Contents

Optimized QuickPass System

1 Introduction

The amusement parks are now increasingly popular among people from all age groups all over the world. The sharp booming of park attendance has caused people waiting in long lines before they can enjoy the rides, consequently there are increasing complaints from customers received by executives of amusement park. In order to solve the problem and increase people's enjoyment of the amusement park, a park adopted a “QuickPass” system, which allows visitors to spend the time used to wait in the line for other amusements. However, the system is not perfect, sometimes it is not stable and the lines for people with Quickpasses are nearly as long and slow as the regular lines. Therefore an alternative Algorithms should be offered to improve the quality of service in the park.

We are now faced with following:

1. How to optimize the time “QuickPass” system tells the visitors to return to the ride?
2. How to remain the stability of the performance of the QuickPass?

Our approach is:

1. First we analyzed the relationship between the time for playing, stated number of people of the ride, the number of visitors waiting in line and the number of people who used the “QuickPass” system to build up our initial model to decide the starting time for visitors come back.
2. We then use simulations to find out the several unknown variables in our “QuickPass” system.

2 Assumptions and Hypotheses

- MOST of visitors in the park follow the instruction printed out by our QuickPass system.
- The running time for a specific ride is fixed.
- The time spend on using QuickPass is negligible.
- The amusement park has made a lot of efforts to introduce this QuickPass system to public, and most visitors to the park know the use of QuickPass.
- The ride using our QuickPass never breaks down during a day.
- The time for the first QuickPass user to come back to the ride is less than 60 minutes.
- When start to use the QuickPass, we assume that the ride has entered its rush hour.
- Once the starting time for a QuickPass user exceeded the close time, the QuickPass would not print out the slip.

3 Further consideration

- The QuickPass system, to describe it more simply and clearly, is just a virtual queue which the machine helps you waiting in the line and therefore cannot reduce the time before you can enjoy the ride. However, aided by this system, you can spend the time standing in the line to enjoy other recreation in the park.
- Besides to increase people's enjoyment of the amusement park, the QuickPass system should also help the park to disperse the flow of visitors to different attractions of the park, which can improve the running efficiency of various of rides and keep visitors staying in the park as long as possible, so as to increase the income of the park.
- The QuickPass should follow the rule that the earlier you come, the earlier you get onto the ride, which shows the equality of all visitors.

4 New facilities and new rules

In order to improve the performance of QuickPass system, we then developed several new facilities and new rules associated with our enhanced QuickPass.

4.1 New facilities

- First, we will build a big clock, connected with our QuickPass, which shows the estimated time for the people currently standing in the line, as well as those who used QuickPass to finish riding, in front of the ride which adopted the QuickPass system.
- In our model with feedbacks, in order to count the number of people who do not want to use the QuickPass currently standing in the regular line, we suggest the park to build facilities (we call it “counter”) such as laser counters and so on to count the people getting into the line and out of the line, so as to help calculating the exact number of standing in the line and send this information to the QuickPass

4.2 New rules

- The visitor who follows the instructions printed out by our QuickPass and return to the ride on time has the priority of playing on the ride.
- There should be two lines, one is for people who did not use the QuickPass, and the other (we call it “QuickPass line”) is for those who hold slip printed out by our system.
- The visitors standing in QuickPass line has the priority to get on the ride.
- ONLY during the time interval stated on slip printed out by QuickPass, the one hold the slip can enter the QuickPass line.
- If the visitor cannot make it to return to the ride on time, his/her slip which ensures his/her priority would expire, and at this moment the QuickPass becomes available to him/her again. So now the visitor can choose to stand in the regular line, use the QuickPass again or just leave the ride.

5 Model with feedbacks

To achieve the goal that optimizes the visitor's dispersion in the park and simplify the problem, we first build a model that needs the feedbacks from the "counter" we mentioned above, to aid the regular performance of our QuickPass.

5.1 Symbols used in this model

T: The running time of the ride (Divided into rush hours and regular hours).

τ_n: The starting time for n *th* QuickPass user to return to the ride.

ζ_0: The average value of adjustment to optimize the waiting time.

ζ_1: A parameter which adjusts the starting time for the first QuickPass users.

$\zeta(n)$: A adjust parameter which helps optimize the waiting time for the n *th* users.

