

Queue management

Field

The present invention relates to queue management in an amusement park or other
5 environment in which queues can form.

Background

Amusement parks (or “theme parks”) seek to improve the experience of visitors.

10 One approach is to reduce waiting times for rides by introducing fast lanes which provide priority access to rides. The queuing time in a fast lane is significantly less than that for a normal lane by design. Visitors can access fast lanes by purchasing fast lane tickets allowing access to rides via fast lanes. Alternatively, visitors may be able to book a specific time slot for a particular ride to access via its fast lane.

15 Reference is made to S. Cheng, L. Lin, J. Du, H. C. Lau and P. Varakantham: “An agent-based simulation approach to experience management in theme parks” 2013 Winter Simulations Conference (WSC), 2013, pp. 1527-1538, doi: 10.1109/WSC.2013.6721536.

Summary

20 According to a first aspect of the present invention there is provided a computer-implemented method comprising dynamically (or “in real time”, for example, every 10 or 15 minutes) calculating supply and demand of queue slots available to visitors to a plurality of spaced-apart attractions (such as rides) distributed across a domain (such
25 as a theme park, integrated resort, leisure complex, transport hub or cruise ship) and dynamically managing reservations of queue slots made by visitors based on the supply and demand and/or dynamically providing incentives (such as vouchers) to visitors for influencing behaviour and/or location of visitors in the domain based on the supply and demand. The method may comprise dynamically balancing supply and demand of
30 queue slots available to visitors to the plurality of spaced-apart attractions.

Thus, the method can be used for helping proactively manage and/or influence visitors in the domain.

35 Dynamically calculating the supply and demand of queue slots available to visitors to the plurality of spaced-apart attractions distributed across the domain may comprise

dynamically predicting the supply of queue slots available to visitors to the plurality of spaced-apart attractions distributed across the domain, the queue slots having specified times, and dynamically predicting demand for the queue slots by visitors for the plurality of spaced-apart attractions

5

Each queue slot can have a respective cost (or price). Visitors can earn a reward (or a fee) depending on behaviour and/or following a recommendation or incentive, which can be used to contribute to, or cover, the cost of the queue slot. The cost and the reward may be expressed in terms of a number of tokens.

10

Each attraction may have a first type of queue (or “normal lane”) and a second, preferential type of queue (or “fast lane”). The queue slot may be used to enter the second type of queue. Each attraction may have only two types of queues.

15

The method may comprise determining location of visitors using terminals at the attractions for recording entry and/or departure of visitors.

The method may comprise generating an itinerary plan for a respective visitor. The itinerary plan may include an attraction based on an allocated queue slot. The itinerary plan may include a list including the attraction. The itinerary plan may include a list of two or at least two attractions, arranged in order, and based on allocated queue slots. The itinerary plan may include a list of three or at least three attractions arranged in order and based on allocated queue slots. The method may further comprises transmitting the itinerary plan to a mobile device associated with the visitor.

25

The itinerary plan may include one or more attractions accessed by normal lanes. The itinerary plan may include time window(s) for the normal lanes and queue slot(s) for fast lanes. An end time of a time window in an itinerary plan for an attraction accessed by a normal lane may be based on a predicted waiting time for the attraction and a buffer time.

30

Dynamically calculating supply and demand of queue slots may comprise receiving current pricing data which includes a current value of a cost of securing a slot to an attraction and a current value of a reward for visiting the attraction, current index data relating to current operational conditions of attractions, target index data relating to desired operational conditions of attractions and sorted pricing data comprising a set of

35

factors which affect a cost of securing a slot to an attraction and a set of corresponding weights for the set of factors, providing the current pricing data, current index data, target index data and sorted pricing data to an optimization algorithm to find adjusted price data and upon reaching a predetermined target, outputting the adjusted price data.

The operational conditions of attractions may include crowd distribution in a domain across which attractions are distributed. The operational conditions of attractions may include waiting times of attractions. The waiting times of attractions may include normal line waiting times and fast line waiting times. The operational conditions of attractions may comprise popularity of attractions.

The reward for visiting the attraction may be expressed as a number of tokens.

The optimization algorithm may be a Bayesian optimization algorithm. The optimization algorithm may comprise a Gaussian process regressor and an acquisition function.

The set of factors which affect the cost of securing the slot to the attraction may include a type of day, wherein types of day include a week day, a day of a weekend or a holiday.

The set of factors which affect the cost of securing the slot to the attraction may include seasonality, wherein seasonality includes peak or off-peak.

Dynamically managing reservations of queue slots made by visitors may comprise, in response to a user request, selecting at least one ride and at least one queue slot which is relatively less popular than a queue slot for the ride or another ride and presenting the selected ride(s) and queue slot (s) to the visitor. The queue slot for the ride or other ride may be that requested by the user in the user request. The queue slot for the ride or other ride may be selected without the user requesting a preferred slot for a given ride. The method may comprise transmitting the selected ride(s) and queue slot (s) to the visitor in a message to a mobile device of the visitor.

Dynamically providing incentives to visitors for influencing behaviour and/or location of visitors in the domain based on the supply and demand may comprise receiving status information about visitor(s) in a domain across which a plurality of spaced-apart

attractions are distributed, generating an incentive for affecting behaviour and/or location of a visitor in dependence upon the status information and transmitting the incentive to a mobile device of the visitor.

5 Status information may be behaviour, fulfilment and/or location information.

The status information may indicate a crowding condition in a first region and the incentive may be for encouraging the visitor to visit a second region which does not have a crowding condition. Crowding condition may be indicated by a number or a
10 density of visitors in a region exceeding a threshold value of the number or the density.

The status information may indicate compliance with a proposed itinerary and the incentive may be for rewarding with a proposed itinerary or encouraging compliance with a future proposed itinerary. The status information may indicate a shortfall in
15 delivery and the incentive is for compensating the visitor.

The incentive may comprise a voucher in relation to a product or service. The incentive may comprise a gift. Generating the incentive may depend on rewards earned by the visitor in the domain.

20 The method may further comprise receiving historical queue data about queue lengths for a plurality of spaced-apart attractions distributed across a domain and receiving predicted queue data about the queue lengths for the plurality of spaced-apart attractions. Generating the incentive may depend on the historical and predicted queue
25 data, and the information about visitors currently in the domain, generating incentives for at least some of the visitors currently in the domain to encourage at least some of the visitors to move to given areas within the domain. The incentives may include vouchers for redemption at retail outlets.

30 According to a second aspect of the present invention there is provided a computer-implemented method comprising receiving a first set of parameters relating to access and/or capacity attributes of a plurality of spaced-apart attractions distributed across a domain, receiving a second set of parameters relating to temporally-variable characteristics affecting at least some of the plurality of spaced-apart attractions,
35 building a model of the plurality of spaced-apart attractions based on the first and second sets of parameters, receiving data relating to a time-dependent supply of visitors

to the domain, determining, for each of the plurality of spaced-apart attractions, a predicted number of visitors to the attraction as a function of time and outputting and/or storing the predicted number of visitors.

5 The method may further comprise receiving, for each of the plurality of spaced-apart attractions, historical data relating to historical numbers of visitors to the attraction as a function of time. Building the model may be based on the data relating to historical data. The method may further comprise receiving, for each of the plurality of spaced-apart attractions, pricing data affecting attractiveness of the attraction to visitors. The
10 first set of parameters may comprise capacity of each of the plurality of spaced-apart attractions. The first set of parameters may comprise capacity of one or more queuing areas for each of the plurality of spaced-apart attractions. The first set of parameters may comprise spatial layout of the plurality of spaced-apart attractions. The first set of parameters may comprise ingress and egress configurations of the plurality of spaced-apart attractions. The model may be a spatial-temporal model.
15

The method may comprise determining, in a spatial view, an effect on the number of visitors to a given attraction by numbers of visitors to at least nearest neighbour attractions to the given attraction. The method may comprise determining, in a
20 temporal view, an effect on a rate of change of the number of visitors to a given attraction by a rate of change of numbers of visitors to at least nearest neighbour attractions to the given attraction.

The domain may be an amusement park. The domain may be a holiday resort, an
25 integrated resort, a recreational village (which may or may not include one or more hotels or other form of lodging), a hotel complex, a cinema complex, a convention center, an exhibition center, a cruise ship, or a transport hub (such as a train station or airport) which includes a variety of attractions. The domain may be a tourist attraction (for instance, a part of a city, a historical site, or a place of natural beauty), an art
30 gallery, a museum, a shopping mall or plaza, or a restaurant. A domain may comprise two or more different types of attractions having a respective limited throughput and for which queues (*e.g.*, fast and normal queues) can form. A limited throughput means that an attraction has a capped or limited capacity to serve visitors in a given period of time (for example, 300 people every 30 minutes).

An attraction may be a cinema, a theatre, a concert hall, or other similar form of performance venue (which may be indoors, outdoors or covered). An attraction may be a casino or a part off a casino (such as a table game). An attraction may be a sporting facility, such as a racket-sport court, a games pitch (such as a synthetic grass pitch), a
5 table (such as a snooker table or table tennis table), a gym, a swimming pool, a climbing wall or other area for physical recreational activity. An attraction may be a playground, a play area or ball pool. An attraction may be a class, such as an exercise class (such as aerobics, spinning and the like), a dance class or a yoga class.

10 According to a third aspect of the present invention there is provided a computer-implemented method comprising receiving historical data relating to a set of factors which affect a cost of securing a slot to an attraction, training a machine learning model using the historical data to obtain a trained model, receiving current data relating to the set of factors and, in dependence upon the trained model and the current data relating
15 to the set of factors, generating a set of weights for the set of factors.

According to a fourth aspect of the present invention there is provided a computer-implemented method comprising receiving current pricing data which includes a current value of a cost of securing a slot to an attraction and a current value of a reward
20 for visiting the attraction, current index data relating to current operational conditions of attractions, target index data relating to desired operational conditions of attractions and sorted pricing data comprising a set of factors which affect a cost of securing a slot to an attraction and a set of corresponding weights for the set of factors, providing the current pricing data, current index data, target index data and sorted pricing data to an
25 optimization algorithm to find adjusted price data and, upon reaching a predetermined target, outputting the adjusted price data.

The operational conditions of attractions may include crowd distribution in a domain across which attractions are distributed. The operational conditions of attractions may
30 include waiting times of attractions. The waiting times of attractions may include normal line waiting times and fast line waiting times. The operational conditions of attractions may comprise popularity of attractions.

The reward for visiting the attraction may be expressed as a number of tokens.

The optimization algorithm may be a Bayesian optimization algorithm. The optimization algorithm may comprise a Gaussian process regressor and an acquisition function.

- 5 The set of factors which affect the cost of securing the slot to the attraction may include a type of day, wherein types of day include a week day, a day of a weekend or a holiday.

The set of factors which affect the cost of securing the slot to the attraction may include seasonality, wherein seasonality includes peak or off-peak.

10

The cost of securing the slot to an attraction may be expressed as a number of tokens.

- According to a fifth aspect of the present invention there is provided a computer-implemented method, comprising receiving status information about visitor(s) in a
15 domain across which a plurality of spaced-apart attractions are distributed, generating an incentive for affecting behaviour and/or location of a visitor in dependence upon the status information and transmitting the incentive to a mobile device of the visitor.

Status information may be behaviour, fulfilment and/or location information.

20

The status information may indicate a crowding condition in a first region and the incentive is for encouraging the visitor to visit a second region which does not have a crowding condition. Crowding condition may be indicated by a number or a density of visitors in a region exceeding a threshold value of the number or the density.

25

The status information may indicate compliance with a proposed itinerary and the incentive is for rewarding with a proposed itinerary or encouraging compliance with a future proposed itinerary. The status information may indicate a shortfall in delivery and the incentive is for compensating the visitor.

30

The incentive may comprise a voucher in relation to a product or service. The incentive may comprise a gift. Generating the incentive may depend on rewards earned by the visitor in the domain.

35

According to a sixth aspect of the present invention there is provided a computer-implemented method comprising generating an incentive for affecting behaviour

and/or location of a visitor in dependence upon rewards earned by the visitor in a domain and transmitting the incentive to a mobile device of the visitor.

5 The method may further comprise receiving historical queue data about queue lengths for a plurality of spaced-apart attractions distributed across a domain, and receiving predicted queue data about the queue lengths for the plurality of spaced-apart attractions, wherein generating the incentive depends on the historical and predicted queue data, and the information about visitors currently in the domain, generating incentives for at least some of the visitors currently in the domain to encourage at least
10 some of the visitors to move to given areas within the domain.

According to a seventh aspect of the present invention there is provided a computer-implemented method comprising receiving historical queue data about queue lengths for a plurality of spaced-apart attractions distributed across a domain, receiving
15 predicted queue data about the queue lengths for the plurality of spaced-apart attractions, receiving information about visitors currently in the domain and, in dependence on the historical and predicted queue data, and the information about visitors currently in the domain, generating incentives for at least some of the visitors currently in the domain to encourage at least some of the visitors to move to given areas
20 within the domain.

The incentives may include vouchers for redemption at retail outlets

25 The given areas may include retail outlets.

According to an eighth aspect of the present invention there is provided a computer program for performing the method of the first, second, third, fourth, fifth sixth and/or seventh aspect.

30 According to a ninth aspect of the present invention there is provided a computer program product comprising a computer-readable medium (which may be non-transitory) storing the computer program of the eighth aspect.

According to a tenth aspect of the present invention there is provided apparatus
35 comprising at least one processor and memory, the at least one processor configured to perform the method of the first, second, third, fourth, fifth sixth and/or seventh aspect.

According to an eleventh aspect of the present invention there is provided a computer-implemented method comprising retrieving, over a wireless network, information relating to times of available slots for least one attraction in a domain, displaying the information relating to times of available slots for the at least one attraction, in
5 response to user input, reserving an available slot for a selected attraction as a reserved slot, in response to user input, presenting a first code containing information about the reserved slot for reading by a validating terminal for allowing access to a queue for the selected attraction, generating a second code which is time dependent and presenting the second code for allowing tracking of the user.

