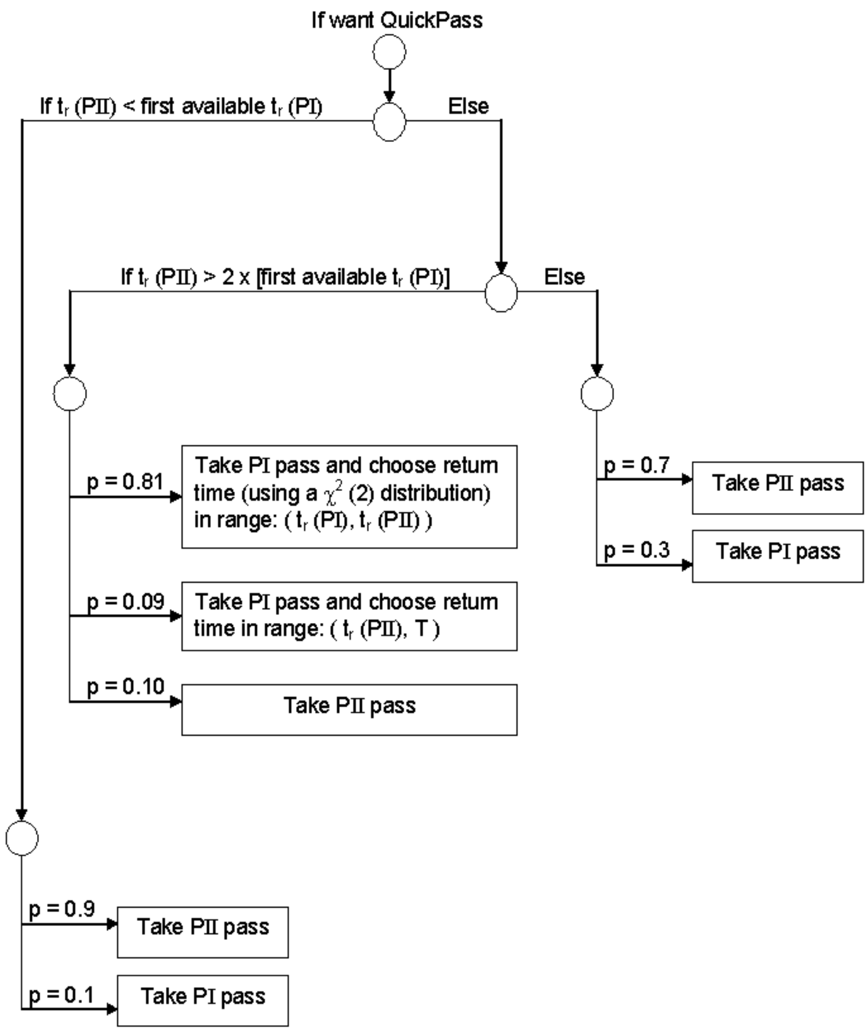


Figure 4. Flowchart of decision algorithm for choosing a QP) or a POP.

Results

Basic Model

The primary parameter in the basic model is λ, the rate for the Poisson process for the arrival of groups; changing λ effectively changes the popularity of the ride.

In **Table 1** we present daily totals for the basic model with a constant inter-arrival rate. Results are shown for various values of λ, with a value of $\beta = 20$ QP tickets before incrementing the internal time by 5 min. The capacity of the ride is 7,200 people/day ($\lambda^{-1} = 23$).

As expected, when the total number of arrivals at the attraction is about 7,200 ($\lambda^{-1} = 23$), the QP system never activates, because we never have a wait-time in the normal queue of longer than 30 min. For both $\lambda^{-1} = 10$ and

Table 1.
Results for the basic model for various values of the parameter λ
(the rate of arrival of groups at the attraction).

λ^{-1}	Guests	QPs	Total riders	QP riders	Normal line riders
10	16,657	2,335	7,200	2,335	4,865
17	10,047	2,362	7,200	2,362	4,838
21	7,863	441	7,157	441	6,716
23	7,371	0	7,025	0	7,025

$\lambda^{-1} = 17$, corresponding to numbers of guests that swamp the system, the basic model issues about the same number of QPs.

In addition to the day-end totals, it is interesting to look at the behavior of the queues (normal and QP), the flow-rates in and out of these queues, estimated waiting times in the queues, the expected wait-time for the ride (the total, at each time interval, of the predicted wait in the physical QP queue and that in the normal queue), start-times issued by the QP system, and the ratio of people choosing the QP queue over the normal queue, all as functions of time. [EDITOR'S NOTE: The authors' complete paper included numerous more graphs and analyses of these features; we cannot include them all here.]