$\lambda_1(n)$: The number of visitors has enjoyed the ride in the regular line before the n *th* QuickPass user.

$\lambda_2(n)$: The total number of visitors has entered the regular line before the n *th* QuickPass user.

$\vartriangle\lambda_2(n)$: The number of visitors enters the regular line between the nth and $(n+1)$ *th* user of the QuickPass.

N_0: The stated number of passengers of the ride.

μ: The allowed time interval for QuickPass users to return to the ride.

$\bar{t}_w$: The average waiting time for our QuickPass users.

t_{wi}: The waiting time in the QuickPass line for the i *th* QuickPass User.

5.2 The starting time for visitors to come back

Follows the "First in First out" rule, and considers the virtual queuing essence of QuickPass, we can calculate out our starting time for the current user by using parameters including the current length of the regular line, the running time and stated number of passengers of the ride. Here we use an iterative equation to find out the starting time for our current user to come back for the ride.

$$\tau_1 = \left[\frac{\lambda_2(1)-\lambda_1(1)}{N_0}\right]T - \zeta_1; \text{--} ①$$

And $\zeta_1 = \left[\left(\left\lceil \frac{\lambda_2(1) - \lambda_1(1)}{N_0} \right\rceil \right) N_0 - \left(\lambda_2(1) - \lambda_1(1) \right) + 1 \right] \zeta_0$;

$\tau_{n+1} = \tau_n + \frac{1 + \vartriangle\lambda_2(n+1)}{N_0} T - \zeta(n+1); (n \geq 1 \,\&\, n \in N)$; ----------------②

And $\zeta(n+1) = \left(1 + \Delta\lambda_2(n+1)\right)\zeta_0$;

Here the formula ① represents the starting time for the first QuickPass user during a day, Using the number of visitors ($\lambda_2(1) - \lambda_1(1)$)currently standing in the line ,both of which can be found out by "Counter" (we mentioned above). We then can easily calculate out the time for these visitors finish riding ($\left\lceil \frac{\lambda_2(1) - \lambda_1(1)}{N_0} \right\rceil T$) once we know the running time (T) and stated number of passengers (N_0) of the ride. In order to reduce the time spent on waiting in our "QuickPass line" and the occupancy factor of the ride, we add a parameter (ζ_1) to adjust the starting time.

Once we know the starting time for the first QuickPass user, the starting time for the second user can be easily calculated by adding the time for visitors come into the regular line ($\frac{1 + \vartriangle\lambda_2(1+1)}{N_0} T$) between them, and the condition is same for the third, fourth and so forth. So we write out the iterative formula ② above, and $\zeta(n+1)$ works as same as ζ_1.

To ensure that the starting time for latter system user would not ahead of the earlier users', we can assert $\zeta_0 < \frac{T}{N_0}$.

The value of ζ_1, which highly influences the occupancy factor of the ride, we find out its value by calculating the number of remaining people in the initial queue who suppose to ride with the first QuickPass user, and hence the remaining empty seats on the ride. And finally multiply the adjust parameter ζ_0 obtained in our simulation

The value of $\zeta(n+1)$, closely associated with the number of visitors came to the regular line between the time interval for two operations of the QuickPass, can be expressed as product of its value and our adjustment parameters.

5.3 The allowed time interval for return to the ride

To decide the allowed time interval, we consider it is closely associated with the running time of the ride. Here we expect that the allowed time interval is equal to the running time of the ride, because if let the allowed time interval be longer than the running time, the user may wait more than one run of the ride, then of course, we can ask him/her to come back later. However, if the running time is very short, then it is impractical to ask the visitor to come back during a very short time period. Then we assume the average expected time for waiting is 15 minutes, which is also easy for visitor to come back on time, we wrote following equation:

$$\mu = \begin{cases} 15; T < 15 \\ T; T >= 15 \end{cases} \text{---③}$$

5.4 The average waiting time for QuickPass users

To evaluate the performance of the model, we use $\bar{t}_w$ as our judging criteria.

$$\overline{t_w} = \frac{\sum_{i=1}^{n} t_{wi}}{n}; \text{--④}$$

The formula above represents the average waiting time in the QuickPass line is equal to the total waiting time in the QuickPass line divides the total number of visitors entered the QuickPass line. We used this formula to calculate the waiting time in our simulation codes.