10

Reserving the available slot may comprise sending a request to reserve the available slot to a remote server. The method may further comprise displaying an itinerary comprising at least one intermediate attraction before the selected attraction. The method may further comprise, in response to receiving a message from a remote server,
15 displaying an incentive for encourage the user to move to a target destination. Presenting the first code may comprise displaying the first code on a display. The first code may be a QR code. Presenting the second code may comprise displaying the second code on a display. The second code may be a QR code. The second code may be updated after a given period. The given period may be between 10 and 60 seconds, for
20 example, 30 seconds. The second code may be updated periodically.

According to a twelfth aspect of the present invention there is provided a computer program for performing the method of the eleventh aspect.

25 According to a thirteenth aspect of the present invention there is provided a computer program product comprising a computer-readable medium (which may be non-transitory) storing the computer program of the twelfth aspect.

According to a fourteenth aspect of the present invention there is provided a mobile
30 device comprising a controller, a wireless network interface and a display, the controller configure, to retrieve, over a wireless network using the wireless network interface, information relating to times of available slots for least one attraction in a domain, to display the information relating to times of available slots for the at least one attraction, in response to user input, to reserve an available slot for a selected
35 attraction as a reserved slot, in response to user input, to present a first code containing information about the reserved slot for reading by a validating terminal for allowing

access to a queue for the selected attraction, to generate a second code which is time dependent and to present the second code for allowing tracking of the user.

5 According to a fifteenth aspect of the present invention there is provided a computer-implemented method, comprising, in response to user input, configuring a dynamic gamification management module so as to cause the dynamic gamification management module to find, for a plurality of attractions distributed across a domain, a cost of securing a slot to an attraction and a value of a reward for visiting the attraction, and, in response to user input, configuring an incentive recommendation module so as
10 to cause the incentive recommendation module, generating incentives for affecting behaviour and/or location of visitors.

According to a sixteenth aspect of the present invention there is provided a computer program for performing the method of the fifteenth aspect.

15 According to a seventeenth aspect of the present invention there is provided a computer program product comprising a computer-readable medium (which may be non-transitory) storing the computer program of the sixteenth aspect.

20 According to an eighteenth aspect of the present invention there is provided apparatus comprising at least one processor and memory, the at least one processor configured to perform the method of the fifteenth aspect.

Brief Description of the Drawings

25 Certain embodiments of the present invention will now be described, by way of example, with reference to the accompanying drawings, in which:
Figure 1 is a schematic block diagram of a theme park;
Figure 2 is a schematic block diagram of a system for managing virtual queues in the theme park shown in Figure 1;
30 Figure 3 is a schematic block diagram of a queue management engine;
Figure 4 illustrates an iterative process of predictive analysis and scenario evaluation and policy optimization;
Figure 5 illustrates time slots for a ride and allocatable slots within time slots;
Figure 6 illustrates tokens which can be earned by users and which are used by a queue
35 management engine;
Figure 7 schematically illustrates normal and fast lanes for an attraction;

- Figure 8 illustrates shows for a fast lane;
Figure 9 illustrates generation of voucher;
Figure 10 illustrates a pop-up notification for a voucher;
Figure 11 illustrates a wallet containing vouchers;
5 Figure 12 illustrates a simulated output of a crowd analysis and prediction module;
Figure 13 is a schematic block diagram of a mobile phone;
Figure 14 is a schematic block diagram of modules and sub-modules of a virtual queue app running on a mobile phone;
Figure 15 is a schematic block diagram of a virtual queue kiosk;
10 Figure 16 is a schematic block diagram of modules running on a virtual queue kiosk;
Figure 17 is a schematic block diagram of a virtual queue token terminal provided at an end of queuing area before entering an attraction;
Figure 18 is a schematic block diagram of a module running on a virtual queue token terminal;
15 Figure 19 is schematic block diagram of a computer system which can be used for a virtual queue management console;
Figure 20 is a schematic block diagram of modules running on a computer system used for a virtual queue management console;
Figure 21 is a schematic block diagram of a virtual queue slot prediction model;
20 Figure 22 schematically illustrates a method of dynamic pricing;
Figure 23 schematically illustrates a method of Bayesian optimization for prising scheme searching;
Figure 24 illustrates an example of an itinerary plan displayed on a mobile phone;
Figure 25 schematically illustrates an algorithm architecture for generating an itinerary
25 plan;
Figure 26 schematically illustrates itinerary item selection;
Figure 27 is a schematic block diagram of multi-agent modelling and simulation system;
Figure 28 schematically illustrates searching for a sequence of attractions to be visited;
30 and
Figure 29 illustrates process of using patron and attraction agent behavioural models.

Detailed Description of Certain Embodiments

In the following, like parts are denoted with like reference numerals.

Herein, a system for and a method of predicting and proactively managing queues in an amusement park (or “theme park”) is described. The system and method, however, can be used for managing queues in other environments (or “domains”) in which queues can form, such as, for example, holiday resorts, integrated resorts, recreational villages, hotel complexes, cinema complexes, casinos, convention centers, exhibition centers, 5 cruise ships, transport hubs (such as airports and train stations), tourist attractions (for instance, a part of a city, a historical site, or a place of natural beauty), art galleries, museums, shopping malls or plazas, restaurants and the like. An attraction may be a cinema, a theatre, a concert hall, or other similar form of performance venue (which 10 may be indoors, outdoors or covered). An attraction may be a casino or a part off a casino (such as a table game). An attraction may be a sporting facility such as a racket-sport court, a games pitch, a table (such as a snooker table or table tennis table), a gym, swimming pool, a climbing wall or other area for physical recreational activity. An attraction may be a playground, a play area or a ball pool. An attraction may be a class, 15 such as an aerobics class, a dance class or a yoga class. A domain may comprise two or more different types of attractions.

Introduction

Referring to Figure 1, a simplified view of theme park 1 is shown.

20

The theme park 1 has at least one entrance/exit 2 (“entrance”) and includes, among other things, attractions 3, 4 (such as rides) and facilities 5 (such as retail outlets and food-and-beverage stalls) distributed across the park 1. Some attractions 3 are more popular than other attractions 4.

25

Ordinarily, a patron (or “visitor”) 6 can purchase a ticket (not shown) either before they arrive at the park 1, for example, through a website (not shown), mobile application (not shown) or operating partner (not shown), or when they arrive at the park 1. The ticket (not shown) may be scanned to gain entry to the park 1. Having entered the park 30 1, patrons 6 are free to move through the park 1, choosing attractions 4, 5 to enjoy, following a route 7 (herein referred to as an “unguided route”) of their own choosing. If a patron 6 wishes to enjoy a ride 3, 4, they can line-up in a “normal” line 8 for the ride 3, 4. They may, however, be able to skip the normal line 8 and join a fast lane 9.

35

In the scenario just described, patrons 6 may need to wait (sometimes for as long as an hour or more) before they can go on a ride 3, 4, especially for popular rides 4 and at

busy periods, such as at a weekend. Moreover, the theme park operator needs to ensure that there is sufficient space to accommodate queues. These factors may limit the number of attractions 3, 4 available to patron 6.

5 A virtual queue management system (or “Virtual Queue (VQ) solution”) is herein described which uses, among other things, advance reservation or dynamic VQ reservation, and VQ gamification to dynamically manage reservation of VQ slots in real time. The virtual queue management system is used to predict supply and demand of queue slots in real-time and proactively to manage and/or to influence visitors in the
10 park 1, for example, using incentives.

This can bring benefits to both visitors and park operators. For example, waiting times in physical queues can be reduced or even avoided. Visitors can enjoy a richer, more diverse experience and can benefit from a tailored itinerary and targeted activities.

15 Park operators can also benefit from better control of crowd distributions and optimizing park operation. Using either advance reservation or dynamic VQ reservation, patrons 6 can better plan their time by reserving or registering intention of reserving rides for a preferred time slot before or during their visit to the park 1.

20 As will be explained in more detail hereinafter, VQ gamification 32 (Figure 3) is a backend algorithm engine (or “module”) for dynamically calculating and balancing supply and demand of VQ slots. An itinerary planning module 33 (Figure 3) and an incentive recommendation module 34 (Figure 3) can be used for dynamically diverting patrons to other attractions 3, 4, especially less popular attractions 4, for efficient
25 crowd distribution and providing a more enriched, diversified experience. For example, the patron 6 may follow a plan (or “itinerary”) 10 (Figure 2) specific to the patron 6 following a guided route 11 which may allow the patron 6 to enjoy more rides 3, 4, (even popular ones) and/or spend less time waiting in queues. Moreover, the patron 6 may be provided with one or more incentives to follow the plan 10 (Figure 2).

30 Incentives can be used in different scenarios. An incentive can be used to complement an itinerary plan to encourage patron to follow the itinerary plan. An incentive can be used to conclude any partially/completely accomplished itinerary plan so as to clear and re-balance the total number of VQ tokens distributed. An incentive can be used to
35 promote less popular and/or less crowded areas by giving VQ slots or vouchers to attract crowds away from more popular or more crowded areas.

Patrons can form a group which follows an itinerary. For example, a group may comprise one or more families and/or one or more friends. In the following, the term “patron” also covers a defined group of patrons, for instance, which may be defined by one of the users in an app 22 (Figure 2). The group can break up and reform, and/or
5 individuals can leave and re-join the group. Thus, the group can vary dynamically during a visit to the park.

VQ system overview

Referring to Figure 2, a system 20 for managing virtual queues (or “VQs”) is shown.

10

The system 20 includes a VQ mobile application 22 (“VQ app”) running on mobile phones 23 of patrons 6, VQ kiosks 24 distributed across the park 1, VQ validation terminals 25 and VQ token terminals 26 provided at each attraction 3, 4 and, optionally, retail outlets 5. Some rides need not have VQ validation terminals 25
15 and/or VQ token terminals 26. A VQ token terminal 26 need not be a static or permanently-installed device, but can take the form of a handheld or mobile device. The system 20 also includes a reservation portal 27 for allowing patrons to make advance reservations and a reservations database 28. The reservation portal 27 may be accessed by patrons using their phone 23 or computer 29 via the Internet 12.

20

Referring also to Figures 3 and 4, the system 20 includes a VQ engine 30 (or “AI Algorithm Engine”) which comprises five modules 31, 32, 33, 34, 35 for making predictions and for implementing and managing virtual queues, and a VQ management console 36 for allowing the park operator to manage the VQ engine 30.

25

The system 20 also includes a database 37 for storing relevant data, such as a table 38 storing a current number of tokens 44 (Figure 6) that have been earned by patrons and which can be redeemed.

30

The system 20 may also include environmental sensors (not shown) for making weather observations such as temperature, humidity, pressure, sun, rainfall, wind speed and the like.

Referring to Figure 5, each ride ($n = 1, 2, \dots$) has a fast lane slot schedule 41 comprising
35 a series of time slots 42 (*e.g.*, 30 minutes-long, starting at 09:00 and the last time slot starting at 16:30) and each time slot 42 containing up to N slots 43 (*e.g.*, where N is 15).

The park operator may designate (or “set aside”), for each time slot 42, a given number of slots 43 (“VQ slots”) as being available for reservation prior to arrival, for example, bookable through the advance reservation portal 27 (Figure 2). The number of VQ slots 43 available for reservation prior to arrival can vary from time slot 42 to time slot 42. For example, for the first time slot 42 of the day, 80% of slots 43 may be available for reservation prior to arrival, for the second time slot 42 of the day, 60% of slot 43 may be available for reservation prior to arrival and the a third and subsequent and, for the third time slot 42 of the day, 20% of slot 43 may be available for reservation prior to arrival.

10

After the park has opened, slots 43 are either booked or available for reservation, i.e., available for VQ slot booking depending.

Referring also to Figure 6, whether or not a patron is able to reserve a slot 43 depends on whether they have earned a sufficient number of tokens 44 (“the required number of tokens”) for their desired ride and a given time slot and is determinized by the VQ engine 30.

As will be explained in more detail hereinafter, patrons can earn tokens 44 by, for example, following a suggested itinerary 10 generated by the VQ engine 30 with a view to managing crowd distribution. The number of tokens 44 which can be collected (“the earnable number of tokens”) depends on several factors including the ride and time slot. The earnable number of tokens 44 can vary, for example, from 0 to 50 tokens and the required number of tokens may be, for instance, 100 tokens.

25

The required number of tokens 44 can vary dynamically (*i.e.*, can change with time). Similarly, the earnable number of tokens 44 can vary dynamically.

The system may be configured such that, on the day, when they are in the park, a patron can continually make reservations as long as their previously-reserved slot 43 is fulfilled. Availability of slots for reservation is subject to the determination by the VQ engine 30 and if a patron tries to reserve a slot for a popular ride and/or at a time of high demand, then the VQ engine 30 may operate in a state which recommends or only offers one or more other (less popular) ride(s) and/or at one or more times of lower demand thereby helping to balance supply and demand. Thus, a patron may be limited as to which slot they can reserve and not have unfettered choice of slots (*e.g.*, be able to

35

choose early slots) for popular rides. However, by using itinerary plan function and earning VQ tokens, they are more likely to secure a VQ slot for a popular ride, even ones earlier in the day.

- 5 The system may also be configured such that a patron can hold up to a maximum number of reservations (for example, one, two, three or more reservations) at any given time. The system may be configured to apply rules to determine whether a previous reservation has been fulfilled. For example, a rule may be that a patron can make up to a given number of modifications and/or cancellations in one day (for example, one,
10 two, three or more) before they are prohibited from making a new reservation.

VQ engine

Referring in particular to Figure 3, the VQ engine 30 comprises five algorithm modules 31, 32, 33, 34, 35.

- 15 A dynamic VQ slot prediction module 31 predicts the available VQ slots 43, in particular a number D (Figure 8) of dynamic VQ slots, as the remaining capacity of the lane available for dynamic VQ slot reservation, and waiting time 86 (Figure 12) for the subsequent future time periods by constantly evaluating supply and demand for each
20 attraction 3, 4. This allows patrons 6 to reserve available slots 43 in advance so that they get access to the fast lanes 9 to avoid long queue in the normal lanes 8.

