Modeling Visitor Movement in Theme Parks

Gürkan Solmaz, Mustafa Ilhan Akbaş and Damla Turgut
Department of Electrical Engineering and Computer Science
University of Central Florida
Email: {gsolmaz,miakbas,turgut}@eecs.ucf.edu

Abstract—Realistic modeling of the movement of people in an environment is critical for evaluating the performance of mobile wireless systems such as urban sensing or mobile sensor networks. Existing human movement models are either fully synthetic or rely on traces of actual human movement. There are many situations where we cannot perform an accurate simulation without taking into account what the people are actually doing. For instance, in theme parks, the movement of people is strongly tied to the locations of the attractions and is synchronized with major external events. For these situations, we need to develop scenario specific models.

In this paper, we present a model of the movement of visitors in a theme park. The nondeterministic behavior of the human walking pattern is combined with the deterministic behavior of attractions in the theme park. The attractions are divided into groups of rides, restaurants and live shows. The time spent by visitors at different attractions is calculated using specialized queuing-theoretic models. We compare the realism of the model by comparing its simulations to the statistics of the theme parks and to real-world GPS traces of visitor movement. We found that our model provides a better match to the real-world data compared to current state-of-the-art movement models.

I. INTRODUCTION

Recent advances in mobile devices enabled the increased popularity and usage of mobile applications. The realistic modeling of human movement is very important for the performance assessment of mobile wireless systems. Human mobility models simulate the movement patterns of the mobile users and are a key component for simulation-based performance evaluation [1].

Early mobility models relied on some type of variations of the idea of a random walk. Examples of this include the random waypoint (RWP) [2] and Brownian Motion [3] models. These models are only a very coarse approximation of human behavior. It was found that one of the most important characteristics of human mobility is the combination of regularity and spontaneity in deciding the next destination point. This behavior can also be defined as making both deterministic and nondeterministic decisions in the same time period. For instance, theme park visitors usually pre-plan their visit. They try to optimize their time on rides while minimizing the time it takes to walk from one attraction to another. Nevertheless, when they are in the park, they may change their decisions spontaneously depending on various factors. Random mobility models are not a good match for this behavior.

The current human mobility models can be classified into trace-based [4] and synthetic [5] models. The trace-based models generally use GPS traces and Bluetooth connectivity observations. However, it is difficult to collect real data and the amount of publicly available data is limited. Therefore, synthetic models, which are defined on mathematical basis, are widely used in the simulations. Most mobility models aim for a generic human movement modeling. However, human movement patterns depend on the application scenario. The mobility of people is driven by their goals when they are moving in the environment. For instance, walks in a city center are driven by the need to rapidly reach specific destinations. In a university campus, walks are constrained in space by the destination of classrooms, meeting rooms, and cafeterias. At the same time, they are constrained in time by the schedules of classes and meetings. In an amusement center, the movement would be determined by the attractions the users planned to visit. These examples illustrate the need for the scenario-specific model of human mobility.

In this paper, we present a mobility model of visitors in a theme park. The outputs of the model are the synthetic movement tracks, pausing locations (waiting points) and pausing times. First, the fractal points are generated by the model in order to create the pausing and stopping locations. The concentrated locations of these fractal points are defined as the meeting locations of visitors or attractions. This method decreases the number of waypoints in a map, allowing the simulation of a large number of visitors. It also makes the mobility model more realistic since real attractions such as restaurants or rides in the environment can be simulated by their individual models. These locations are grouped into four main attraction types of theme parks: main rides, medium-sized rides, live shows, and restaurants. The waiting times of visitors at these attractions are modeled using queuing theory.

Let us now consider how such a model can be used for wireless mobile applications. For instance, a wireless sensor network (WSN) can be deployed in a theme park for purposes such as finding the fastest way to move from one location to another considering the current density of the crowds in different areas of the park. Such WSN can actually rely on the personal mobile devices of the visitors and can be used to offer an interactive theme park experience. Social networking applications or multi-player games can be offered to the visitors with the support of a deployed wireless system or in an ad hoc working mode. The performance of such a system would highly depend on the mobility of the users and must be evaluated by simulation before deployment.

Another class of applications of the model is supporting

theme park administration. Theme park administrators must direct visitors efficiently among attractions and balance the number of visitors at each attraction. It is desirable to balance the density of the crowd in different areas of the park. The administrators can use the mobility model to estimate the impact of different actions they can take, for instance, by deploying live entertainers and arranging the paths for pedestrian traffic dynamically. The predictive results of the models can be used to decide the locations of security personnel.

II. RELATED WORK

There are various human mobility models presented in the literature. One collection of these are the random walk model group, in which the next destination of a node and the velocity are chosen randomly based on some probability distribution. RWP [2] mobility model is the most commonly used random walk model and it creates random movement patterns for independent nodes in an area without any constraints. RWP serves as the base and has been extended for many other models [5]. However, random walk models are not appropriate for human mobility in realistic scenarios such as a theme park, since the nodes in such models do not have any prior planning in their movements.