5.5 The performance of our model

To test the performance of our model, we made series of simulation with our programmed codes.
In our simulation codes, we use the running time, stated number of passengers of the ride, and the number of people already in the regular line when the first visitor uses the QuickPass as input variables, to find out the average waiting time for QuickPass users.

To justify the stability of our model, we made simulations to test the independency of our input variables and drew our outputs as below.

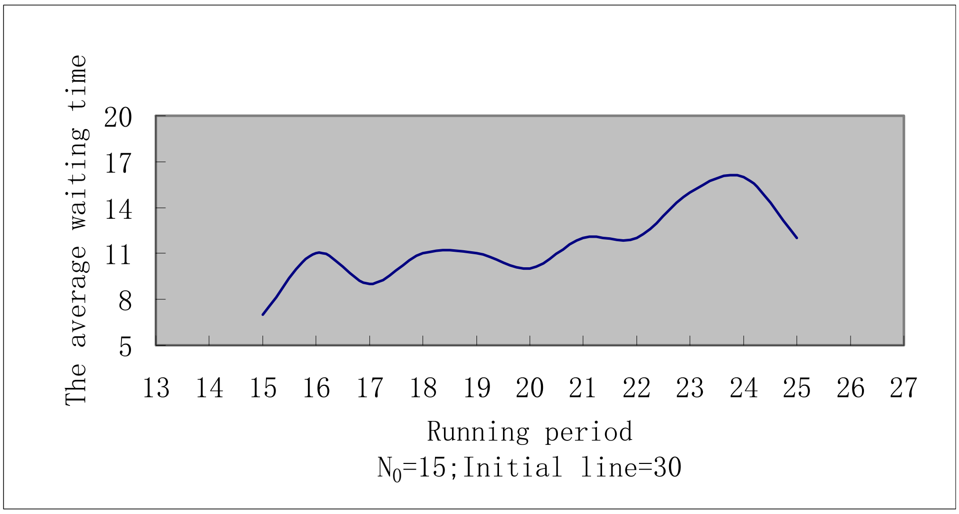


Figure 1 The relationship between running period and the average waiting time with fixed stated number of passengers and initial length of the regular line.

From the chart above, we can see the waiting time for our QuickPass users is less than fifty percent of the running time of the ride when the stated number of passengers is fixed and the running period is a variable.

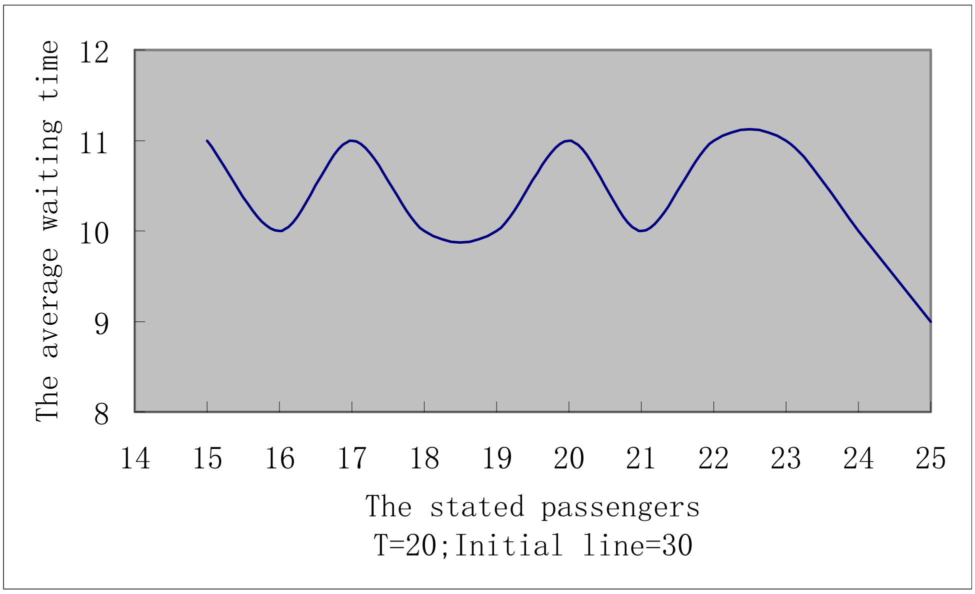


Figure 2 The relationship between the stated passengers and the average waiting time with fixed running period and initial length of the regular line.