- The VQ slot prediction module 31 receives input data 171 from the crowd analysis and prediction module 35 which is updated in real-time. The input data 171 includes visitor
25 factors 175 (Figure 21) such as the number of people currently in the park, temporal features 176 (Figure 21) such as the number of people in each queue which varies in real time and price factors 177 (Figure 21), such as price (expressed in terms of tokens), which also vary in real-time.

- 30 The output 172 from the VQ prediction module 31 includes an average number of people 178 in a corresponding queue lane (fast or normal) in future periods of time. The output 172 also includes derived information such as an estimated waiting time in the lane, which corresponds to a duration of time between tap-in at the VQ validation terminal 25 (Figure 7) and tap-out at the VQ token terminal 26 (Figure 7) and which is
35 based on a total capacity L (Figure 8) of a lane.

The VQ slot prediction module 31 generates the set of updated outputs at a configurable interval (such as, for example, every 15 minutes) according to current values of inputs. Dynamic information (*i.e.*, real-time information) is factored in at a discrete time interval.

5

A dynamic gamification management module 32 generates a VQ token allocation scheme 207 for each attraction for specific time slots 41 dynamically according to the forecasted VQ slots 43 for each attraction 3, 4. VQ tokens 44 can be exchanged for vouchers to be redeemed at respective facilities or attractions or VQ slots 43 of rides to
10 access the fast lane.

Referring also to Figure 6, there are two prices 45₁, 45₂ in terms of the number of VQ tokens 44 to be assigned to each attraction. A first price 45₁ is the cost that is needed to redeem the VQ slot 43 of a desired ride (*i.e.*, a popular ride 4) during the corresponding
15 slot period of time. A second price 45₂ is the award that a patron can collect by visiting the attraction 3, 4 as part of a recommended itinerary 10. This module 32 generates a set of updated outputs 34 in real-time upon update of the VQ slot prediction results 172. It can have the same configurable interval (*e.g.*, every 15 minutes).

20 Referring still to Figure 3, the itinerary planning module 33 runs an itinerary planning algorithm 221 (Figure 25) and offers patrons a recommended plan 10 for the next few hours to secure a VQ slot 43 for a desired ride or non-ride attraction for a desired time slot. Itinerary planning inherently manages crowd distribution by redirecting patrons to the appropriate attractions before they avail their desired choice. Itinerary plan is an
25 online service provided with the mobile app 22 that provides prompt response upon request. Parameters used by the module 33 include an attraction basic and relation data 229 (Figure 25) and candidate attractions 227 (Figure 25). These parameters can be updated at a configurable fixed interval, for example, to keep in time with updates from first and second processes 226, 228 and so as to keep the planning algorithm 221
30 (Figure 25) up-to-date.

Referring to Figure 9, the park operators can create vouchers 51 via an interface 52 through the management console 36 (Figure 19) which can then be allocated by the VQ engine 30 to patrons.

Different types 53 of vouchers 51 can be defined having different properties or parameters specifying time windows 54, outlets 55 and voucher-related values 56.

5 Selecting a voucher type 73 can set other properties 57, such as customer type 57₁ (e.g., adult or child), usage type 57₂, number of times the voucher can be used 57₃, whether the voucher is monetary or non-monetary 57₄, a voucher amount 57₅, face value 57₆, consideration value 57₇ and intercompany rate 57₈, i.e., amount for intercompany charge to campaign owner. Properties 57 may have default values which can be adjusted by the user.

10

Referring to Figure 10 and 11, vouchers 51 sent to the mobile devices 23 (Figure 2) and displayed as pop-ups 58 on the display 82 and can be stored in a wallet 59.

15 Each voucher 51 includes the information such as merchant, voucher category and corresponding information, time constraints and any other terms and conditions.

Referring again to Figure 3, an incentive recommendation module 34 utilizes vouchers 51 (Figure 9) and other incentives to enhance patron experience and influence crowd distribution. There are three ways incentives 51 are given and utilized. First, incentives 20 51 can be distributed to attract patrons to non-ride attractions or facilities 5, such as food and beverage stall, retail malls and the like, to divert them from over-crowded areas. Incentives 51 can take the form of vouchers, free gifts and the like which would. Secondly, incentives 51 are used as rewards to encourage patrons to continue use the VQ app 22 and follow the recommended itinerary plans 10. Thirdly, incentives 51 can 25 be provided as compensation to encourage patrons to continue reserving VQ slots 43 in cases they are not allocated their desired VQ slots 43.

Referring also to Figure 4, a crowd analysis and prediction module 35 collates data from a range of sources and predicts overall crowd movement and distribution in the 30 following short period of time (e.g., one hour), and evaluates scenarios for strategy optimization (i.e., how will the crowd distribution evolves with the current VQ token assignment, itinerary plan and incentives given).

Each of these modules 31, 32, 33, 34, 35 will now be described in more detail.

35

Dynamic VQ slot prediction

Referring also to Figures 2, 5, 7 and 12, rides 3, 4 are typically provided with two lanes 8, 9 for queuing, namely a normal lane 8 and a fast lane 9. The fast lane 9 is a priority lane with expected shorter waiting time, with a shorter physical queue length and a larger throughput to join the ride, compared to the normal lane 8.

5

Users (*i.e.*, patrons) are entitled to a quota, *i.e.*, specific number, of advance reservations for different categories of rides 3, 4 before arrival.

10 Once they enter a park 1, a user can manually reserve VQ slots 43 of rides 3, 4 limited as outlined hereinbefore (*e.g.*, if they have fulfilled a previous slot) and the VQ system 30 has available capacity at the time of request of reservation for that ride and that time slot.

15 By using itinerary plan and by indicating a desired ride (usually popular ones without available capacity for manual reservation), a user can gain a VQ slot of their desired ride at a later time, which is determined by the VQ system 30, upon fulfilment of the recommended itinerary, in other words, they visit recommended attractions during corresponding time period on the itinerary plan.

20 If they fail to fulfil the complete itinerary plan (herein referred to as “partially completion”), the VQ system 30 automatically exchanges VQ tokens 44 earned so far into incentives (such as vouchers or a VQ slot of a less popular ride) and clear (*i.e.*, re-sets) the number of VQ tokens. If, however, the user successfully completes the itinerary plan, the user is awarded with a VQ slot 43 of their desired ride 3, 4. They
25 may be awarded with other vouchers, depending on the number of earned VQ tokens 44 and the required number of tokens for the desired ride.

To get access to the fast lane 9, patrons reserve a time slot 43 through the VQ app 22. To ensure that walk-in customers have sufficient timeslots availability, not all capacity
30 is released for advance reservations prior to their arrival to the park. The ratio of the number of slots which can be reserved in advance to the number of slots which are not reservable in advance can be configured in the management console module 36 (Figure 2).

35 The dynamic VQ slot prediction module 31 predicts available VQ slots 43 (with a total capacity D) and a waiting time for the subsequent future time periods by constantly

evaluating the supply versus demand for each ride or non-ride attraction. This allows patrons to reserve these available slots 42 in advance so that they get access to the fast lanes 9 to avoid long queue in the normal lanes.

- 5 As the fast lane 9 is designed for much shorter queueing time, the capacities of the fast lanes L per ride over different time periods will be underutilized if they are only fulfilled with the VQ slots through advance reservations 43.

- 10 A patron can reserve a limited number of (advance reserved) VQ slots 43, subject to availability, upon purchase of tickets before their visit to the theme park 1. However, the best in-advance plan may not fully cover the capacities of fast lanes 9 due to the dynamic behaviour of patrons. For example, patrons may change their plans or simply miss their VQ slots 43. In this case, the balanced VQ slot capacity D can be dynamically monitored and predicted for each ride and awarded to users as incentives while they
15 participate in the virtual queue solution, so that the full capacities of fast lanes 9 over all time period can be utilized to reduce the total waiting time of patrons.

- Two terminals 25, 26 are used, namely a VQ validation terminal 25 at the beginning 46₁ of the lane 9 and a VQ token terminal 26 at the end 46₂ of the lane. The VQ validation
20 terminal 25 has a reader 47 for reading a QR code 48 presented by the VQ app 22 or paper ticket (not shown) and the VQ token terminal 26 has a reader 49 for reading a QR code 50 presented by the VQ app 23. In some embodiments, the VQ token terminal 26 has a display for presenting a QR code for reading by the mobile phone 23.

- The QR code 50 is linked to the ticket of the user on the app 22 and so serves as a
25 personal identifier. The QR code 50 refreshes, for example, every 30 seconds, but is still linked to the specific individual ticket at the backend. Dynamically refreshing the QR code 50 helps to prevent people from taking a screenshot and sending it to others for them to scan.

- 30 At the start 46₁ of the lane 9, a patron 6 taps-in when they enter the lane 9 by presenting the QR code 48 to the terminal 25 for scanning in. The patron taps-out at the end 46₂ of lane 9 using the phone 23 by presenting a dynamically-changing QR code 50 to the VQ token terminal 26 to collect VQ tokens. Tokens earned by the patron in this way are updated in a database 37 in the backend and are exchanged into corresponding
35 incentives (such as vouchers or VQ slots) when their itinerary plan is concluded.

Tapping-in and tapping-out can be used to predict an expected waiting time for both normal and fast lanes 8, 9 and a number of VQ slots 43 available. These can be of use to both patrons and the park operator. Patrons can view these results through the VQ mobile app 22 and reserve their slots in advance, while the operator can view the results through the management console 36.

The predicted results are evaluated using the factors set out below, namely:

- current and planned ride status for historical/real time
(for example, capacity, cycle time/throughput, popularity);
- daily reservation status by ticket, by ride and by time slot;
- ticket QR code, tap-in timestamp (at VQ validation terminal 25),
tap-Out timestamp (at VQ Token Terminal 26) in real time;
- historical VQ slot quota, number of patrons at fast/normal lanes, waiting time at
fast/normal lanes, historical reservation slots at corresponding time slot; and
- a buffer range reserved for the operation team.

Both historical and real-time data can be used to build a precise prediction model. Methods such as time-series analysis, multi-variate analysis and recursive neural network, such as, long short-term memory (LSTM) can be used to build the prediction model. These can achieve high accuracy in a time-series prediction and/or provide robustness with multiple data sources. These are methods that can be applied for time-series prediction. As will be explained in more detail hereinafter in relations to VQ slot prediction, machine learning methods can be used.

25

Dynamic gamification management

The dynamic gamification management module 32 generates a VQ token allocation scheme for each attraction for specific time slots dynamically according to the forecasted VQ slots for each attraction. VQ tokens are used for internal calculations in the algorithm engine, and patrons are rewarded with VQ slots 43 (Figure 5), vouchers 51 (Figure 9) or both through the VQ mobile app 22.

For locations without VQ terminals 25, 26, such as food-and-beverage stalls and shops 5, manually-entered data regarding crowdedness (or “crowding”) in a location may be periodically supplied by a member of the operations team into the system and batch updates of voucher utilization through the integration with the e-voucher validation

system. These manual inputs can be automated using sensors and other ways of collecting data (*e.g.*, based on image recognition using cameras).

As will be explained in more detail later, through the itinerary planning function 94 (Figure 14) of the VQ mobile app 22, patrons can choose, at that time, a desired VQ slot 43 for a ride at a given time, and the app 22 can recommend a list of activities in subsequent time periods as an itinerary plan 10.

There are three ways for patrons to get the voucher as a result of dynamic gamification management. First, when the patrons complete the recommended itinerary, they secure their desired VQ slot 43. To show appreciation for their efforts and for contributing to the overall crowd distribution, the system awards the patron with vouchers as a bonus (*i.e.*, in addition to the VQ slot 43). Secondly, when the patron follows the recommended itinerary, but could not secure their desired VQ slot 43, as compensation and also to encourage them to use an itinerary again, the system appropriately awards the patron with vouchers. Thirdly, if the park is overcrowded, the system allows the operation team to release vouchers to redirect patrons to other facilities, such as dining or shopping, immediately.

The allocation scheme adjusts according to the dynamic forecast of supply/demand of the VQ slots 43, which is the output from the dynamic VQ slot prediction module 31. The demand for VQ slots 43 may vary as a complex interaction consequence of the patron crowd which is continuously analysed and prediction about it made in the crowd analysis and prediction module 35.

Factors which influence the VQ token allocation scheme include, real-time and predicted normal and virtual queue lengths and VQ slots, estimated demand of a ride at particular time slot using historical and time-series data, historical VQ token allocation scheme for a corresponding ride and time slot, and other factors such as ride status, weather condition *etc.*

Dynamic itinerary plan recommendation

Referring again to Figure 3, the dynamic itinerary plan recommendation module 33 offers patrons a recommended plan for the next few hours (*e.g.*, two hours) to secure a VQ slot for a desired ride for a desired time slot. Itinerary planning inherently manages crowd distribution by redirecting patrons to suitable attractions by taking into

consideration queuing time, promotions, popularity, crowdedness and the like, before they avail their desired choice, i.e., enjoy their chosen ride. This works closely with dynamic VQ prediction and dynamic gamification management modules 31, 32 to generate several recommended itineraries with a pool of candidate
5 activities/attractions in subsequent time periods. If patrons follow the recommended plans, they are allocated VQ slots for their desired ride time slots.

If patrons cannot reserve their desired slots through advance reservation (due to lack of availability of the slots) they choose itinerary plan recommendation by indicating their
10 desired ride slot. This process includes the following steps. First, a patron chooses their desired ride. Secondly, itinerary plan recommendation module 33 offers a set of attraction choices before they can enjoy the chosen ride and slot. The patron follows the itinerary plan, and the system allocates the VQ Slot for the desired ride and slot.

15 Factors which influence itinerary planning include current estimated waiting time and/or attendance time for attractions, which rides the app user has indicated an interest and their corresponding predicted VQ slot quotas at different time slots, which would integrate with the dynamic VQ slot prediction and dynamic gamification management modules 31, 32, the last checked-in location and current estimated
20 location of the app user, and currently-available reward vouchers to integrate with the incentive recommendation module 34.