Human mobility has several characteristic features, which have been observed by different measurement methods. Examples of these features are truncated power-law distributions of pause times, inter-contact times, fractal waypoints and heterogeneously defined areas of individual mobility. Rhee et al. [6], [7] shows that these properties are similar to the features of Lèvy walks and used these properties to design Self-similar Least Action Walk (SLAW) model. SLAW is a context based Lèvy Walk model, which produces synthetic walk traces by taking the degree of burstiness in waypoint dispersion and heavy-tail flight distribution as inputs.

According to SLAW mobility model, the mobile nodes walk from one waypoint toanother. The clusters of waypoints form the areas where the people pause or stop most of the time. In a theme park, these waypoints correspond to the areas which attract people. SLAW models human mobility in a general context where the waiting times at the waypoints are determined according to a power law distribution. However, for our particular theme park scenario, the waiting times must be defined according to the characteristics of the attractions. The attractions at theme parks can be combined into groups of main rides, medium-sized rides, live shows and restaurants [8]. The waiting times of visitors at these attractions are modeled using queuing theory. Basically, in our model, queueing theory is integrated with the mobility model, to create a realistic user mobility model for the theme parks.

There are scenario specific mobility models offered in the literature for the evaluation of wireless mobile systems. Liu et al. [9] develops a physics-based model of skier mobility in mountainous regions by considering the physical effects of gravity and the steepness of the terrain. The goal of the model is to evaluate the effectiveness of wireless communication devices in improving avalanche safety. The weighted waypoint mobility model [10] by Hsu et al. describes the pedestrian movement patterns among preferred locations on a campus. The preferred locations are predetermined in the environment and assigned with "weights", which define the probability of being selected as the destination by the pedestrians. Ro and Van Hanh [11] presented a mobility model to evaluate their solution for supporting global mobility in PMIPv6 networks. ParkSim [12] by Vukadinovic et al. is a software tool simulating the mobility of theme park visitors. The tool is used for the simulation of wireless ad hoc networks and its mobility model is driven by the possible activities the visitors would perform in the park.

III. HUMAN MOBILITY MODEL

A. Overview of the model

In this section, we present a scenario-specific mobility model for the theme park visitors. Before describing the model, let us briefly explain the fundamental characteristics of these entertainment areas. Theme parks are large areas with one or more "themed" landmarks that consist of attractions. Visitors of a theme park plan to see a subset of these attractions by walking during their scheduled visit. SLAW model provides an effective strategy in representing social contexts of common gathering places of walking people by fractal points and heavy-tail flights on top of these fractal points. We use this idea as a baseline for a more realistic mobility model and apply queuing models to represent the behavior and effects of different types of attractions on mobility of theme park visitors.

B. Modeling a theme park

In this subsection, we describe the five main phases to model a theme park. The model of a theme park is built upon these phases starting with the first phase of fractal points generation.

1) Fractal points: We use the term *fractal points* based on the usage for the SLAW mobility model [7]. As a human behavior, people are more attracted to visit popular places. This characteristic of human walk is expressed in the mobility model by the fractal points. In our model, the fractal points are initially created in an empty area using the same technique in SLAW. A fractal point can be considered as a waypoint at the beginning. All fractal points and the area, in which these points are generated in, are used to form a landmark. Fig. 1 demonstrates the first phase of the model in a scenario, where 1000 fractal points are generated in an area of 1000×1000 meters.

2) Clusters: After generation of fractal points, we determine the parts of the area with highest fractal points. The goal of this phase is finding the areas where people are more attracted to gather together.

We use a modified version of DBScan [13] algorithm on the generated fractal points to find the attraction locations. DBScan is a density based clustering algorithm for discovering clusters with noise points, which has two input parameters, epsilon "Eps" and minimum number of points (neighbors) "MinPts".
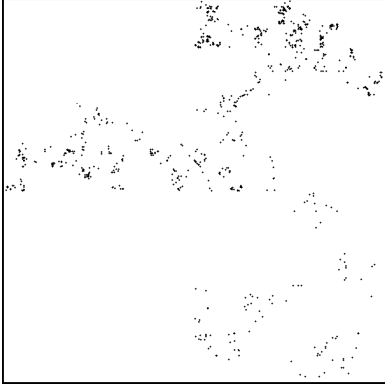
Fig. 1. Fractal point generation phase of the model.

Fig. 2. Clusters generated by DBScan over 1000 fractal points.

The *Eps-neighborhood* of a point p, denoted by $N_{Eps}(p)$, is defined by $N_{Eps}(p) = \{q \in D | dist(p,q) \leq Eps\}$ where D is the database of points. In our DBScan approach, for each point in a cluster, there are at least "MinPts" neighbors in the *Eps-neighborhood* of that point.

In our model, DBScan algorithm is modified based on our requirements. The input parameters include epsilon, minimum number of neighbors, number of clusters and proportion of noise points among all fractal points. The number of clusters and noise point ratio are used to specify a landmark. For instance, if there are 25 attractions in a theme park, the number of clusters becomes 25. The noise point ratio is used to determine the nondeterminism in mobility empirically (e.g. 10%) or based on statistical data collected from the visitors of a theme park. The values of the minimum number of neighbors and epsilon are searched with a heuristic, which alters the values of these two parameters based on the previous results. Changing the values of these two parameters directly changes resulting number of clusters and noise point ratio. For instance, if epsilon has a larger value and minimum number of neighbors has a smaller value compared to the previous iteration, DBScan produces less number of clusters, with a smaller noise point ratio.