When Demand Is Near Capacity . . .

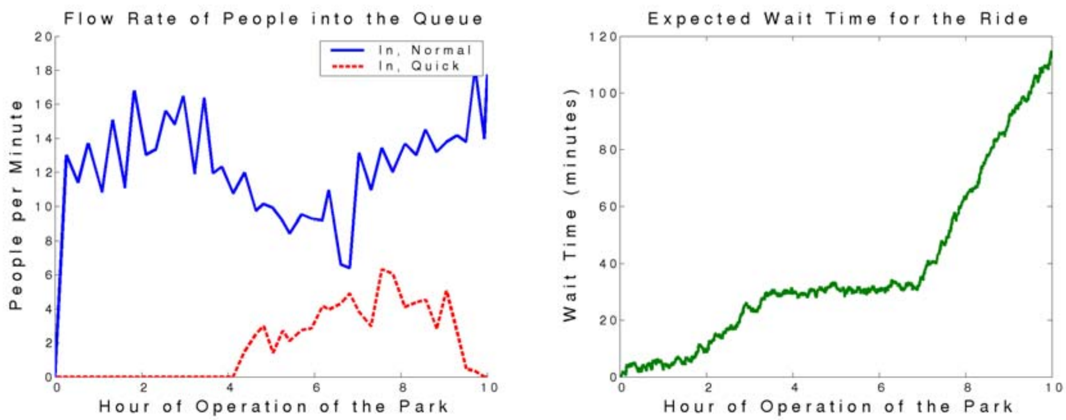
The only parameter of the QP model that can be adjusted is β, the maximum number of QPs that can be issued before the system's time clock (t_{syst}) advances. For example, with $\lambda^{-1} = 20$ (the mean interarrival time between groups is 20 sec), and with $\beta = 20$ QPs that can be issued before incrementing the system time by 5 min, we issue about 1,000 QPs. When we can distribute ten times as many—$\beta = 200$ QPs—before incrementing, we issue only about twice as many QPs; the limited number of people arriving during the 5-minute interval prevents a huge increase in the number of QPs issued (**Figure 5**).

What Happens When Demand Swamps Capacity?

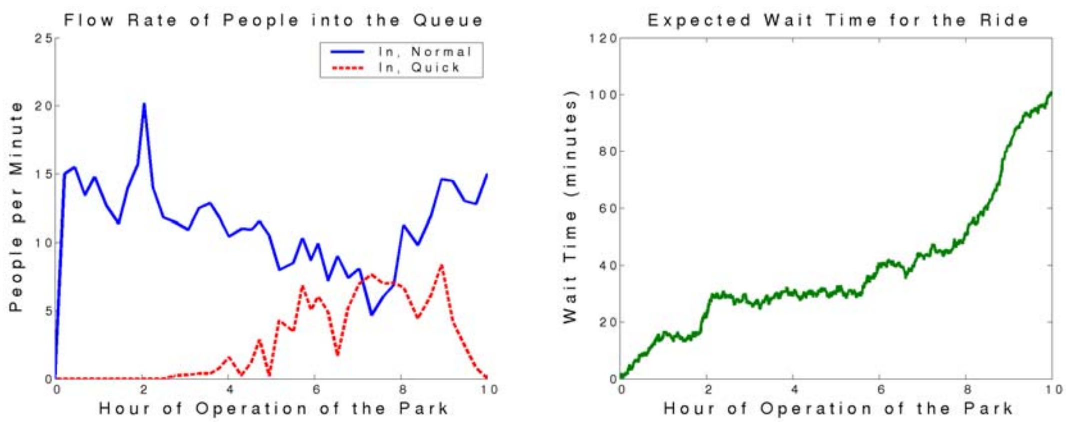
We compare the situation of $\lambda^{-1} = 20$ (the number of guests is approximately the capacity of the ride) with $\lambda^{-1} = 10$ (twice as many guests as the capacity), considering in each instance the cases $\beta = 20$ and $\beta = 200$, the maximum number of QPs issued before the QP clock is advanced by 5 min.

For the $\lambda^{-1} = 20$ cases, the overall wait-time increases linearly from when the park opens up until the QP system goes online, at which point it plateaus; the wait times increase again once the QP system is sold out for the day. The wait-time of the QP queue stays very short (< 4 min).