Multi-constraints optimization methods, such as evolutionary algorithm, genetic programming, mixed integer programming, dynamic programming and reinforcement
25 learning, can be used to recommend patrons with customized constraints and preferences, while methods such as collaborative filtering from the classic recommender system approach can be used to enhance the performance with more historical data.

30 *Incentive recommendation*

Referring still to Figure 3, besides dynamic gamification management, another approach to divert patrons to more attractions (including those inside and outside the theme park) is through incentive recommendation. This mechanism allows patrons to be diverted by offering incentives 51 to a targeted audience. The incentive 51 may
35 include meal vouchers, limited-time discount vouchers for participating merchants, and other activities and promotions reserved for the operations team. Patrons can

receive incentives 51 in the VQ mobile app 22. They can use this voucher 51 at the participating merchants by scanning a QR code when paying. Utilization of vouchers captured at the existing E-voucher system are synchronized in batches (*e.g.*, every 10 mins) to the VQ system backend system 30 and vouchers 51 are distributed to patrons in the aforementioned three cases, and their VQ tokens will be deducted accordingly.

The incentive recommendation module 35 generates an incentive distribution scheme 71 for assigning an available pool of incentives 51 to different visitors that is likely to maximize overall utilization of the vouchers and to improve the overall crowd distribution. This module 35 takes the form of an online service at the backend 30 to enable real-time request of incentive exchange from VQ tokens in any of the three cases that trigger incentive utilization hereinbefore described.

Factors which may influence incentive recommendation include non-personally identifiable demographic information, such as age, gender, nationality, residential status and the like, individual footprint of activities, such as VQ gaming, purchase, voucher and the like, real-time and predicted normal and virtual queue lengths, and historical utilization rate and ratio of incentives issued. A user profile (not shown) can be retrieved through the VQ app 22 which includes height, ride preference, food preference, thrill level of rides and other preferred area of interests. Alternatively, a user profile can be retrieved through a resort membership system, which may provide more information and so provide even better customised incentives.

Incentive recommendation employ recommendation-based node clustering, such as k-means, collaborative filtering, and transfer learning. Historical data can be used for training and performance can be refined as more data is collected.

Crowd analysis and prediction simulation

Referring still to Figure 3, the crowd analysis and prediction module 35 receives outputs 10, 34, 71. 172 of the other modules 31, 32, 33, 34 and simulates overall patron crowd movement and distribution in the premises of theme park in the subsequent short period of time (*i.e.*, in the near future). This module 35 runs on updates of the inputs of other modules and may provide output at a fixed configurable interval (*e.g.*, every 15 minutes) to synchronise with the frequency of update of the other modules.

From a modelling and simulation perspective, the theme park is a complex system where patrons autonomously move around based on their itineraries, yet interact with each other. Furthermore, patrons also interact with different attractions (*e.g.*, queueing time and throughput of different rides, merchants with incentives *etc.*). These interactions are complex, but an emerging, overall outcome of these interactions can be found so as to generate useful indices to reflect the status of the theme park. The generated indices, in turn, feedback to other modules as guidance for refining their operation. This module 35 can conduct “what-if” analysis, scenario studies to evaluate different status and indices according to the updated data information. In order to consider the complex interactions of entities in the theme park in a comprehensive manner, agent-based modelling and simulation can be used. Data-driven calibration of the model can be used. It can be formulated as an optimization problem: to find the set of parameters and rules that can most closely reproduce the observed status. Evolutionary Algorithms (EA) can be used.

Referring to Figure 12, an illustration of the output of the crowd analysis and prediction module 35 is shown for a simplified scenario.

The output includes a map 61 illustrating rides 3, 4 in the theme park 1 and information 62, 63 regarding rides and waiting times. A target 64 was set to minimize the average waiting time of patrons, namely 29 minutes as of the simulated scenario at a time 11:23, and to maximize the average number of rides 65 patron can visit, namely 3.13 rides as of the simulated scenario at 11:23, by assigning different tokens to rides at specific time periods dynamically so as to attract patrons more evenly to different rides, on other words, to cause more even waiting time of all rides as illustrated in the bar chart 66.

The simplified scenario and behavioural rules are set out below.

Scenario:

- Two entrances with equal probabilities to enter (the probabilities are tuneable);
- 19 rides with designed timing information;
- simulation starts from 08:00 (i.e., 8 am in the morning);
- different rides can have different rewards to be collected at different time,
- simulating the reward collection mechanism in the gamification solution; and

- there are fast pass holders and normal day pass holders, with a ratio of 1:9 (the ratio is tuneable)

Other scenarios may be used.

5

Rides:

- each ride operates according to designed timing information provided. Key information used includes: theoretical hourly ride capacity as the hourly throughput, dispatch capacity, dispatch interval, normal lane capacity, fast lane capacity, and admission ratio (fast: normal) = 4:1 (the ratio is tuneable);
- at the end of each (riding) cycle, the ride will clear the batch of patrons and take a new batch of them from the fast and normal lanes according to the empirical ratio of 4:1. If not enough patrons are waiting in the fast lane, the remaining will be taken from normal lane to fill the dispatch capacity;
- each ride will be initialized with a random popularity to attract patrons to visit (the popularity is tuneable); and
- VQ Tokens can be manually added to each ride to mimic the dynamic supply of VQ tokens in the VQ gamification system to attract patrons.

20 Patron behaviour:

- there are two types of patrons: hardcore (intense) and normal. For hardcore visitors, they target to play as many (different) rides as possible (the number of rides is tuneable);
- a patron visits one ride after another, with random resting time in-between, e.g., normal distribution around 15 mins (the rest time is tuneable);
- the choice of the next ride is based on multiple factors, which currently include: ride popularity, distance from the current location, estimated waiting time of the ride, and the VQ Tokens to be earned;
- the longer the waiting time, the less “resting and shopping” time from the interval of rides.

30

Different patron profiles can be reflected by using different parameter configurations. For example, model parameters θ_1 , θ_2 , θ_3 (Figure 29) can be varied.

35 Even with a simplified scenario and model, the simulated results reflect the complicated interaction among different entities in real-time.

Functions and Components

Referring again to Figure 2, the front-end touchpoints for users (*i.e.*, the patrons) are the VQ mobile application 22, VQ kiosks 24, the VQ validation terminals 25 (or “EPOS information terminals”) and VQ token terminals 26. The management console 36
5 allows the park operation team to perform management and configuration tasks of the VQ system (“VQ solution”).

The AI Algorithm Engine 30 correlates data from VQ user touch points 25, 26 and management console 36, as well as existing systems, such as ePOS and integrated
10 circuit card (ICC) systems, to perform continuous analysis of collected data and to provide predictions and recommendations.

Mobile device and VQ mobile application

Referring to Figure 13, the mobile phone 23 includes a processor-based controller 81, a
15 display 82, user input devices 83, a microphone 84, speakers 85, memory 86, cameras 87, a wireless network interface 88 to a mobile network (*e.g.*, 4G, 5G *etc.*) and wireless network interface(s) for short-range network(s) 89 (such as, BlueTooth, WiFi, NFC/RFID *etc.*) The display 82 and at least some of the user input 83 may be integrated in the form of a touch screen.

20 Referring to Figure 14, the VQ mobile app 22 includes several functions (or “modules”) including a VQ slot reservation module 91, an advance reservation module 92, a gamification module 93, an itinerary planning management module 94 and an incentive recommendation management module 95.

25 The VQ slot reservation module 91 includes several sub-modules including VQ information sub-module 96 which can present a ride wait time and availability of VQ slot to the user, a VQ notification/reminder sub-module 97 which can notify the patron of reserved VQ slots and provide reminders about the reserved VQ slots, a VQ slot
30 reservation sub-module 98 for allowing the patron to reserve a VQ slot, a VQ slot validation sub-module 99 for presenting a VQ QR code 47 (Figure 5) at a VQ validation terminal 25 (Figure 7) to validate a reserved slot which enables the patron to use the fast lane 9 (Figure 7) after verification and a VQ status sub-module 100 for allowing the patron to check reserved VQ slots and activity/ride history.

The advance reservation module 92 includes a ride advance reservation sub-module 101 which allows a patron to reserve a ride in advance prior arrival at theme park and VQ slot in the theme park and a reservation management sub-module 102 which allows a patron to manage advance reservation such as view, edit or cancel.

5

The gamification module 93 includes an earn VQ token sub-module 103 for generating a QR code 50 (Figure 7) which can be presented to a VQ token terminal 26 (Figure 7) to earn a VQ token. The QR code 50 is repeatedly updated, for example, every 30 seconds, to make it harder to copy and be used by another patron. The number of VQ tokens is
10 calculated by AI algorithm engine 30.

The itinerary planning management module 94 includes an itinerary planning sub-module 104 which allows a patron to select a desired ride and follow the recommended itinerary to secure a VQ slot for the desired ride at desired time slot. Itinerary
15 execution (*i.e.*, commitment) progress is tracked through tap-ins and tap-outs of recommended attractions at corresponding period in the itinerary plan

The incentive recommendation management module 95 includes a view voucher sub-module 105 for allowing the patron to view a received voucher for, for instance, a food-and-beverage unit, retail outlet or the like, a notification/reminder sub-module 106 for
20 receiving notifications about received vouchers, used vouchers and expired or expiring vouchers, and a utilize voucher sub-module for allowing a patron redeem a voucher at a food-and-beverage unit, retail outlet or the like by displaying the voucher code at a cashier within validity period.

25

VQ kiosk

Referring to Figure 15, the VQ kiosk 23 includes a processor-based controller 111, a display 112 (for example in the form of a 32-inch/81-cm display), user input devices 113, a microphone 114, speakers 115, memory 116, camera or scanner 117, a network
30 interface 118 and wireless network interface(s) 119 for short-range network(s) (such as BlueTooth, WiFi, NFC etc.), and a printer 120. The display 112 and at least some of the user input devices 113 may be integrated in the form of a touch screen.

Referring to Figure 16, the VQ kiosk 23 includes several functions (or “modules”) including an account tagging module 121, a VQ slot reservation & QR code printing
35 module 122 and a VQ status module 123.

The account tagging module 121 can be used for RFID and QR codes. For a non-mobile application user, a patron can bind their ticket to a VQ account by scanning a QR code on ticket or RFID at a VQ kiosk 23.

- 5 In relation to the VQ slot reservation & QR code printing module 122, a patron may have made reserved VQ slots or advance reservation. In case the patron does not have a phone, the theme park can provide a facility for the patron to print out the QR code so that they can enjoy the benefit of the VQ solution.
- 10 Using the VQ status module 123, a patron can check VQ activities, such as VQ tokens, advance reservation, VQ slot reservation, itinerary, *etc.*) by scanning a QR code or RFID.

VQ token terminal

- 15 Referring to Figure 17, the VQ token terminal 26 includes a processor-based controller 124, a reader 49, memory 125 and network interface 126. The terminal 26 is installed at a boarding area of a ride and is used to read a QR code 50 (Figure 7) which is periodically generated by the VQ app 22 (Figure 7). As explained earlier, the terminal 29 can take the form of a portable or hand-held device, such as a mobile phone or tablet
- 20 computer.

- Referring also to Figure 18, the VQ token terminal 26 runs a dynamic QR code reading module 127 which reads a QR code 50 (Figure 7) and transmits an identifier to the backend 30. A patron can present the dynamic QR code using the VQ mobile app 22
- 25 (Figure 7) and earn VQ tokens.

VQ management console

- Referring to Figure 19, a computer system 140 is shown. The computer system 140 comprises at least one processor 141, memory 142 and an input/output interface 143
- 30 interconnected by a bus system 144. The system 140 includes a graphics processing unit 145 and a display(s) 146. The system 140 includes user input device(s) 147 such as keyboard (not shown) and pointing device (not show), a network interface 148 and storage 149 for example in the form of hard-disk drive(s) and/or solid-state drive. The storage 149 stores software or a code 150 for providing management console functions.

Referring also to Figure 20, the VQ management console 36 allows theme park operation team to control the VQ solution and configure the system according to operational needs to improve efficiency of theme park operation.

- 5 The VQ management console 36 includes several functions (or “modules”) including a VQ slot management module 151, a gamification management module 152, a gamification module 93, an incentive recommendation management module 153 and a crowd analysis and prediction module 154.
- 10 The VQ slot management module 151 includes a VQ ride selection and settings sub-module 155, an overriding sub-module 156, a VQ slot rules setting sub-module 157 and an information dashboard 158. The ride selection and settings sub-module 155 can be used to select a ride to enable VQ solution and set recommended settings based on AI algorithm computation result and expert experience. These setting may include VQ slot
- 15 opening hours, for example half an hour after theme park open and one hour before theme park closed, etc. The operation team can override the recommended settings when necessary, for example, in case that a ride break down, there is bad weather, *etc.*, and to inform patrons via VQ mobile application 22 (Figure 2). The overriding sub-module 156 allows the operator to take control and override computed results. The VQ
- 20 slot rules setting sub-module 157 allows an operator to define a set of business rules for a VQ slot. Rules may include, for example, that a patron cannot select more than three popular rides, or select consecutive popular rides, set a maximum number of VQ slots, and the like. The information dashboard 158 can provide a visual insight at a moment in time (or a “snapshot”) on utilization of advance/VQ slots per ride, in park attendance
- 25 numbers, standby queue time per ride and the like.

The gamification management module 152 includes gamification ride selection and settings sub-module 159, a gamification score computation sub-module 160, a gamification status view sub-module 161 and a patron view sub-module 162.

- 30 The gamification ride selection and settings sub-module 159 can be used to enable gamification for a selected ride. Recommended settings are applied based on AI algorithm computational results and expert experience including VQ slot opening hours, when to stop the gamification function, *etc.* The operations team can override
- 35 the recommended settings when necessary, for example should a ride break down, if there is bad weather, *etc.* and inform the patrons via VQ mobile application 22 (Figure

7). The gamification score computation sub-module 160, based on an AI algorithm model, can compute the number of VQ tokens of each ride. The gamification status view 161 can be used to check how many VQ tokens have been issued and reserved for each ride. The patron view sub-module 162 can be used to an at-a-glance view of
5 tokens earned and redeemed by each patron.