Let us assume a landmark is required to have 5 queues and the proportion of noise points to be approximately 0.10. The initial epsilon and minimum number of neighbors are set as 30 meters (for dimensions of 1000×1000 meters) and 8 empirically. After setting the initial values, the fractal points are scanned iteratively and the values of new epsilon and number of neighbors parameters are set based on the results of the last iteration. When there are expected number of queues and expected approximate proportion of noise points, the clustering of fractal points is finalised and the parameters get their final values.

After the clustering of fractal points, the most dense areas are determined. Fig. 2 shows an example clustering output, over 1000 fractal points in an area of 1000×1000 meters. In this example, 15 clusters are generated and marked, whereas the fractal points, with a proportion of 10% among all fractal points are not clustered.
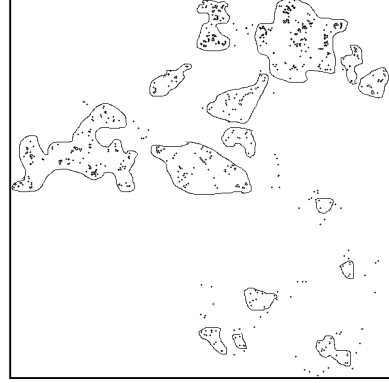
3) Attractions and Noise Points: The noise points are obtained from the previous step, and we categorize clusters as queues. The most dense areas found by clustering step are marked as "attractions". In our model, attractions are represented by queuing models. We decide on the types and the weights of the queues based on the number of waypoints and previous work on theme park design. The weight of a queue is defined according to the number of fractal points included in its corresponding cluster. The central point of a queue is the average position of all the fractal points included in its corresponding cluster. Non-clustered fractal points are marked as "noise points". According to Wanhill [8], the attractions in a theme park are defined by queuing models and their percentages are distributed as given in Table I.

TABLE I
ATTRACTION PERCENTAGES

Attraction	Queue model	Percentage
Main rides (RD)	M/D/n	17%
Medium-size rides (M-RD)	M/D/n	56%
Restaurants (RT)	M/M/1	17%
Live shows (LS)	M/M/n	10%

Each attraction has a corresponding queue type and the queue types have their particular properties. $M/D/n$ queue has a constant service time, whereas $M/M/1$ and $M/M/n$ queues have service times according to exponential distribution.

In our model, $M/D/n$ queue model is used for the main rides and the medium-sized rides since they have very similar queue behaviors. $M/M/1$ queue model is used for restaurants and $M/M/n$ queue model is employed for live shows. Restaurants and live shows have exponential service rates, while main rides and medium-sized rides have constant service rates.

The attractions and noise points in the model are defined as the waiting points in a landmark. Basically, attractions and noise points form a landmark in our model.

4) Landmarks: Landmarks are generated as a result of the previous steps, including the generation of fractal points, density-based iterative clustering and generation of queues according to their weights, queue types, and service rates to
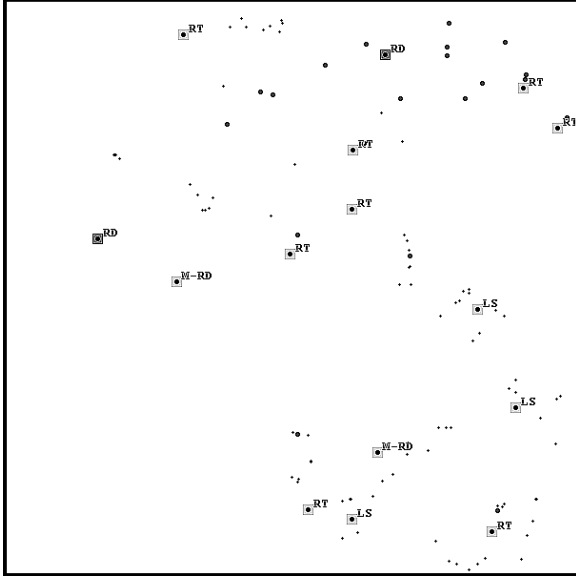
Fig. 3. A landmark model including queues, noise points and initially distributed mobile nodes.

represent attractions in a landmark.

A landmark is a place where there are multiple queues and noise points. Each landmark has two dimensions specifying its size. Fig. 3 shows a landmark model with initial placement of 20 visitors (mobile nodes), queues and noise points in an area of 1000×1000 meters. In this figure, central points of the queues are represented by squares. The noise points and initially located mobile nodes are shown by small dots and circles respectively. Each queue is presented with its queue type: main rides (RD), medium-sized rides (M-RD), live shows (LS), and restaurants (RT).

The proposed landmark model is used to model areas such as a landmark in a theme park or an open-air event on a university campus by assigning the number of queues and the proportion of noise points accordingly.