The figure above also shows with a fixed running time of the ride, the waiting times are satisfactory as the stated passengers vary.

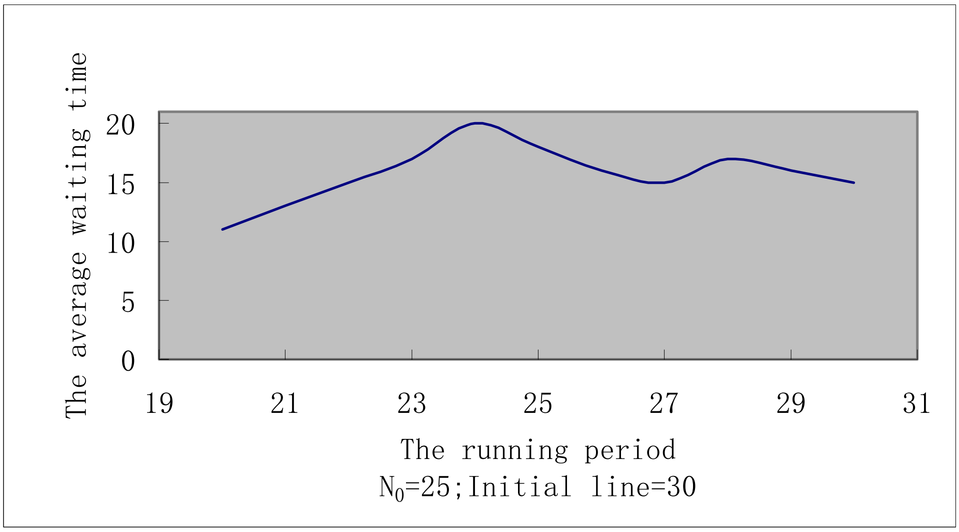


Figure 3 The relationship between running time and average waiting time with fixed stated number of passengers and initial length of the regular line

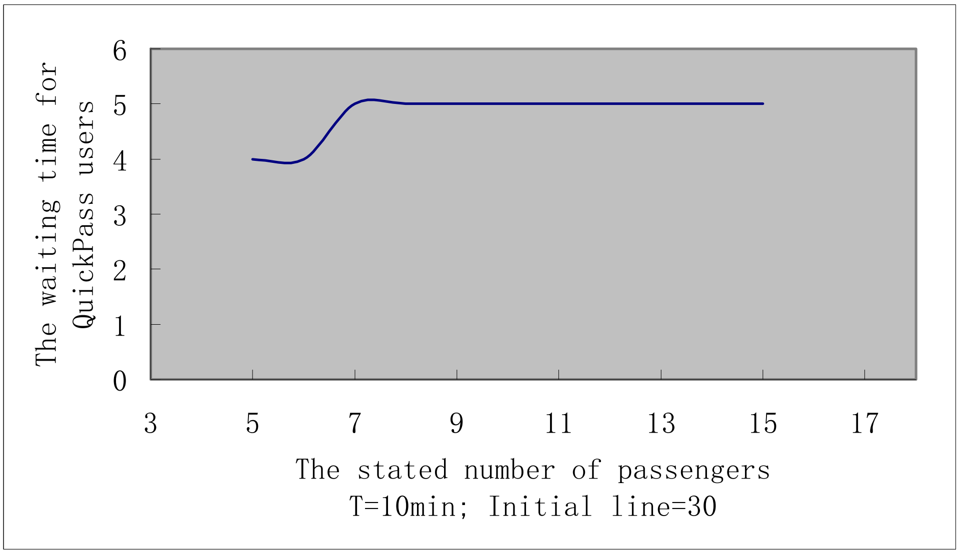


Figure 4 The relationship between stated number of passengers and the waiting time with fixed running time and initial length of the regular line.

To further strengthen the stability of our model, we used other sets of input variables to simulate our model. The results above are still acceptable and therefore we now can assert our model is **STABLE** and can be put into real practice.