The incentive recommendation management module 153 includes an incentive recommendation settings sub-module 163, an incentive recommendation computation sub-module 164 and an incentive redemption view sub-module 165.

10

The incentive recommendation settings sub-module 163 can be used to enable incentive recommendation for selected place of interest (POI). The recommended settings are applied based on AI algorithm computational results and expert experience, including POI opening hours, when to stop the function, etc. The operation team can override the
15 recommended settings when necessary. The incentive recommendation computation sub-module 164, based on AI algorithm model, can compute the number of vouchers and voucher value to be issued, whom to be receive vouchers. It can be integrated with an existing e-voucher management system. The incentive redemption view sub-module 165 can be used to check voucher redemption records for a patron.

20

The crowd analysis and prediction module 154 includes an existing patron heat map sub-module 166 and a crowd trend forecasting sub-module 167. The existing patron heat map sub-module 166 calculates the number of patrons at each ride or point of interest from multiple data inputs and the crowd trend forecasting sub-module 167 can
25 forecast crowd trend in the defined forecast period (for example over the next two hours, next day, next week) by using historical data and current patron data.

AI algorithm engine

Referring again to Figure 3, the AI theme park algorithm module 30 is the core engine
30 behind the applications 31, 32, 33, 34, 35.

The VQ slot management module 31 estimates the current/next period of patron's amount with the segmentation of advance reservation/VQ slot holder and normal ticket holder, using AI and machine learning, to dynamically compute where spare waiting
35 slots can be generated and the corresponding amount for patrons to redeem and

reserve in order to decrease the waiting time for patron in physical queues, and leave them for other activities. VQ slot will update automatically based on real-time data.

The gamification management module 32 diverts patrons to rides having shorter
5 waiting times to improve patron experience with dynamic gamification score. In this
function, rides are dynamically assigned with different required/awarding gamification
score based on their popularity and crowdedness calculated by an AI algorithm. As
hereinbefore mentioned, the gamification score can be used to redeem the VQ slot of
rides at different time period (with different amount) as motivation for patrons to earn
10 such scores. They can go to less crowded rides to earn gamification score. As a result,
patrons are shifted away from popular rides to other locations according to the dynamic
balancing of the gamification score.

The incentive recommendation management module 33 aims to divert patrons to other
15 facilities such as shops, restaurants and food kiosks, and to increase the total
revenue by encouraging spending with the provided incentives such as vouchers. AI
algorithm will generate the distribution policy through different channels (e.g.,
broadcast on big screen, broadcast in App or push to App users) of different incentives
(e.g., vouchers of different types and amount) to different user segments in the most
20 efficient way so as to divert patrons away from popular rides and generate revenue at
the same time.

The crowd analysis and prediction module 35 calculates the number of existing patrons
in the theme park, the crowdedness by rides, and forecast the crowd distribution across
25 different rides/POIs along time of a day.

VQ slot prediction

Referring to Figure 21, a model 170 used for VQ slot prediction in the VQ slot prediction
module 31 is shown, together with model inputs 171 and model output 172.

30

The model inputs 172 include fixed parameters 173 (or “spatial features”) such as
capacity of rides, capacity of queues for each ride (may switch VIP queue and normal
queue), spatial layouts of rides (such as distance and connectivity between rides), slot
duration and arrangement for each ride. The model inputs 172 include seasonal and
35 weather factors 174 such as season, weekday or weekend, whether it is a holiday or not,

time (such as morning, midday, afternoon and so on), and weather (such as sunny, rainy, cloudy and so on).

5 The model inputs 172 include visitor factors 175 such as number of people in the theme park (which may be expressed in terms of a time series), visitor's features (such as gender, age, reserved or not, consumption capacity and so on). The model inputs 172 include historical factors 176 (or "temporal features") such as number of people in each queue for each ride (time series), and price factors 177, such as promotion efforts, price, VIP qualification

10

The model output 172 consists of a number of people 178 in the target queue in the future with different lead time.

15 The model 170 is used to forecast the number 178 of people in each queue for each ride with different lead time. Models 170 are built for each queue each ride and each lead time.

20 The model 170 employs machine learning algorithms 179 and uses feature engineering module 180, a smoothing module 182 and a regression module 183 to prepare data for training the machine learning algorithms 179. Model structures for each ride are the same, while model parameters are calibrated together through multi-task multi-view learning 184. The model 170 employs an AI algorithm platform 185, such as Alibaba (RTM) Cloud PAI, and a big data platform 186, such as Alibaba (RTM) MaxCompute.

25 The output 178 of the module 170 is the estimated average number of people in each queue (fast and normal) at selected rides of interests (*e.g.*, all rides) in the future slots periods. Thus, it can derive the average waiting time of normal and fast lane, and remaining capacities of each lane. For fast lane, there are predefined slots (*e.g.*, every 30 minutes), but for normal lane, there is no constraints of slots (as patrons can walk in
30 at any time). In some embodiments, the same slots are used as period of time to calculate the average number of people for both normal and fast lanes of a same ride.

35 The machine learning algorithm 179 take the form of discrete time series regression, which is a supervised-learning regression problem. That is, based on historical values of the average number of people in both lanes of a ride during corresponding slots and their correlation with feature values 173,174,175,176, 177 considered. The future

average number of people is predicted in both lanes of a ride from the current time. Models such as decision-trees (such as random forest) or stochastic gradient descend can be used due to their efficiency and accuracy. In addition, models such as decision-trees are good for feature engineering (to analyse feature importance) and their
5 interpretability.

In multi-task learning 184, machine-learning tasks for different rides can be considered to be similar. The regression model 179 that is effective for one is likely to predict the outputs of other rides as well. Even if a separate supervised-learning model is trained
10 individually for each ride, they should share similar model parameter values. Based on the assumption mentioned above, supervised-learning models can be built with training data from different rides to train the model together and with dedicated ride's training data to get subtle emphasis of parameters for that particular ride. An
15 additional benefit besides using less data for faster training is, as mentioned above, that multi-task learning can apply a trained model using mixed data of other rides to predict the values of a ride that may lack of part of the inputs data 171. For example, at some rides, the number of people might not have been measured and/or might not have been manually input from a controller center.

20 The regression model can be considered either from a spatial perspective, which considers how a patron is likely to transit from one attraction to another due to the spatial proximity and connectivity, or from a temporal perspective, which considers the historical time-series features of one ride on the average waiting time of both lanes in the following periods. By building both features (*e.g.*, spatial layout under fixed
25 parameters 173 and number of people in each queue under temporal features 176), the model is comprehensive to captures different views simultaneously.

The model 170 considers two views, namely spatial and temporal views.

30 In the spatial view, the number of people in the nearby rides are considered and generally two effects are considered. One effect is that the degree of crowding in nearby rides may affect the number of people in the target ride in the future. The other effect is that when people finish one ride, they may transfer to other rides nearby.

35 In the temporal view, the cumulating rate of people in the target queue is considered.

Apart from the two views, two adjustment procedures are applied. One adjustment considers the effects of seasonal factors and price factors. The other adjustment considers the total number of people in the park to provide reasonable forecasting results. The first adjustment avoids using discrete variables in the model and helps
5 improve the accuracy and interpretability. The second adjustment avoids unreasonable forecasting results of total number of people and further improves the forecasting accuracy.

Dynamic pricing method for crowd management and visitor experience enhancement

10 A general definition of dynamic pricing is to adjust a product's price dynamically based on the match between market's demand of this product and the purchasing power of the buyers. In the context of queue management system in a theme park (or other similar environment), the price of an attraction is symbolized by VQ tokens and represents the operational status of the attraction, *i.e.*, the real-time demand. Thus,
15 dynamic pricing here determines VQ token assignment for both attractions and rewards for patron's activities (*e.g.*, itinerary) to visit these attractions. With VQ tokens earned through these activities, patrons can redeem different incentives such as VQ slot of fast lane for popular attractions or vouchers. The dynamic pricing here achieves a balance between crowd distribution driven by patrons' activities and patrons'
20 experience.

In so-called "black-box optimization", a black-box function $f(X)$ maps inputs X to output Y , without a known analytical expression for the mapping function f . Black-box optimization is a process of optimizing the function f over some set of desired
25 hyperparameters to be search such that the mapping function accurately provides a value of Y from sampling X .

Gaussian process optimization is a non-parametric method that use Gaussian process (GP) prior to applying regression analysis on the observed data.

30

Bayesian optimization is a black-box optimization method for deriving the optimization (maximum or minimal) without known analytical function. Bayesian optimization incorporates prior belief about function f and updates the prior with samples drawn from f to get a posterior that better approximates the function f . A Gaussian process
35 model can be used for approximating the objective function. Bayesian optimization also

uses an acquisition function that directs sampling to areas where an improvement over the current best observation is likely.

In theme parks, there are various zones and suggested routes based on the design of themes. Each zone contains attractions, facilities and other activity service provisions. Unlike traditional leisure parks, guests of theme parks want an immersive experience involving deep interactions with rides, shows and other services. Different attractions with different themes and design may also attract different groups of visitors, *e.g.*, thrilling type of roller-coasters or family-oriented shows. A theme park provides a rich diversity of experience to the guests, and usually popular attractions are extremely crowded with very long waiting times. In order to improve guest experience, a VQ solution provides information (such as waiting time of attractions), guidance (such as itinerary plan), and rewards (such as fast lane VQ slot, which is a virtual reservation of an attraction during a specific period of time when guests can experience in the attraction through the fast lane, or voucher incentives). The mobile app 22 (Figure 2) is provides this informative and interactive functions to the guests while they visit the theme park.

As a key function, guests can participate in the recommended itineraries as individual or a group. By following the itinerary, they will be rewarded with fast lane VQ slot or vouchers. The total amount of fast lane VQ slots and vouchers for daily operation are constrained by the operational status and overall planning.

Referring to Figure 22, to maximise the benefits from such incentives, a dynamic pricing algorithm 190 used. The algorithm takes the historical and real-time operational status, planned resources (*e.g.*, vouchers) and itinerary plans of guests as inputs and determines two VQ token assignment schemes as outputs, namely the number of VQ tokens needed to redeem specific attraction's VQ slot (for fast lane access) or vouchers of different cash values; and the number of VQ tokens to be earned by a guest by fulfilling partially or fully of the recommended itineraries.

Static pricing strategy can be used for VQ token assignment. Pre-park opening market research and analysis can provide guidance on the forecasted popularity levels of different attraction. Based on the order of popularity levels, different VQ token values can be assigned to redeem different attraction's fast lane VQ slots. Similarly, for different cash values of vouchers, different VQ tokens can be assigned to redeem them

correspondingly. However, the real-time demand and operational status of different attractions emerge from all visitors and thus change dynamically. Static pricing strategy can find it hard to adapt to the changes and thus influence visitors' behaviours in return as a measure for implicit crowd control.

5

The objects of interest to define the dynamic pricing (VQ token values) include attractions and facilities of consumption functions, where vouchers can be redeemed and consumed.

10

For each object of interest, two VQ token values are assigned, namely the VQ tokens needed for redemption of the incentives at corresponding period of time, *i.e.*, fast lane VQ slot for attractions such as rides, and voucher with time-constrained redemption period, and the VQ tokens to be collected by patron if they visit the object at corresponding period of specified by the itinerary plan.

15

The two VQ token value assignment scheme depends on various operational status such as the visitor crowd distribution, the waiting time at both normal and fast lanes of attractions, attraction popularity (in terms of historical visited counts by visitors) etc. Through dynamic adjustment of VQ token values, the system interacts with the visitors in the way that their attraction choices or itineraries are implicitly influenced (by the corresponding incentives), in return, the collective itineraries of visitors determine the overall crowd distribution in the park across different attractions, and by coordinating on different itinerary plans, the overall operational key performance indicators can be optimized.

25

The dynamic pricing strategy is composed of two algorithmic modules, namely a pricing factor sorting module (PFSM) 191 and a pricing scheme search module (PSSM) 192.

30

In PFSM 191, the incentive redemption prices are sorted by importance degrees of different factors. Specifically, there are seven steps:

35

Model training is performed to learn the correlation between historical key factors 193 and importance of factors (step S1). The training data relies on historical data or manual marking/labels 194 from operators' experience.

With the trained model together with parameter optimization, an importance degree prediction model 195 is generated (step S2). In this setup, a linear regression method may be used to formulate the prediction model. Current key factor values 196 are input into the trained importance degree prediction model 195 to predict importance degrees
5 of the factors in the current situation. Sorted importance degrees of factor data 197 (“sorted importance data”) are generated, which will be used later in PSSM 192. The sorted importance data 197 comprises key factors 198 and corresponding weights 199.

Current pricing data 200, current index data 201 (the quantitative operational
10 indicators to be optimized, *e.g.*, average number of rides taken by visitors, or maximum waiting time of attractions *etc.*), targeted index data 202 (the optimization target, which can be defined by the improvement ratio due to the adjustment of the pricing scheme), and the sorted importance data 197 are input into a black box optimization
15 algorithm 203 to search for the adjusted price data 204 (step S5). The black box optimization algorithm 201 iterates to find the best inputs that will generate the targeted outputs. The whole optimization iteration stops when a set of adjusted price data 204 is found that achieves the targeted index value.

The current pricing data 200 includes, for each attraction, two initial or current values
20 205, 206, namely, a number 205 of VQ tokens needed for redemption of incentives at a given period of time and a number 206 of VQ tokens which will be earned by a patron if they visit the attraction a corresponding period specified by the itinerary plan.

Likewise, the adjusted pricing data 204 includes, for each attraction, two updated
25 values 207, 208, namely, a number 207 of VQ tokens needed for redemption of incentives at a given period of time and a number 208 of VQ tokens which will be earned by a patron if they visit the attraction a corresponding period specified by the itinerary plan.