5) Theme Park Map: For modeling the theme park, we use a graph theoretical approach. The theme park map is modeled as a graph consisting of vertices and weighted non-directional edges. Each vertex in the graph represents a landmark. If there is a path between two landmarks, an edge is added with a weight corresponding to the transportation time between the two landmarks.

Theme parks are usually large areas with transportation services among the main locations of attractions such as buses, trains and cars. In our model, we separate the landmarks in a theme park such that people only walk inside, and there is no fast, motorized transportation available. Furthermore, most of the theme parks are located in non-uniform 2D areas, which creates another challenge to simulate a theme park with a model assuming a uniform 2D area. By separating the areas of attractions as landmarks in a theme park and adding weighted non-directional edges between the landmarks, we generalize

the model of human mobility in a landmark to the human mobility in the large theme park. The mobility model includes the landmarks for human walks and the edges between them for transportation of visitors. We do not assume a theme park as a uniform 2D area, since it includes geographical obstacles such as areas without pavements for pedestrians and paths or roads used for transportation. These characteristics of our mobility model allows our model to be more realistic compared to the existing mobility models of theme park visitors.

C. Visitor model

In the model, the visitors of the theme park are represented by mobile nodes. We define the states of the mobile nodes in a landmark as "initial", "inQueue", "moving", "inNoisePoint" and "removed". At the beginning of the simulation, all mobile nodes are in "initial" state. A mobile node changes its state to "inQueue" when it starts waiting in a queue. It changes its state to "inNoisePoint" when it starts waiting in a noise point. There are two different states for waiting, since waiting in a noise point and in a queue are not the same. When a mobile node is changing its location to arrive to a new destination, which may be a queue or a noise point in the landmark, it is in the "moving" state. The state of a mobile node is "removed" when the hangout time of the node passes.

Initially, each visitor decides a hangout time for the particular landmark, a time a visitor plans to stay in the landmark. Hangout times are generated randomly by exponential distribution for each visitor. After deciding the hangout time, each of the visitors selects a subset from the set of all queues in the landmark to visit with particular hangout time. The size of the subset (the number of queues to visit) selected by a visitor is proportional to the corresponding hangout time of that visitor. If the visitor is not in "inQueue" state when the hangout time ends, the visitor leaves the landmark. In other words, the visitor arrives at an exit point of the landmark. If the visitor is waiting in a queue, (in "inQueue" state), the visitor continues to wait in the queue and leaves the landmark after being serviced. We assume every visitor has a constant speed of 1 m/sec. After queue selection, the visitors move according to the least action principle among the selected queues and noise points. If a visitor visits a queue or a noise point, that point is marked as visited and is not visited again by the same visitor.

Visitors decide their next destinations by Algorithm 1, which is a modified version of Least Action Trip Planning (LATP) [7] algorithm. In the LATP algorithm, a visitor tries to minimize the Euclidean distance traveled from a waiting point to a new waiting point (destination). The waiting points are the queues to be visited and the noise points in the landmark. This strategy is different than Dijkstra's Shortest Path algorithm, since it does not always cause the new destination to be the nearest waiting point, where every unvisited point has a probability to be the next destination. The parameter alpha is used to determine this probability. We modified the algorithm to match the requirements of our mobility model. Instead of using identical waiting points, weighted waiting points are used. The weight of noise

point is always 1 and the weight of a queue is set as the number of fractal points included in its corresponding cluster. In this approach, queues with larger weight values, such as main rides, have more probability to be selected as the new destination points. In other words, visitors are more attracted to gather in queues with larger weight values. For calculation of Euclidean distances, we used the exact positions of the noise points in the landmark and positions of the central points of queues.

Algorithm 1 Algorithm for deciding the next destination

1: Q: Set of all unvisited queues planned to visit
2: N: Set of all unvisited noise points
3: W: Set of weights for queues
4: cp: Current position of the visitor
5: Pr: Empty set of probabilities
6: **for** each n in N **do**
7:    $d(cp, n) :=$ Euclidean distance between cp and n
8:    $Pr(n) := (\frac{1}{d(cp,n)})^\alpha$
9: **end for**
10: **for** each q in Q **do**
11:    $d(cp, q) :=$ Euclidean distance between cp and q
12:    $Pr(q) := (\frac{1}{d(cp,q)})^\alpha$
13:    $Pr(q) := Pr(q) * W(q)$
14: **end for**
15: Select a point p according to probabilities Pr from the set $Q \cup N$
16: **if** $p \in N$ **then**
17:    **return** Position of the point p
18: **else**
19:    **return** Position of a random sit-point in the queue p
20: **end if**

At each iteration of the simulation, we check the queues to find the number of visitors serviced and the visitor states for possible changes. For instance, if a visitor is serviced by a queue at a time step, the state of the visitor must change from "inQueue" to "moving".

When a queue is selected, the visitor goes to a random sit-point inside the cluster area as the new destination position. Each queue in the landmark has a service rate and the number of visitors per service. Waiting time of a visitor in a queue depends on the number of visitors already waiting in the queue ahead of that visitor, service rate and the number of visitors per service of the queue. For instance, if the queue is modeled by an $M/M/1$ queue, number of visitors per service is equal to 1, and if the queue is modeled by an $M/M/n$ queue, number of people per service is equal to n. When a visitor goes to a noise point, the visitor starts waiting in the position of the noise point. The waiting time of the visitor is generated randomly by truncated Pareto distribution.