For $\lambda^{-1} = 10$, guests begin entering the QP queue only an hour after the park opens, and QPs sell out 3 to 5 h after the park opens (5 h is halfway through the day), depending upon how many QPs are allowed.



a. $\lambda^{-1} = 20, \beta = 20.$



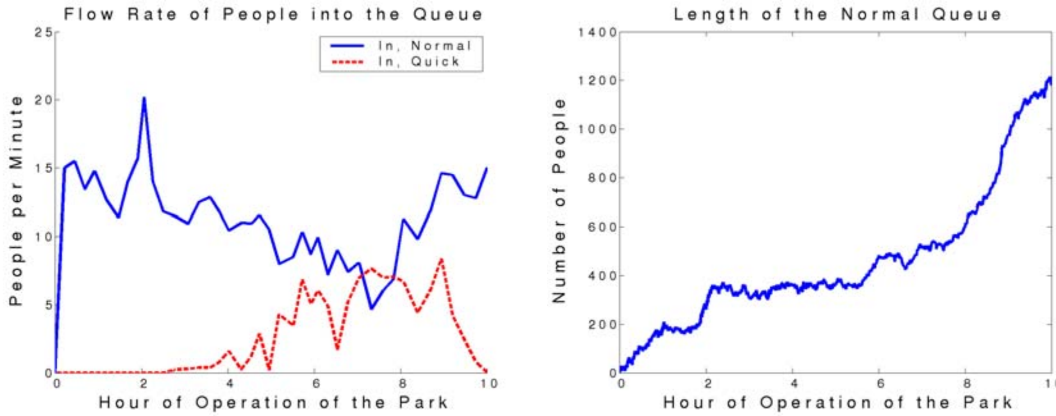
b. $\lambda^{-1} = 20, \beta = 200.$

Figure 5. Basic model: Inflow and expected wait time with $\lambda^{-1} = 20$ (8,000 guests), for two values of β, the number of QPs that can be issued before the QuickPass system increments the start time by 5 min: **a.** $\beta = 20$. **b.** $\beta = 200$.

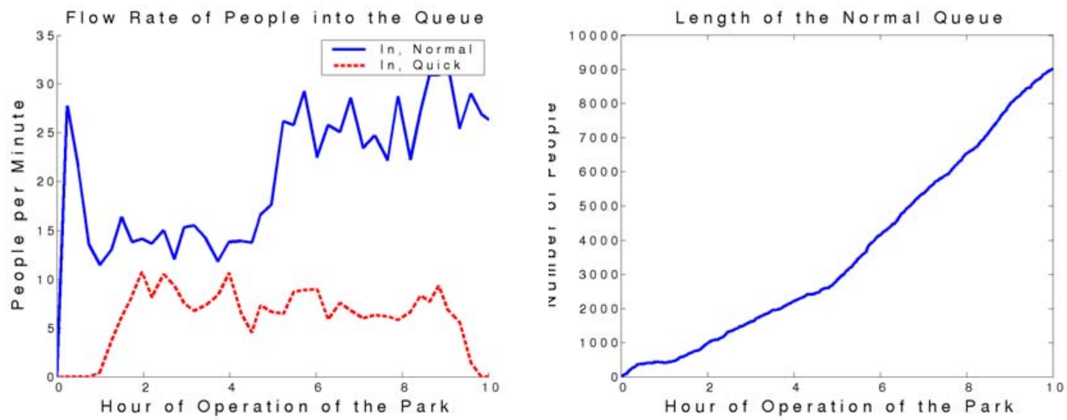
Because the number of total arrivals stays approximately constant throughout the day ($\sim$ 30 people/min), the high priority for QP riders substantially slows down the normal queue (even though when the QP system is active, the flow rate into the normal queue drops to ~ 15 people/minute), increasing the average wait-time for people in the normal line and leaving an enormous number of people in the normal line when the park closes (**Figure 6**). This is clearly not an ideal system.

Improved Model

The basic model has groups take a QP 50% of the time when the normal queue had a predicted wait-time longer than 30 min. In the improved model, the decision of which queue to enter is based on the desire to ride the attraction, the length of the normal queue, and how much later in the day they would return if they took a QP.



a. $\lambda^{-1} = 20$ (8,000 guests), $\beta = 200$ max QPs before time increment.



a. $\lambda^{-1} = 20$ (8,000 guests), $\beta = 200$ max QPs before time increment.

Figure 6. Basic model: Inflow and length of normal queue with $\beta = 200$, for varying arrival intensity. **a.** $\lambda^{-1} = 20$ (8,000 guests). **b.** $\lambda^{-1} = 10$ (16,000 guests).

The behavior for the two decision algorithms is similar when the number of arrivals corresponds reasonably closely with the ride capacity ($\lambda = 1/20$). When arrivals overwhelm capacity, however, the new decision algorithm tends to "flood the queue." The plots in **Figure 7** give the predicted wait-time for the QP queue throughout the day with the new decision model, for two levels of ride demand. Because the number of arrivals is so much greater than the ride capacity when $\lambda^{-1} = 10$, the normal queue has no outflow during the middle of the day and its length grows so long that more and more people take a QP, until the QPs sell out. Even though many more people choose QPs in this situation, the QP queue still empties out nicely by the end of the day.