30 One of the aims of PFSM 191 is to determine the key factors that will affect the prices of objects of interest and on how they affect the prices. In the context of theme park, based on operational experience, 19 factors 193 have been identified, together with their qualitative impact (correlation) on the prices, as set out Table 1 below:

Table 1

Key Factors to Determine Pricing	Incentive Redemption Cost	Gain through Itinerary
Day type (workday, weekend, holiday)	Negative	Positive
Seasonality (Peak, Off-peak)	Positive	Negative
Initial Period of Park Opening	Positive	Negative
Last Period of Park Closing	Negative	Positive
Ride Area Congestion Level	Negative	Positive
Resting Area nearby	Positive	Negative
Crowdedness	Positive	Negative
Historical Popularity	Positive	Negative
Serving Cycle Time	Positive	Negative
Capacity Per Serving Cycle	Positive	Negative
Remaining Capacity of Fast Lane	Negative	Positive
Remaining Capacity of Normal Lane	Negative	Positive
Total Normal Lane Queue Length	Positive	Negative
Total Fast Lane Queue Length	Positive	Negative
Serving throughput (queue clearance time)	Negative	Positive
Cost of One Service Cycle	Positive	Negative
Preference = Visited Count / Total Visitors	Positive	Negative
Targeted service group diversity	Positive	Negative
Connectivity	Positive	Negative

The objective of the PSSM 192 is to search for the adjusted price data that meet the target index values from the black-box model. In this example, Bayesian optimization
5 algorithm is used to solve the black-box optimization problem.

Referring to Figure 23, the Bayesian optimization algorithm 203 is shown.

An updated Gaussian process regressor model 209 is trained based on the current
10 index data 201 and current pricing data 200 (step S5.1). The acquisition function 212 of the Gaussian Process regressor is used to decide whether the index values generated from the pricing model meet with the required target index values (step S5.2).

If the index values generated from the pricing model meet the required target index values, then the algorithm has found the adjusted pricing data 204 and the data is output.

- 5 If the index values generated from the pricing model do not meet the required target index values, then the process continues as follows. The acquisition function is designed based on the outputs of the trained Gaussian process regressor 209, which is then used to measure the weight of points in the Gaussian process. The extreme values of acquisition function are used to determine the next sampling point in the Gaussian
10 process and to refresh the pricing data accordingly (step S5.3).

The updated pricing data 204 in the black-box model is applied to predict the updated index data 210 (step S5.4).

- 15 Steps S5.1 to S5.4 are iterated until the updated index data 210 meets the desired targets (step S5.5).

The dynamic pricing method in the VQ solution can adapt according to the dynamic crowd distribution and operational status of the attractions in the park. Together with
20 other VQ functions such as itinerary planning and incentive recommendation, dynamic gaming is a tool can be used to optimize the operation of theme park in two ways. First, it diverts the crowds to avoid overly concentrated congestion at selected areas. Secondly, it improves user experience as an incentive to encourage diverse experience in the park.

25

Spatiotemporal network-based itinerary planning method

In a concentrated zone of attractions that offer different service and experience to visitors (such as theme parks), there is an emerging need to control and divert crowds due to reasons such as safety, operational efficiency and load balancing for operation
30 management. On the other hand, visitors often expect customized information and guided service such as itinerary plan to enhance their visiting experience. Based on both needs of crowd diversion and customized information and guidance to visitors, a customized itinerary planning function serves this purpose and it is a core function in the VQ Solution.

In constructing a network topology to represent the information in a theme park, every node includes both spatial and temporal information. When designing computation on the network, the edges between nodes represent certain correlation between nodes based on their spatial and temporal information.

5

Mixed integer programming involves specific types of linear programming algorithms, which are usually used for optimization problems. In integer programming, the variable values to search for are limited to integers.

10 Constraint programming is a type of search algorithm to find feasible solutions from a very large set of candidates. Constraint programming is based on feasibility, but not absolute optimization of the solution, and it focuses on constraints and variable space instead of objective function as other optimization methods. In fact, a constraint programming algorithm can be also applied to find solution without an explicit
15 objective function. In such cases, a constraint programming utilizes constraints to restrict solution set to subsets which are feasible to be solved within reasonable time.

Users can gain a certain number of rewarding points (referred to as a “token”) after they follow the itinerary plan to visit items on the recommended itinerary. By collecting
20 tokens, a visitor can gain incentives such as fast lane VQ slot or vouchers. It is a mechanism to encourage visitors to use and stick to the itinerary plans provided by the system.

As explained earlier, there is an emerging need for effective crowd control and
25 diversion need due to reasons such as safety, operational efficiency and load balancing for operation management. On the other hand, visitors often expect customized information and guided service such as itinerary plan to enhance their visiting experience. Based on both needs of crowd management and customized information and guidance for visitors, a customized itinerary planning function serves the purpose
30 and it is a core function in the VQ Solution.

By considering personal factors such as thrill level preference, ride and themed zone preference *etc.*, a recommended itinerary plan can filter suitable attractions, sort them in order with suggested visiting time slots to visitors. At the same time, by considering
35 operational status factors such as crowded level of different attractions, which is quantified as VQ tokens in the VQ solution, the generated itinerary plan to coordinates

among different visitors by diverting them more evenly to various attractions at a same time slot, and to encourage them to commit to such itinerary plans, higher-value or additional VQ tokens might be awarded to them so that to ensure they can gain corresponding tokens soonest. For example, when a visitor has specific desired ride fast lane VQ slot as an incentive to be earned, such itinerary plan needs to make sure of enough tokens to be earned collectively by following such an itinerary plan.

There are mainly two other crowd diversion methods.

In forecasted crowded scenarios, a diversion plan is predefined. For example, if attraction A is forecasted to be very popular during a specific time periods, the operation management can prepare corresponding contingency measures specifically for different crowded levels. They might include temporary close of queues, special crowd split control etc. The main disadvantage of this method is its flexibility to handle the situations that dynamically change and happen. Besides, such contingency measures usually have side-effect on visitor experience.

In another approach, a number of itineraries for visitors which can be chosen can be fixed. For example, a *Night Safari of Singapore* ride has specially-designed night train routes to provide guided tours for the majority of visitors. However, for a theme park, it is difficult to fix such itineraries with different personal preferences, and such methods lack of support for customized information and guidance.

Neither of these crowd diversion methods are, therefore, considered to provide a satisfactory solution.

Referring to Figure 24, an itinerary plan 10 for display on a mobile device 23 (Figure 2) is shown.

The itinerary plan 10 can be provided to visitors as a timeline 213 of activities with one or more alternative attractions 214₁, 214₂, 214₃ during each phase 215₀, 215₁, 215₂. Once the visitor fulfils each phase 215₀, 215₁, 215₂ by visiting one of the recommended attractions 212, 213, 214, and successfully tap-out by scanning the QR code 50 (Figure 7) using their mobile app 22 (Figure 7) at the VQ token terminals 26 (Figure 7) at the end 46₂ (Figure 7) of the queueing area 8, 9 (Figure 7) during the recommend period of time, the visitor earns certain VQ tokens 44 (Figure 6). Sufficient VQ tokens can later

be used to redeem corresponding incentives (automatically triggered by an algorithm) such as the VQ slot for fast lane access of a desired ride 216.

5 The itinerary plan 10 is presented to the user on their mobile device 23. The user does not need to select which of the attractions 214₁, 214₂, 214₃ they choose to ride. Instead, the system 20 can track the user through tapping in and tapping out.

10 The itinerary plan 10 may include times for attractions entered via a normal lane and so the itinerary plan 10 may include times and queue slots for normal and fast lanes. For a recommended attraction that is accessed through a normal lane, the plan may use the start time of a VQ slot (which uses the fast lane), a currently-predicted future waiting time for an attraction accessed via a normal lane as a duration and a buffer (i.e., a time buffer) to set the end time of a sub session. For example, an itinerary plan may include three rides, namely rides A, B and C, and may consist of ride A, accessed via the normal
15 lane, specifying a time window of 09:00 to 09:20, ride B, also accessed via a normal lane, specifying a time window of 09:30 to 10:15 and ride C, accessed via the fast lane, with an existing VQ reservation of 10:30 to 11:00. In this example, it is assumed that VQ slots for rides A, B and C have a duration of 30 minutes. However, the end periods for rides A and B are not the starting time plus 30 minutes (i.e., based on VQ slot
20 duration), but instead predicted normal lane waiting times (in this case, 20 minutes and 45 minutes respectively).

Referring to Figures 25, 26, 27 and 28, a method of generating a customized itinerary plan 10 is shown.

25 Referring to Figure 25, an algorithm architecture 221 is shown, together with sources 222, 223, 224, 225 of data. The algorithm architecture 221 includes ride selection 226 for generating candidate attractions 227 and data pre-processing 228 for generating basic information about attractions 229. These are supplied to an itinerary generation
30 algorithm 230.

Referring to Figures 25 and 26, in itinerary item selection: candidates are found based on the degree of match between the operational status 222, 223 and features 224 of attractions, such as flow patterns of different ride types, and user's preferences 225, for
35 instance tendencies of visiting particular types of rides 3, 4, user profile such as age, weight, height, and group properties. In Figure 26, shaded rides are those that

matched preferences of the patron. The size of the circle indicates the slot duration of the ride, *e.g.*, smallest circles correspond to a slot duration of 10 minutes, larger ones to 30 minutes *etc.*

- 5 Referring in particular to Figure 27, in itinerary plan generation 230, a search is performed for a feasible sequence 232 of attractions (or groups of attractions to provide alternative candidates) from candidates 227 (Figure 25) found itinerary item selection based on their spatial and temporal relationships 229 (Figure 26).
- 10 Specifically, itinerary item selection 228 (Figure 25) chooses a set of attractions which meets an individual visiting demand. When an individual user triggers itinerary planning function in the app 22 (Figure 2), personal data 225 including age, weight, thrill level *etc* is provided to the system, and the system performs a filter check to find a matching set of attractions meet personal preference and time-constraints. User
- 15 visiting history is also used to avoid duplicating attraction recommendation in itinerary generation. After ride selection process, candidate attractions 229 are selected for this individual user.

Itinerary plan generation 230 is configured to make efficient use of park facilities and

20 to make the user's visiting experience smooth and rewarding. In this module, parameters like waiting time from VQ prediction 222 (Figure 25) and attraction value 223 (Figure 25) from dynamic gamification management are used. The itinerary takes existing reservation into account, by reserving the corresponding time slots for the reserved attractions. To search for a feasible sequence of attractions, a spatiotemporal

25 network is constructed involving the relationship of attractions based on their operational slots, and spatial connection and geometrical topology.

In a spatiotemporal network model, there are sources, sinks, nodes and arcs as basic factor. An arc contains a ground arc 233 and connect arc 234. A ground arc 233 is a

30 possible session of an itinerary. It reflects actual ride slots group, mainly containing two slots of different rides. If there is only one ride slot, this ground arc has a low token price. A connect arc 234 connect different ground arcs 233, reflecting walking or connection between rides. For a connect arc, start time of arrival ground arc is earlier than departure ground arc. In this spatiotemporal network model, it is a directed graph

35 and satisfy other flow network constraints. A feasible itinerary plan is a path on the network that link a sequence of nodes together with their edges. In the context of the

VQ solution, itinerary plan serves as an inherent crowd diversion method while providing rewards to improve user experience. Thus, the task of itinerary plan construction is formulated as an optimization problem to search for paths on the spatiotemporal network with two specific goals to optimize: to maximize VQ tokens to be earned through the itinerary (to be more rewarding to visitors) and to minimize waiting time of all attractions (both average and standard deviation so that the crowd is more evenly distributed to different attractions).

To solve the optimization problem on the spatiotemporal network, Mixed Integer Programming (MIP) is used comprising:

$$\begin{aligned} \text{Max: } & \sum c_i x_{i,i} + \sum w_{i,j} x_{i,j} \\ \text{s.t. } & \\ & \sum_k x_{k,i} = \sum_j x_{i,j} \\ & \sum_k x_{k,i} \leq 1, \forall i \in \text{Ride}_j \\ & \sum_k x_{k,i} = 1, \forall i \in \text{Reserve}_j \end{aligned}$$

$x_{i,j}$ is a decision variable, if i equals j , it represents a ground arc (ride group arc, mainly contains two slots of different rides). Otherwise, it represents a connect arc (connect different ground arc is earlier than departure ground arc, $x_{i,j}$ is 0).

c_i is the average token of arc. At phase 1a stage, a connect arc token is 0.

$w_{i,j}$ is an average cost of arc $x_{i,j}$. At phase 1a stage, it is set as 0, but if more data collected, ground arc waiting time cost can be normalized and connect arc walking time and cost can be normalized in this term.

The constraints can be mainly divided in three types, namely ground arc in and out balance arc, ground arc in-arc up bound, ride number constraint and reserve arc constant constraint.

To enhance user experience, diversity in the generated itinerary plan is introduced such that similar itineraries are generated but with slight differences in the itinerary plan in different requests within short period of time by the same visitor (e.g., the user requests for a new itinerary after preview and finds the provided one not meet the expectation).

5 To solve this problem, constraint programming (CP) is used together with MIP, which finds feasible solutions instead of optimal solutions as MIP does. Constraints differ from the common primitives of imperative programming languages in that they do not specify a step or sequence of steps to execute, but rather the properties of a solution to be found. In addition to constraints, users also need to specify a method to solve these
10 constraints. Different methods, such as clause learning, implication graph search and build, local search, backtracking search *etc.*, can be used. Methods such as call-back can be used to ensure diversity of solutions. Solution time limit and solution size limit can be set both to help diversity of solutions. Current searched solutions can also be set as constraints to make different solution output.