D. Application to real scenario

The mobility model can be easily applied to model a real theme park scenario. Each smaller park in a large theme park

Fig. 4. An illustration of the application of model to a real-world scenario: Disney World parks in Orlando.

would be modeled as a landmark. For each park, real dimension lengths are used to specify the 2D rectangle area of a landmark. OpenStreetMap [14] can be used to determine the longitudes and latitudes for the theme parks.

In a real scenario, the number of attractions and types of those attractions are generally known. If types of every attraction in the park are not known, the queue types and numbers specified in this paper can be potentially used. With this approach, a portion of a theme park can be modeled as a landmark. After generation of landmarks, edges and the weight values are set.

Each visitor in this scenario has a total hangout time which is the amount of time to spend in the theme park. Initially, visitors decide the parks (landmarks) to be visited in an order that provides minimum times for transportation (minimum weights) between them. For each landmark, a visitor also plans to visit particular attractions (queues) whenever the visitor comes into a new landmark. Whenever a visitor finishes hanging out in a park, the visitor goes to the next planned park, through the edge between the two parks.

Assume a large theme park consisting of three main parks, as shown in Fig. 4. In this figure, landmarks are the vertices and the lines connecting landmarks are the edges with different weights. OpenStreetMap [14] is used to illustrate the model on this map. Fig. 4 contains three main parks of Disney World in Orlando. These main parks are Epcot, Animal Kingdom and Hollywood Studios. The last main park, Magic Kingdom is not included for illustration purposes. As you can see in the figure, the parks have labels L_1, L_2 and L_3, and the weights of the edges between the parks have labels W_1, W_2 and W_3. Landmarks can be generated according to the real sizes of the areas of the main parks, and the weights are set with the real transportation times. Dimension lengths and numbers of attractions are also set for each main park.

The simulation of the model applied to the real scenario by this graph theoretical approach, generates realistic synthetic traces of visitors mobility in theme parks.
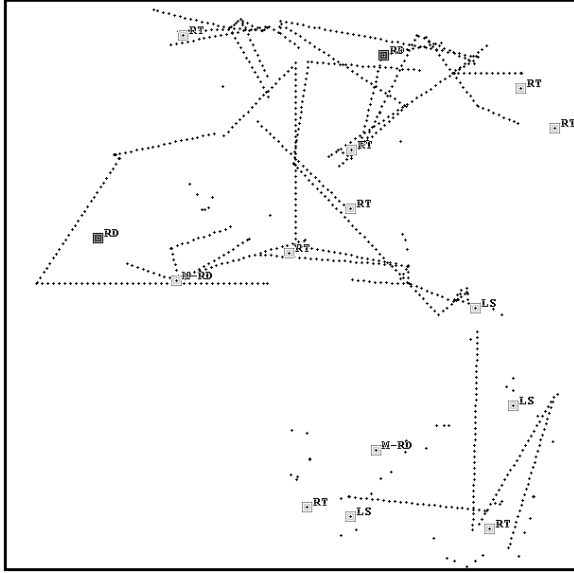
Fig. 5. A simulation of movement of 20 mobile nodes in the landmark after 1 hour simulation time.
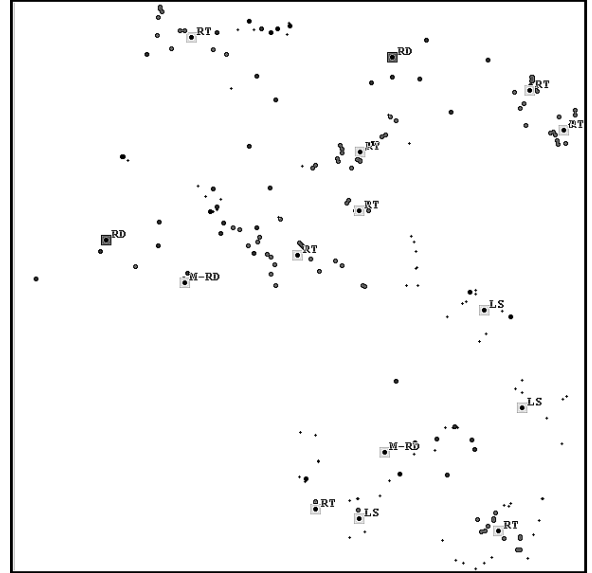

Fig. 6. Positions of 200 mobile nodes distributed in the landmark after 1 hour simulation time.

IV. SIMULATION STUDY

A. Simulation environment

In this paper, the experiments are carried out to validate our mobility model in landmarks. The set of landmarks with edges between them can be used to simulate the mobility in and among the landmarks in a theme park map.

The simulation of our model generates synthetic mobility traces of mobile nodes in a specified 2D area, with dimension length, number of queues and noise point ratio. Figure 5 shows an output example of a simulation run with 20 mobile nodes, which is taken when simulation time is 3600 seconds.