Basic Model, Variable Interarrival Rate

We revert to the original "dummy" decision algorithm, where people choose the QP 50% of the time when the normal line's wait-time exceeds 30 min and the interarrival spacing is given in **(2)**. We choose a standard set of parameters

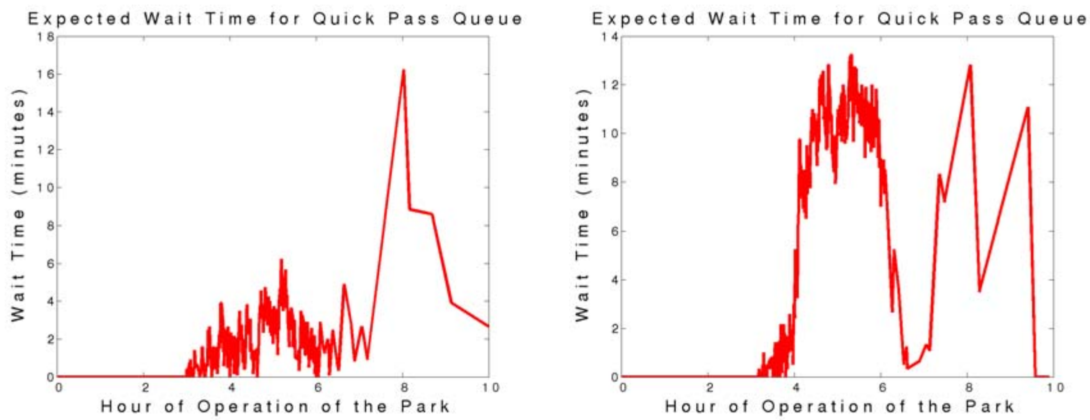


Figure 7. Basic model with improved decision algorithm: Expected wait time in the QuickPass with $\beta = 200$ QPs issued before the QP system increments t_{start} by 5 minutes, for two levels of arrival intensity. **a.** $\lambda^{-1} = 20$. **b.** $\lambda^{-1} = 10$.

that result in a rush of arrivals during the peak hours but that does not cause a total demand greater than the ride's capacity. We define the peak arrival time to be between $2.5 < t < 6$ h after the park opens. The standard parameters that we use are: $M = 5$, $m_B = 1.5$, and $m_E = 0.1$ groups/min, where M is the maximum number of groups/min during the peak-times of the day, m_B is the minimum number as seen at the beginning of the day, and m_E is the minimum as seen at the end of the day. With the interarrival spacing dependent on the time of day (and hence on the occupancy of the park), the normal queue nearly empties by closing time.

Variable Rate and Improved Decision Algorithm

Using both of the improvements to the basic model, and approximately 7,400 people arriving at the attraction, we issue 1,500 QPs if we allow $\beta = 20$ QPs per 5 min and 3,000 if we allow $\beta = 200$ QPs per 5 min. The peak wait-times are respectively 3 h and 2 h.

Dynamically Varying Parameters

[EDITOR'S NOTE: Although the authors presented results on separately varying the boarding fraction α and the number of QPs, space does not permit including those results here.]

Varying Both α and the Number of QuickPasses

The final model allows both the borading ratio and the number of QPs issued per time interval to vary dynamically. **Figure 8** shows the results of this model with our standard set of parameters describing arrival times throughout the day. We begin the day with $\alpha = 0.5$. During the day, this value climbs as high as 0.6. The number of QuickPasses made available per 5 min period rises to 700 slightly more than 2 h after the ride opens, before declining to 0 about 2 h later. Although the expected wait time in the QP queue has a maximum at a little over 1 h, wait times in the normal queue reach 4 h, with the queue more than 1 h long at closing time.