15 Thus, a spatiotemporal network is used to construct attraction connections in the theme park. Such a data structure can specify details involved in itinerary plan, such as every attraction's operation time slides, spatial connection and topology among attractions. The spatiotemporal network also provides flexibility to cater for different
20 operational configurations, such as connection among attractions reflected by the edge weights, operational status such as waiting time of attractions being characterized by in-and out- degree, and weights of nodes *etc.* Furthermore, feasible itinerary plans can be found with diversity driven by VQ tokens enables customized solution to users, while serving operational crowd control purpose at the same time.

25 Multi-agent modelling method for patron attraction choice behaviour simulation

A virtual queue is a time-slot based reservation system allowing users (*i.e.*, patrons) to claim their right to access a particular attraction or resource (such as a ride in a theme park) through a fast lane.

30 A virtual queue slot is a period of time defined to allow a batch of patrons who have reserved particular attraction or resource to physically enter the fast lane.

A fast lane is a dedicated queue/passage for patrons to access a particular attraction or
35 resource with expected shorter waiting time in the queue. In the context of a theme park herein described, the fast lane is dedicated for VQ slot holders and VIP guests.

A normal lane is a queue/passage area for all patrons to wait before accessing a particular attraction or resource.

5 An agent is a virtual entity with autonomous decision-making and behaviours to interact with other entities in a specific environment (such as a theme park). Different agent types define their decision-making process and behavioural rules correspondingly.

10 Herein, two types of agents are mainly considered, namely a patron agent and an attraction agent. A patron agent is an entity used to simulate visitors of the park in a computer. More specifically, their decision-making process is modelled for the next attraction choice and queueing behaviours under different situations (for example, queueing in fast lane or normal lane). An attraction agent is an entity used to simulate the operation of facilities (such as rides) that provides different experience to patrons in
15 the park. More specifically, the operation process of the attraction is modelled including their queue admission (through tap-in validation for fast lane), queue clearance (to load a batch of patrons from the queues to rides), and ride/show service (to host a batch of patrons in the attraction for a fixed period of time). The interactions of these two types of agents emerge in different quantitative indicators that reflect the
20 status of the theme park, such as crowd density, average waiting time of a ride and so on.

25 An evolutionary algorithm (EA) is a subset of evolutionary computation, namely a generic population-based metaheuristic optimization algorithm. An evolutionary algorithm uses mechanisms inspired by biological evolution, such as reproduction, mutation, recombination, and selection. Candidate solutions to the optimization problem play the role of individuals in a population, and the fitness function determines the quality of the solutions.

30 Theme parks generally have a high density of crowds. Furthermore, crowd distribution is not even in space and time due to the nature of variant popularity levels of attractions due to their design. Some rides attract more patrons than others, and facilities like food-and-beverage kiosks and restaurants attract more guests during specific period of time (for example, during lunch or dinner). As a result, a long queueing time at
35 popular attractions quickly becomes one of the most emerging challenges for park operators to address. Under a specific theme park setup, a solution is sought as to how

to make best use of the limited available space available, such as queues of different attractions, and divert patrons more across the park more evenly to different locations based on the current situation.

5 Data sensing and understanding patron distribution is important in trying to understand the current situation. Even with sensors deployed with the VQ solution such as the VQ validation terminals 25 (Figure 2) and the VQ token collection terminals 26 (Figure 2) at selected locations, footprints of patrons can only be obtained at a coarse level of granularity both in space and time. Herein, multiagent modelling is used
10 to improve the level of granularity. The objective is to provide higher granularity of crowd distribution information through simulating patron's next attraction choice behaviours based on existing data, such as sensed density of localities, partial footprints of patrons, operation status of attractions and VQ functions utilized by patrons through the provided VQ app 22 (Figure 2), such as itinerary planning and incentive utilization.

15 Furthermore, key parameters regarding patron behavioural choices in the simulation model is calibrated using evolutionary algorithm approach, which can be achieved with collected observed crowd count data at selected locations along time in the theme park.

20 Referring to Figure 28, an agent-based modelling and simulation system 235 is shown. The system 235 models patrons and attractions as autonomous agents and simulates their behaviour and interaction over time.

The system 235 generally includes patron agents 236 and attraction agents 237 which
25 define a model 238 and a scheduler 239 which controls agent activation and handles time in the model.

Referring also to Figure 29, the patron agent 236 include a patron decision making module 241 for deciding a next attraction choice and a patron behavioural module 242
30 for executing a current choice, and the attraction agent 237 includes an attraction behaviour module 243 for interacting with patrons.

Upon arrival of the theme park and after binding their tickets into the VQ app 23 (Figure 2) (step S24.1), the patron interacts with the VQ system. Decision logic of the
35 patron agent 236 determines whether the patron has "Next Attraction Choice?" (step S24.2) to identify along which branch further computation will proceed.

If the decision is “No”, then patron agent 236 executes the patron decision making module 241 to decide next attraction choice. Otherwise, the patron agent 236 executes the patron behavioural module 242 to execute a current decision to simulate the process of arriving at an attraction, joining a queue, waiting in the queue, enjoying the ride and leaving the attraction. The process also depends on the interaction with the attraction agent 237.

Patron decision making module

The decision-making process of a patron agent is based on the status of the agent with regard to the VQ functions. For example, they can make an advance reservation of VQ slots for an attraction to join the fast lane at corresponding time slot. Additionally or alternatively, they can use the itinerary plan function 94 (Figure 14) of the VQ app 22 (Figure 13) to obtain guided itineraries, where they secure incentives upon completion. Additionally or alternatively, they can get time-constrained vouchers to redeem at specific facilities, such as restaurants, as incentives to encourage them to visit. The decision-making process of a patron agent is also based on user profile.

Referring to Figure 29, the decision process checks whether patron agent “Has Immediate Reservation” in the following simulation time period T_i' , which is a period defining how near the upcoming reservation is from the current simulation time. If “yes”, then based on a model parameter θ_1 , the algorithm determines whether the patron agent 236 has an upcoming reservation (or “immediate reservation”). Here, θ_1 represents a tendency of a patron to commit to their advance reservation and can be tuned computationally or calibrated through data collection from the actual patrons’ footprint in practice. If “no”, then the algorithm proceeds to the next decision logic whether the patron “Has Itinerary Plan?”

The algorithm checks whether the patron agent 236 “Has Itinerary Plan”. If “yes”, then, based on a model parameter θ_2 , the algorithm determines whether the patron agent 236 chooses the next attraction on the itinerary plan as the next attraction choice. Here, θ_2 represents a tendency of the patron to commit to their itinerary plan and can be tuned computationally or calibrated through data collection from the actual patrons’ footprint in practice. If “no”, then the algorithm proceeds to the next decision logic whether the patron “Has Time-Constrained Voucher Redemption?”

Third, the algorithm checks whether the patron agent 236 “Has Time-Constrained Voucher Redemption” in the following simulation time period T_2' , which is a period defining how near the time-constraint of voucher redemption from the current simulation time. If “yes”, then based on a model parameter θ_3 , the algorithm
5 determines whether the patron agent chooses the voucher redemption facility as the next attraction choice. Here, θ_3 represents the tendency of patron to utilize the obtained voucher incentive at the specific simulation time and location and can be tuned computationally or calibrated through data collection from the actual patrons’ footprint in practice. If “no”, then proceed to the “Choice Model with User Profile” to determine
10 the next attraction.

If the patron agent 236 does not fulfil any criterion, the algorithm determines the next attraction choice based on a model (“choice model”) chosen according to a user profile. The model includes a comprehensive list of factors including popularity of ride,
15 distance from the ride, waiting time of normal lane of the ride, user ride preference, user ride thrilling level preference, user food preference, the current simulation time of the day, and itinerary history of the patron on the day will determine the next attraction choice. Such factors can form a weighted sum to calculate a utility value of each ride at a given simulation time. According to the utility value, the top M choices with the
20 highest utility can be generated, and the actual choice is chosen with random out of the M choices. M may be, for example, three.

Patron behavioural module

The simulation seeks to refine the local density of attractions and, more specifically, the
25 waiting time in queues. Thus, the patron behaviour module 242 focuses on the process of arriving, waiting and leaving an attraction, and it is then specified in five patron execution (PE) steps:

In step PEO, the patron agent arrives at the designed attraction and the module updates
30 its location at simulation time t_o

In step PE1, the patron agent joins a queue (fast or normal lane) depending on the interaction with the attraction agent. The attraction agent behaviour design is described in more detail hereinafter.

In step PE2, the patron agent waits in a queue and the module update its sequence in the queue at simulation time t_1 .

5 In step PE3, the patron agent leaves the queue and enjoy the ride as one of the batches of patrons to be loaded into the ride at simulation time t_2 .

Finally, in step PE4, the patron agent leaves the attraction after taking the ride and update patron agent's current attraction choice as none at simulation time t_3 .

10 The algorithm proceeds to the next iteration of simulation to execute the patron decision making process.

Attraction Behavioural module

15 Referring still to Figure 29, an attraction at a theme park can be seen to serve patrons through four main processes:

In step A1, the attraction agent admits the patron ("queue admission"). At simulation time t_o' , the attraction agent checks the VQ slot validity of the visiting patrons. If a patron agent has a VQ slot at the corresponding time, the patron agent is added into the fast lane. An assumption is made in the simulation that the fast lane will not be physically full due to the fact that the fast lane is mainly used to serve VQ slot holders and VIPs. The solution will moderate the total number of VQ slot holders during a slot period to prevent over-crowdedness. With the uncertainty of VIP group to join the fast lane, operation can still manoeuvre the output throughput ratio between fast lane and normal lane to prioritize the fast lane guests with shorter waiting time in practice. Thus, an assumption is made to simplify the queue admission process in the simulation model. For guests without a valid VQ slot at time t_o' , a normal lane can admit guests if the normal lane has capacity, *i.e.*, is not full. Otherwise, the patron agent returns to the initial status of waiting to be admitted into queues.

30

In step A2, the attraction agent accommodates guests in queues ("host guests in queues"). During simulation period T_l' , the attraction agent checks the simulation time to determine "Is Next Ride Cycle Ready?". T_l' can be determined based on the ride's dispatch time, and full cycle time, which is further defined as the sum of the dispatch interval, ride cycle and loading and unloading time, which can be obtained from the operational fact sheets as scheduled. If a new ride cycle arrives at simulation time t_l' (=

35

$t_o' + T_1'$), a set number α of patron agents are removed from the queues of fast lanes and normal lanes with a predetermined ratio β (e.g., $\beta = 4:1$ as a guided empirical value based on operational experience). A list of patron agents are simulated to enjoy the rides in the following period of time T_1' . α represents the number of patrons to be
5 loaded into ride at each ride cycle, and it is upper-bounded by the per dispatch-capacity of the ride, which is another operational parameter to be set according to the design of a particular attraction.

In step A3, the attraction agent accommodates guests in the ride (“host guests in rides”).
10 Patron agents who were removed from the queues, now virtually experience the ride during the period of time T_1' .

In step A4, the attraction agent unloads guests from rides (“unload guests from rides”). Patron agents are removed from the attraction agent at the end of time period T_1' to
15 simulate the fact that patrons leave the attraction after taking the ride. The patron and attraction agents iterate their decision and behaviour cycles autonomously, and their interaction emerges into the final patron distribution of crowds in the theme park. Several parameters can be estimated in the simulation that may result in different crowd distribution results. Here, an Evolutionary Algorithm (EA) is used based
20 optimization method to calibrate such parameter values in order to generate crowd density that best matches with the observation at limited locations.

Reference is made to J Zhong, N Hu, W Cai, M Lees, L Luo: “Density-based evolutionary framework for crowd model calibration”, Journal of Computational
25 Science (2015) and N Hu, J Zhong, JT Zhou, S Zhou, W Cai, C Monterola: “An automatic crowd control framework using multi-objective genetic programming” Applied Soft Computing (2018) which are incorporated herein by reference.

In this case, the EA optimization method is formulated as follows.

30

The search space is defined as a vector of parameters to be calibrated. For example, if one wants to consider patrons’ intention of commit to their reservation, itinerary plan, or voucher redemption accordingly, the vector X is defined as:

$$X = \overleftarrow{\theta}, i \in [1,2,3], \theta_i \in [0,1]$$

35

The objective function is defined as a difference between the observed value and the simulated values. For example, at N given locations, one can collect the observed patron count C_i , $i \in [1, N]$, at an interval of T minutes (for example, through CCTV videos or manual counts from operation personnel at different attractions), then the
5 observed crowd counts (distribution) across these locations along time can be represented as a vector Y:

$$Y = \overline{C_i^t}, i \in [1, N], t = t_o + nT, \text{ where } n \in [0, \frac{t_c - t_o}{T})$$

10 Where t_o is the opening time of the park where the observation starts, t_c is the closing time of the park where the observation ends.

In in simulation, crowd counts can be collected at the same time instances and locations in simulated scenarios as a vector Y'. Then the difference between these two
15 vectors ΔY can be measured by calculating their cosine similarity as:

$$\Delta Y = \frac{y \cdot y'}{\|y\| \cdot \|y'\|} = \frac{\sum_{i=1}^n (y_i * y'_i)}{\sqrt{\sum_{i=1}^n (y_i)^2} * \sqrt{\sum_{i=1}^n (y'_i)^2}}$$

In order to find the minimal difference $\min(\Delta Y)$, the EA algorithm searches for the set
20 of values X to be used in the simulation and generates a set of simulation values Y' until the objective value ΔY converges.

Reference is made to MASON multiagent simulation toolkit:

<https://cs.gmu.edu/~eclab/projects/mason/>

25

Modifications

It will be appreciated that various modifications may be made to the embodiments hereinbefore described. Such modifications may involve equivalent and other features which are already known in the design, manufacture and use of queue management
30 systems and component parts thereof and which may be used instead of or in addition

to features already described herein. Features of one embodiment may be replaced or supplemented by features of another embodiment.

Although claims have been formulated in this application to particular combinations of
5 features, it should be understood that the scope of the disclosure of the present
invention also includes any novel features or any novel combination of features
disclosed herein either explicitly or implicitly or any generalization thereof, whether or
not it relates to the same invention as presently claimed in any claim and whether or
not it mitigates any or all of the same technical problems as does the present invention.
10 The applicants hereby give notice that new claims may be formulated to such features
and/or combinations of such features during the prosecution of the present application
or of any further application derived therefrom.