Each mobile node in the landmark draws its trajectory lines while moving. These trajectory lines are the consecutive points in the figure, that illustrate mobility of the mobile nodes in the landmark. The waiting points are the points where direction of a mobile node changed in the trajectory. The waiting points are either noise points or random sit-points close to a central point of a queue. Figure 6 demonstrates another simulation with 200 mobile nodes after 3600 seconds of simulation time. In this figure, by looking at the positions of the mobile nodes represented by small circles, we can see the expected human behavior to gather in common places such as the dense regions of waypoints in the model. The use of fractal points and planning trips according to the least action principle somewhat represents this social human behavior.

We conducted simulation experiments for square landmarks with dimensions of 1000×1000 meters and 2000 mobile nodes (visitors). For all the experiments, total simulation time is 36000 seconds. Mobile nodes have hangout times exponentially distributed between 2 hours and 10 hours. For a 1000×1000 landmark, we used 15 queues and approximately 0.1 noise point ratio. Each queue has a service rate and a number of visitors

per service parameters. For a simulation of 2000 mobile nodes, we used 40 as the number of visitors per service for main rides, 20 for medium rides and live shows. Mobile nodes are initially randomly distributed to the fractal points as their initial locations. We assume each mobile node has a constant speed of 1 m/s. Minimum waiting time in a noise point is 30 seconds and Pareto alpha value is 1.5. In Algorithm 1, alpha value used for deciding the next destinations is 3.0.

At the beginning of each simulation, each mobile node is assigned with the queues to visit, according to its hangout time and number of queues in the landmark, before starting to move. A mobile node exits the area if its hangout time is over. If a mobile node is waiting in a queue when hangout time is passed, it keeps waiting and is being removed after serviced by the queue. Mobile nodes keep track of visited queues and noise points so that once a mobile node visits a queue or a noise point, it does not come back to the same queue again. For a waiting point, the visitors visited that point in a particular time and their departure times from that waiting point are also keep tracked, for each queue and noise point in the landmark.

B. Simulation results

In this section, the simulation results are analyzed by comparing 41 GPS traces (taken from CRAWDAD archive at Dartmouth College) collected from 11 volunteers who spent their Thanksgiving or Christmas holidays in Disney World, SLAW [7] and RWP [2] mobility model simulations. We examine fundamental characteristics of mobility features, including distribution of flight lengths, distribution of waiting times (pause times) and waiting rate of visitors.

Equally sized areas are used for the comparison of our model, SLAW and RWP. For the GPS traces, we assume that a visitor is not walking if the visitor moves for more than 150 meters
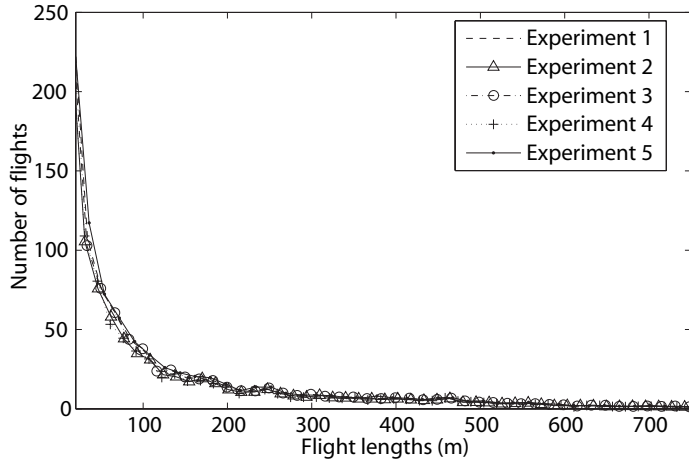
Fig. 7. Normalized flight length distributions from mobility simulation.

in 30 seconds sampling time, which would exceed the regular human speed of 5 m/s. Accordingly, we filtered the data where the visitors are not walking, but possibly traveling in a bus or another vehicle in the theme park. If a mobile node is in a circular area with a radius value of 10 meters in the time period in consecutive sampling times of 30 seconds, we assume that the mobile node is waiting in a waiting point.

1) Experiment 1: Flight Lengths. A flight length is the distance between two consecutive waiting points of a visitor. Flight length distribution is one of the most significant characteristics of human mobility models, since it allows us to make realistic comparison of human mobility extracted from GPS traces with random walk models and other mobility models. In this experiment, we compared flight length distribution of the simulation with GPS traces, SLAW and RWP mobility models.

The first set of experiments are conducted by using the same parameter settings to verify that the simulation is consistent according to the flight length distributions. For each experiment, we normalize the number of flight lengths to 1000. The flight length distributions of 5 randomly selected experiments are given in Fig. 7. These experiments have flight length counts between 64000 and 68000; however, all the experiments are normalized to the flight length count of 1000. As expected, flight length distributions are consistent, have similar characteristic, and there is no significant difference between the distribution lines. The experiment shows the similar expected behavior of the synthetic simulation of the mobility model, among different runs of the simulation model.