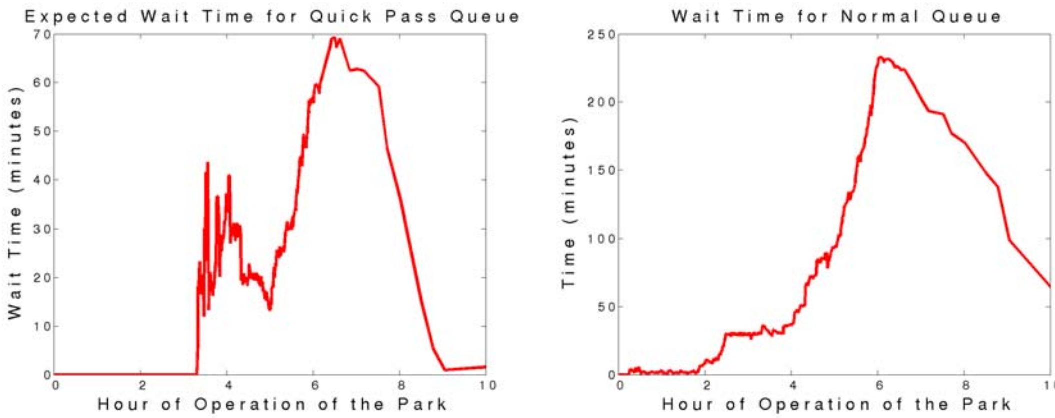


Figure 8. Expected wait times in the queues with dynamically varying α (proportion of people boarding from the normal queue) and β (the number of QuickPasses issued before incrementing the system time), for an initial $\alpha = 0.5$.

Statistical Analysis

We ran two-month trials for each QP system and summarized the overall performance of each QP system in terms of mean and standard deviation of various quantities. The average hourly wait-times are very similar for the four models, but with a larger variance (by as much as a factor of two) for the models that vary the number of QPs issued. Those tend to result in fewer people remaining in the queues when the ride shuts down for the day; they also tend to issue fewer QPs. The total and maximum queue wait-times are surprisingly consistent across the four models.

Strengths and Weaknesses of the Models

Strengths

- Our models are fairly robust to changes in parameters, including the two most important parameters, the boarding ratio and the number of QPs to issue.
- Our QP system cannot move "backward" in time. For example, it will not

print out a QP for four hours in the future and then a half hour later print a ticket for one hour in the future.

Weaknesses

- All the models rely on flow data, which can vary rapidly over short time intervals. Thuse, the wait times for the two queues can change rapidly, even when we use linear regression to estimate better the average flows. Because we use flows to determine wait time, the average wait time is only a rough approximation of the actual average; to obtain a better sense of the average, we would need to follow *individuals* through each queue and determine exactly how long each guest waits for the ride.
- The models assume that everyone who obtains a QP returns during their allotted window and that everyone in line stays in line until they reach the ride. In reality, some guests with QP tickets miss their window or decide not to return, and some in the normal queue get frustrated with the wait and leave.
- In addition, our models look at only a single ride. If several rides have a QP system, all ride systems must interact to determine how many QPs to give out for each ride in a single interval. A more complex model would have to take into consideration how people move between rides and how long they are willing to wait based on the lines of other rides in the park.

Conclusion

We model the arrival of groups at an attraction and their decision process when faced with the option of obtaining a QP. We analyze the effect of different versions of a QP system, including dynamically adjusting the system. Our system avoids current problems, such as printing sooner return times than those previously issued.

For all our models, we obtain reasonable behavior when the number of people arriving at the attraction does not greatly exceed its capacity. Averaging behaviour over a two-month period, we find that the total waiting time, the number of people in each queue at the time the park closes, and the number of QPs issued per day are consistent across all our models within their statistical errors. Results of individual days show larger differences when the ride is "slammed."

Finally, we developed but did not implement a more sophisticated algorithm with an additional PriorityOnePass option.

References

Burden, R.L., and J.D. Faires. 2001. *Numerical Analysis*. 7th ed. Belmont, CA: Brooks/Cole, Thomson Learning.

Jayne, A.W. 2003. How much time does Disney's Fast Pass save? `http://members.aol.com/ajaynejr/fastsave.htm`.

R.Y.I. Enterprises, LLC. 2004. Fastpass. `http://allearsnet.com/tp/fastpass.htm`.

Ross, S.M. 2002. *A First Course in Probability*. 6th ed. Upper Saddle River, NJ: Prentice Hall.

________. 2003. *Introduction to Probability Models*. 8th ed. New York: Academic Press.

Strang, G. 1988. *Linear Algebra and its Applications*. 3rd ed. Philadelphia, PA: Harcourt College Publishers.

Tijms, H.C. 1994. *Stochastic Models: An Algorithmic Approach*. New York: John Wiley & Sons.

Wackerly, D.D., W. Mendenhall, and R.L. Scheaffer. 2002. *Mathematical Statistics with Applications*. 6th ed. Duxbury, CA: Duxbury, Thomson Learning.

Werner Technologies. n.d. Disney Fastpass information. `http://www.wdwinfo.com/wdwinfo/fastpass.htm`.

Yakowitz, S.J. 1977. *Computational Probability and Simulation*. Reading, MA: Addison-Wesley Publishing Company.