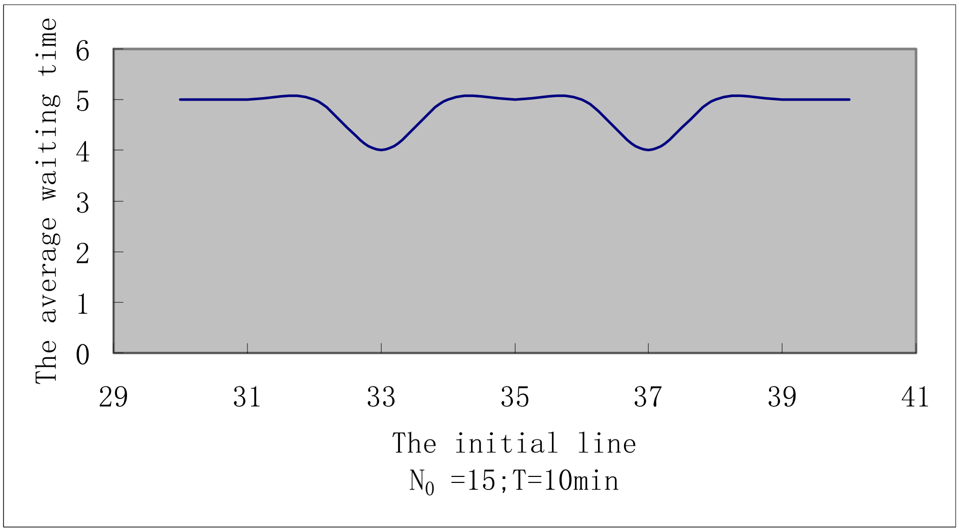


Figure 5 The relationship between the length of initial line and average waiting time with fixed stated number of passengers and running time.

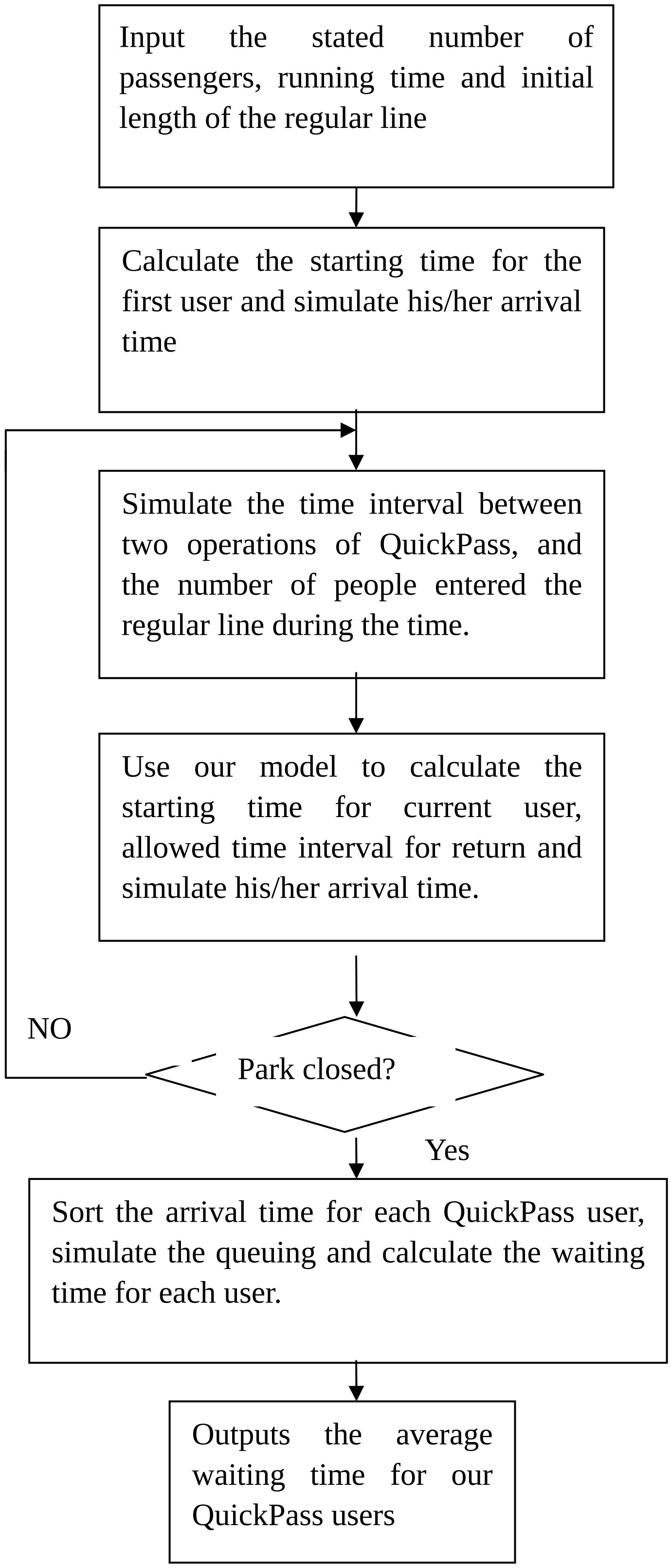
Finally, we tested the sensitivity of our model with the initial length before the first user of our QuickPass system. Again, our model's outputs work very well within our mentioned assumption, which the time for initial line to finish riding is below 60 minutes.