In another set of experiments, flight length distribution of the simulation model is compared to GPS traces, SLAW and RWP mobility simulation results. The results of the simulations are given in Fig. 8. Our proposed mobility model outperforms other two synthetic mobility models. The flight length distribution of our model is closer to the GPS traces compared to SLAW or RWP mobility models. SLAW has a similar characteristic but shorter flight lengths and RWP model has a uniform distribution as expected. Fig. 8 also shows that flight length distributions of

RWP model is significantly different than the GPS traces, which proves modeling human mobility with this model is not realistic. On the other hand, our simulation model, GPS traces and SLAW mobility model have heavy-tail flight distributions.
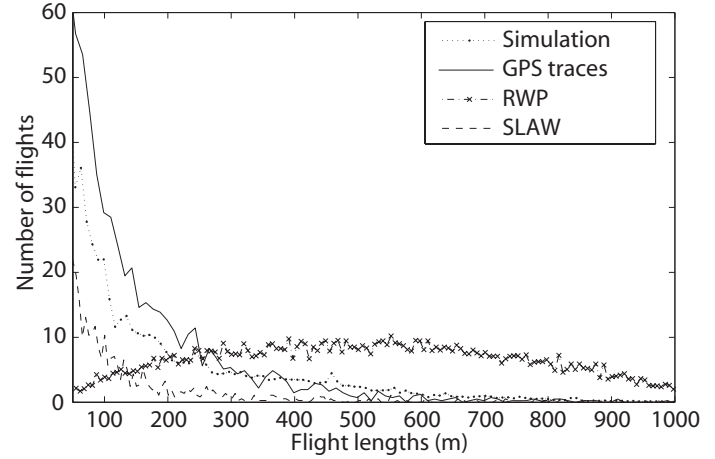

Fig. 8. Normalized flight length distributions from the mobility simulation, Orlando Disney World GPS traces, SLAW and RWP Mobility Models.

In the last set of experiments for flight lengths, we modeled Disney Magic Kingdom, which is one the four main parks in Disney World. The square area for Magic Kingdom is modeled as a landmark with dimensions approximately equal to 850x850 meters using OpenStreetMap [14]. RWP model is not used for this experiment set, since our model and SLAW provide better results in the previous set of experiments.

For the Magic Kingdom landmark simulation, our mobility model performs significantly better than SLAW. Fig. 9 shows that the flight length distribution of our model is very close to the flight length distribution of GPS traces.

2) Experiment 2: Number of Waiting Points. In this experiment, we analyze the number of waiting points averaged for one hour for the synthetic mobility models and the GPS traces. Average number of waiting points of GPS traces is approximately 10.5, which means every visitor is waiting at
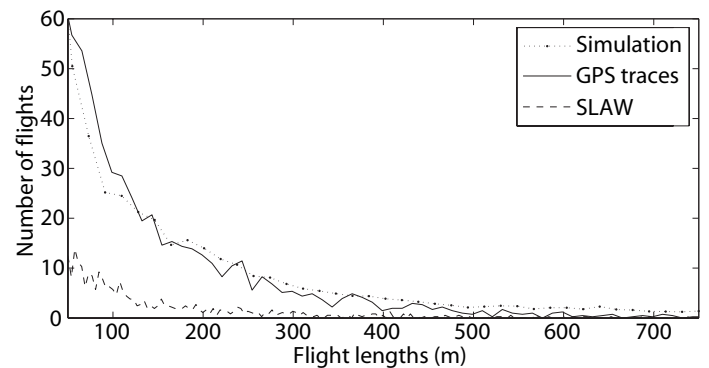

Fig. 9. Normalized flight length distributions from the mobility simulation, Orlando Disney World GPS traces and SLAW.

roughly 10 different locations in an hour on average. For SLAW mobility model, the average numbers of waiting points are close to 20, that doubles the GPS traces, while it is approximately 7.5 for our simulation and 3.3 for RWP model. The results of the experiment set are given in Figure 10 which shows that our mobility model performs significantly better than the other two synthetic simulations in terms of similarity to the GPS traces.
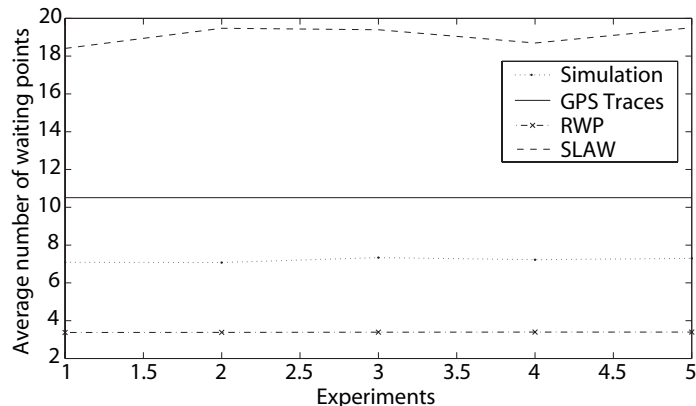

Fig. 10. Average waiting points per one hour from the simulation, Orlando Disney World GPS traces, SLAW and Random Waypoint Mobility Model.

3) Experiment 3: Waiting Times. In this experiment, we compare waiting time distribution of the simulation with GPS traces and SLAW. RWP model is not used in this experiment since the mobile nodes have constant waiting times. This constant waiting time value can be changed by the constant input parameter of the RWP model.