6 Stability and Sensitivity Analysis

Combine the results of our model's simulations, we are CONFIDENT that our model is suitable for different kind of rides in an amusement park. No matter the values of the running time and stated number of the passengers of ride, our model can ensure the users of our QuickPass would only wait in the line for a considerable shorter time and consequently can improve their enjoyment of the park. Therefore, if most of the visitors in the park follow the rules we set, both visitors and the park would benefit from newly developed QuickPass system.

7 Simulation Process

In order to illustrate our simulation procedure, we drew a flowing chart of our simulation as below.



With input variables mentioned above, considering the law of possibility, we generate random number to represent the time interval between two operations of the QuickPass and number of people entered the regular line between it. Moreover using parameters including the starting time, allowed time interval to return, we generate the arrival time for all QuickPass users and therefore can easily calculate the waiting time for each individual with running time and stated number of passengers. With all these data, we can output the average waiting time of our QuickPass users.

8 Further possible modification of the model

In our feedback system, we need other accessories such as "counter" to ensure the proper performance of our model, which may increase the running cost of the park. Therefore secondly, we try to build a non-feedback system which using parameters such as the historical attendance of the ride during the day, the frequency of the use of the QuickPass and so on to predict the current length of waiting line.

9 Strengths and weaknesses

9.1 Our advantages

- We make it clear that the average waiting time in the "QuickPass Line" is the most critical judging criteria for all alternative systems.
- We put great emphasis on the equality of all visitors, no mater whether he/she used the QuickPass, in most cases, the earlier he/she comes, the earlier he can enjoy the ride.
- In the simulation, our system is quite stable and easy to use, so we expect it can put into daily use with few modifications.

9.2 Our weaknesses

- We obtained several parameters with our assumptions and simulations, more specific data are needed to modify our model.

10 Non-technical summary for park executives

The most fatal deficiency of currently used QuickPass system is unstable, and therefore treats different users unequally. To modify this problem, we adopted the "First in, First out" rule for newly developed system, which means we can ensure that the earlier a visitor comes, the earlier he/she can get onto the ride.

To further relieve the pressure of the popular rides during the rush hours, we suggest build a big clock, in front of each ride which adopted our QuickPass system, shows the estimated waiting time for the ride, which gives visitors a direct figure about how long he/she has to wait in order to enjoy the ride, consequently helps them decide whether use the QuickPass or just wait in the line. In some sense, the more convenience you give to your visitors, the more you would profit from them.

Moreover, our QuickPass system can keep the occupancy factor of the ride in fairly high level, in another words, the running cost is lowered.

11 References

[1].www.scottware.com.au/theme;Accessed on Feb. 6th 2004;

[2].http://alwayswholesaletravel.com/california/mkt_insider_fp.html; Accessed on Feb. 6th 2004

3.http://www.mouseplanet.com/dtp/dlguide/6_Parks/general/fastpass.htm; Accessed on Feb. 6th 2004

4.http://www.mouseplanet.com; Accessed on Feb. 6th 2004

5.http://www.dlp-guide.com/HintsInfos/Fastpass.htm; Accessed on Feb. 6th 2004

6.http://www.intercotwest.com; Accessed on Feb. 7th 2004

7.http://www.laughingplace.com; Accessed on Feb. 7th 2004

8. Mathematics and Operations Research in Industry Dennis E. Blumenfeld, Debra A. Elkins, and Jeffrey M. Alden

9. http://www.kingdommagictravel.com; Accessed on Feb. 7th 2004