To compare the waiting times of the two mobility models with the GPS traces, the results of these three traces are normalized to 1000. Fig. 11 shows that waiting time distribution of our model is again closer to the GPS traces, compared to SLAW mobility model, and both simulations have similar statistical behavior of heavy-tail waiting time distribution. By setting service rate and number of visitors per service parameters of each of the different queue types realistically, one can obtain more accurate results
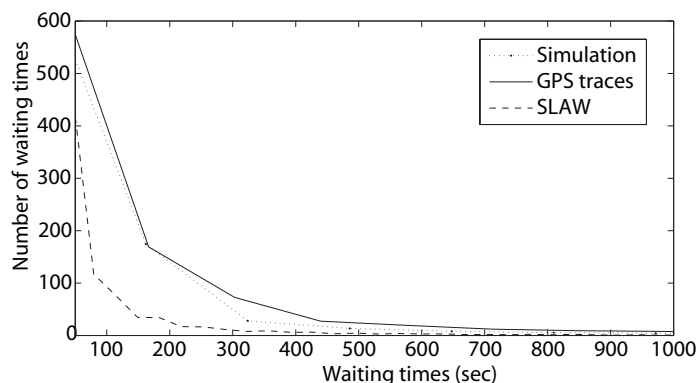

Fig. 11. Normalized waiting time distributions from the simulation, Orlando Disney World GPS traces and SLAW.

very similar to the real-world scenario of theme park visitors mobility.

V. Conclusion

In this paper, we present a model of the movement of visitors in a theme park. In this model, we combine the nondeterministic behavior of the human walking pattern with the deterministic behavior of attractions in the theme park. We divide the attractions into groups of main rides, medium-sized rides, live shows and restaurants. We use queuing-theoretic models to calculate time spent by visitors at different attractions. We validate accuracy of our model through extensive simulations using theme park statistics, GPS traces collected in a real theme park and the data generated by simulations of other mobility models. We find our model provides a better match to the real-world data compared to current state-of-the-art movement models.

As a future work, we plan to include basic micro-mobility behaviors in the model to achieve more realistic synthetic simulations. Furthermore, we are planning to use the mobility model to evaluate applications of wireless sensor networks with mobile elements for theme park scenarios.

References

[1] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research," *Wireless Communications and Mobile Computing*, vol. 2, no. 5, pp. 483–502, Sep. 2002.

[2] J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, and J. Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocols," in *Proc. of ACM/IEEE MobiCom*, Oct. 1998, pp. 85–97.

[3] R. Groenevelt, E. Altman, and P. Nain, "Relaying in Mobile Ad Hoc Networks: The Brownian Motion Mobility Model," *Wirel. Netw.*, vol. 12, no. 5, pp. 561–571, Sep. 2006.

[4] N. Aschenbruck, A. Munjal, and T. Camp, "Trace-based mobility modeling for multi-hop wireless networks," *Computer Communications*, vol. 34, no. 6, pp. 704–714, May 2011.

[5] F. Bai and A. Helmy, "A Survey of Mobility Models in Wireless Ad Hoc Networks," in *Wireless Ad Hoc and Sensor Networks*, A. Safwat, Ed. Springer Verlag, Oct. 2006, pp. 1–30.

[6] I. Rhee, M. Shin, S. Hong, K. Lee, S. J. Kim, and S. Chong, "On the Lèvy-walk Nature of Human Mobility," *IEEE/ACM Transactions on Networking*, vol. 19, no. 3, pp. 630–643, Jun. 2011.

[7] K. Lee, S. Hong, S. J. Kim, I. Rhee and S. Chong, "SLAW: A Mobility Model for Human Walks," in *Proc. of INFOCOM*, Apr. 2009, pp. 855–863.

[8] S. Wanhill, "Theme Parks: Their Development and Operation," in *Proc. CAUTHE Conference*, Feb. 2006, pp. 1889–1921.

[9] X. Liu, C. Williamson, and J. Rokne, "Physics-based modeling of skier mobility and avalanche rescue in mountainous terrain," in *Proc. of IEEE LCN*. IEEE Computer Society, Oct. 2010, pp. 645–652.

[10] W. Hsu, K. Merchant, H. Shu, C. Hsu, and A. Helmy, "Weighted waypoint mobility model and its impact on ad hoc networks," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 9, no. 1, pp. 59–63, Jan. 2005.

[11] S. Ro and H. V. Nguyen, "Global mobility support in Proxy Mobile IPv6 for mobility-unaware hosts." in *Proc. of IEEE LCN*, Oct. 2009, pp. 593–600.

[12] V. Vukadinovic, F. Dreier, and S. Mangold, "A simple framework to simulate the mobility and activity of theme park visitors," in *Proc. of WSC*, Dec. 2011, pp. 3248–3260.

[13] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise," in *Proc. of the International Conference on Knowledge Discovery and Data Mining*, Aug. 1996, pp. 226–231.

[14] M. Haklay and P. Weber, "OpenStreetMap: User-generated street maps." *Pervasive Computing*, vol. 7, no. 4, pp. 12–18, Dec. 2008.