Claims

1. A computer-implemented method, comprising:
dynamically calculating supply and demand of queue slots available to visitors to a
5 plurality of spaced-apart attractions distributed across a domain; and
dynamically managing reservations of queue slots made by visitors based on the
supply and demand and/or
dynamically providing incentives to visitors for influencing behaviour and/or
location of visitors in the domain based on the supply and demand.
10
2. The method of claim 1, wherein dynamically calculating supply and demand of
queue slots available to visitors to a plurality of spaced-apart attractions distributed
across a domain comprises:
dynamically predicting the supply of queue slots available to visitors to a plurality
15 of spaced-apart attractions distributed across a domain, the queue slots having
specified times; and
dynamically predicting demand for the queue slots by visitors for the plurality of
spaced-apart attractions.
- 20 3. The method of claim 1 or 2, wherein each queue slot has a respective cost and
dynamically managing reservations of the queue slots is based on costs of the queue
slots.
4. The method of any one of claims 1 to 3, further comprising:
25 allocating a reward to a user which is redeemable to secure a given queue slot.
5. The method of any one of claims 1 to 4, further comprising:
generating an itinerary plan for a visitor, the itinerary plan including a list which
includes an attraction based on an allocated queue slot or a list of two or more
30 attractions or three or more attractions, arranged in order, having allocated queue slots
for the visitor.
6. The method of any one of claims 1 to 5, wherein dynamically calculating supply
and demand of queue slots comprises:
35 receiving current pricing data which includes a current value of a cost of securing
a slot to an attraction and a current value of a reward for visiting the attraction, current

index data relating to current operational conditions of attractions, target index data relating to desired operational conditions of attractions and sorted pricing data comprising a set of factors which affect a cost of securing a slot to an attraction and a set of corresponding weights for the set of factors;

- 5 providing the current pricing data, current index data, target index data and sorted pricing data to an optimization algorithm to find adjusted price data; and upon reaching a predetermined target, outputting the adjusted price data.

7. The method of claim 6, wherein the operational conditions of attractions include crowd distribution in a domain across which attractions are distributed.

8. The method of claim 6 or 7, wherein the operational conditions of attractions include waiting times of attractions.

9. The method of claim 8, wherein the waiting times of attractions include normal line waiting times and fast line waiting times.

10. The method of any one of claims 6 to 9, wherein the operational conditions of attractions comprise popularity of attractions.

11. The method of any one of claims 6 to 10, wherein the reward for visiting the attraction is expressed as a number of tokens.

12. The method of any one of claims 6 to 11, wherein the optimization algorithm is a Bayesian optimization algorithm.

13. The method of any one of claims 6 to 12, wherein the optimization algorithm comprises a Gaussian process regressor and an acquisition function.

14. The method of any one of claims 6 to 13, wherein the set of factors which affect the cost of securing the slot to the attraction include a type of day, wherein types of day include a week day, a day of a weekend or a holiday.

15. The method of any one of claims 6 to 14, wherein the set of factors which affect the cost of securing the slot to the attraction include seasonality, wherein seasonality includes peak or off-peak.

16. The method of any one of claims 1 to 15, wherein dynamically managing reservations of queue slots made by visitors based on supply and demand comprises:
in response to a user request, selecting at least one ride and at least one queue slot which is relatively less popular than a queue slot for the ride or another ride and
5 presenting the selected ride(s) and queue slot (s) to the visitor.
17. The method of claim 16, wherein the queue slot for the ride or other ride is requested by the user in the user request.
- 10 18. The method of any one of claims 1 to 17, wherein dynamically providing incentives to visitors for influencing behaviour and/or location of visitors in the domain based on the supply and demand, comprises:
receiving status information about visitor(s) in a domain across which a plurality of spaced-apart attractions are distributed;
15 generating an incentive for affecting behaviour and/or location of a visitor in dependence upon the status information; and
transmitting the incentive to a mobile device of the visitor.
19. The method of claim 18, wherein the status information indicates a crowding
20 condition in a first region and the incentive is for encouraging the visitor to visit a second region which does not have a crowding condition.
20. The method of claim 18 or 19, wherein the status information indicates compliance with a proposed itinerary and the incentive is for rewarding with a
25 proposed itinerary or encouraging compliance with a future proposed itinerary.
21. The method of any one of claims 18 to 20, wherein the status information indicates a shortfall in delivery and the incentive is for compensating the visitor.
- 30 22. The method of any one of claims 18 to 21, wherein the incentive comprises a voucher in relation to a product or service.
23. The method of any one of claims 18 to 22, wherein the incentive comprises a gift.
- 35 24. The method of any one of claims 18 to 23, wherein generating the incentive depends on rewards earned by the visitor in the domain.

25. The method of any one of claims 18 to 24, further comprising:
receiving historical queue data about queue lengths for a plurality of spaced-apart attractions distributed across a domain;
receiving predicted queue data about the queue lengths for the plurality of
5 spaced-apart attractions;
receiving information about visitors currently in the domain; and
in dependence on the historical and predicted queue data, and the information about visitors currently in the domain, generating incentives for at least some of the visitors currently in the domain to encourage at least some of the visitors to move to
10 given areas within the domain.
26. The method of claim 25, wherein the incentives include vouchers for redemption at retail outlets.
- 15 27. A computer-implemented method, comprising:
receiving a first set of parameters relating to access and/or capacity attributes of a plurality of spaced-apart attractions distributed across a domain;
receiving a second set of parameters relating to temporally-variable
characteristics affecting at least some of the plurality of spaced-apart attractions;
20 building a model of the plurality of spaced-apart attractions based on the first and second sets of parameters;
receiving data relating to a time-dependent supply of visitors to the domain;
determining, for each of the plurality of spaced-apart attractions, a predicted
number of visitors to the attraction as a function of time; and
25 outputting and/or storing the predicted number of visitors.
28. The method of claim 27, further comprising:
receiving, for each of the plurality of spaced-apart attractions, historical data relating to historical numbers of visitors to the attraction as a function of time;
30 wherein building the model is based on the data relating to historical data.
29. The method of claim 27 or 28, further comprising:
receiving, for each of the plurality of spaced-apart attractions, pricing data affecting attractiveness of the attraction to visitors.

30. The method of any one of claims 27 to 29, wherein first set of parameters comprises:
capacity of each of the plurality of spaced-apart attractions.
- 5 31. The method of any one of claims 27 to 30, wherein first set of parameters comprises:
capacity of one or more queuing areas for each of the plurality of spaced-apart attractions.
- 10 32. The method of any one of claims 27 to 31, wherein first set of parameters comprises:
spatial layout of the plurality of spaced-apart attractions.
- 15 33. The method of any one of claims 27 to 32, wherein first set of parameters comprises:
ingress and egress configurations of the plurality of spaced-apart attractions.
34. The method of any one of claims 27 to 33, where the model is a spatial-temporal model.
- 20 35. The method of any one of claim 27 to 34 comprising determining, in a spatial view, an effect on the number of visitors to a given attraction by numbers of visitors to at least nearest neighbour attractions to the given attraction.
- 25 36. The method of any one of claim 27 to 35 comprising determining, in a temporal view, an effect on a rate of change of the number of visitors to a given attraction by a rate of change of numbers of visitors to at least nearest neighbour attractions to the given attraction.
- 30 37. The method of any one of claims 1 to 36, wherein the domain is:
an amusement park, a holiday resort, an integrated resort, a recreational village, a hotel complex, a cinema complex, a convention center, an exhibition center, a cruise ship, or a transport hub, or a part thereof; or
a tourist attraction, a cinema, a theatre, a concert hall, a casino, a sporting facility,
35 a playground, or a play area, or a part thereof; or
a class.

38. A computer program for performing the method of any one of claims 1 to 37.

39. A computer program product comprising a computer-readable medium storing the computer program of claim 38.

5

40. Apparatus comprising:
at least one processor; and
memory;

the at least one processor configured to perform the method of any of claims 1 to 37.

10

41. A computer-implemented method, comprising:
receiving historical data relating to a set of factors which affect a cost of securing a slot to an attraction;
training a machine learning model using the historical data to obtain a trained
15 model;

receiving current data relating to the set of factors; and
in dependence upon the trained model and the current data relating to the set of factors, generating a set of weights for the set of factors.

20

42. A computer-implemented method, comprising:
receiving current pricing data which includes a current value of a cost of securing a slot to an attraction and a current value of a reward for visiting the attraction, current index data relating to current operational conditions of attractions, target index data relating to desired operational conditions of attractions and sorted pricing data
25 comprising of set of factors which affect a cost of securing a slot to an attraction and a set of corresponding weights for the set of factors;
providing the current pricing data, current index data, target index data and sorted pricing data to an optimization algorithm to find adjusted price data; and
upon reaching a predetermined target, outputting the adjusted price data.

30

43. The method of claim 42, wherein the operational conditions of attractions include crowd distribution in a domain across which attractions are distributed.

35

44. The method of claim 42 or 43, wherein the operational conditions of attractions include waiting times of attractions.

45. The method of claim 44 wherein the waiting times of attractions include normal line waiting times and fast line waiting times.
46. The method of any one of claims 42 to 45, wherein the operational conditions of attractions comprise popularity of attractions.
47. The method of any one of claims 42 to 46, wherein the reward for visiting the attraction is expressed as a number of tokens.
48. The method of any one of claims 42 to 47, wherein the optimization algorithm is a Bayesian optimization algorithm.
49. The method of any one of claims 42 to 48, wherein the optimization algorithm comprises a Gaussian process regressor and an acquisition function.
50. The method of any one of claims 41 to 49, wherein the set of factors which affect the cost of securing the slot to the attraction include a type of day, wherein types of day include a week day, a day of a weekend or a holiday.
51. The method of any one of claims 41 to 50, the set of factors which affect the cost of securing the slot to the attraction include seasonality, wherein seasonality includes peak or off-peak.
52. The method of any one of claims 41 to 51, wherein the cost of securing the slot to an attraction is expressed as a number of tokens.
53. A computer program for performing the method of any one of claims 41 to 52.
54. A computer program product comprising a computer-readable medium storing the computer program of claim 53.
55. Apparatus comprising:
at least one processor; and
memory;
- the at least one processor configured to perform the method of any of claims 41 to 52.

56. A computer-implemented method, comprising:
receiving status information about visitor(s) in a domain across which a plurality
of spaced-apart attractions are distributed;
generating an incentive for affecting behaviour and/or location of a visitor in
5 dependence upon the status information; and
transmitting the incentive to a mobile device of the visitor.
57. The method of claim 56, wherein the status information indicates a crowding
condition in a first region and the incentive is for encouraging the visitor to visit a
10 second region which does not have a crowding condition.
58. The method of claim 56 or 57, wherein the status information indicates
compliance with a proposed itinerary and the incentive is for rewarding with a
proposed itinerary or encouraging compliance with a future proposed itinerary.
15
59. The method of any one of claims 56 to 58, wherein the status information
indicates a shortfall in delivery and the incentive is for compensating the visitor.
60. The method of any one of claims 56 to 59, wherein the incentive comprises a
20 voucher in relation to a product or service.
61. The method of any one of claims 56 to 60, wherein the incentive comprises a gift.
62. The method of any one of claims 56 to 61, wherein generating the incentive
25 depends on rewards earned by the visitor in the domain.
63. A computer-implemented method, comprising:
generating an incentive for affecting behaviour and/or location of a visitor in
dependence upon rewards earned by the visitor in a domain; and
30 transmitting the incentive to a mobile device of the visitor.
64. The method of any one of claims 56 to 63, further comprising:
receiving historical queue data about queue lengths for a plurality of spaced-apart
attractions distributed across a domain; and
35 receiving information about visitors currently in the domain; and

in dependence on the historical and predicted queue data, and the information about visitors currently in the domain, generating incentives for at least some of the visitors currently in the domain to encourage at least some of the visitors to move to given areas within the domain.

5

65. A computer-implemented method, comprising:
receiving historical queue data about queue lengths for a plurality of spaced-apart attractions distributed across a domain;
receiving predicted queue data about the queue lengths for the plurality of
10 spaced-apart attractions;
receiving information about visitors currently in the domain; and
in dependence on the historical and predicted queue data, and the information about visitors currently in the domain, generating incentives for at least some of the visitors currently in the domain to encourage at least some of the visitors to move to
15 given areas within the domain.

66. The method of claim 65, wherein the incentives include vouchers for redemption at retail outlets

20 67. The method of claim 65 or 66, wherein the given areas include retail outlets.

68. A computer program for performing the method of any one of claims 56 to 67.

69. A computer program product comprising a computer-readable medium storing
25 the computer program of claim 68.

70. Apparatus comprising:
at least one processor; and
memory;
30 the at least one processor configured to perform the method of any of claims 56 to 67.

71. A computer-implemented method, comprising:
retrieving, over a wireless network, information relating to times of available slots for least one attraction in a domain;
35 displaying the information relating to times of available slots for the at least one attraction;

in response to user input, reserving an available slot for a selected attraction as a reserved slot;

in response to user input, presenting a first code containing information about the reserved slot for reading by a validating terminal for allowing access to a queue for the selected attraction;

5 generating a second code which is time dependent; and
 presenting the second code for allowing tracking of the user.

72. The method of claim 71, wherein reserving the available slot comprises sending a request to reserve the available slot to a remote server.

10

73. The method of claim 71 or 72, further comprising:
 displaying an itinerary comprising at least one intermediate attraction before the selected attraction.

15

74. The method of any one of claims 71 to 73, further comprising:
 in response to receiving a message from a remote server, displaying an incentive for encourage the user to move to a target destination.

20 75. The method of any one of claims 71 to 74, wherein presenting the first code comprises displaying the first code on a display.

76. The method of any one of claims 71 to 75, wherein presenting the second code comprises displaying the second code on a display.

25

77. A computer program for performing the method of any one of claims 71 to 76.

78. A computer program product comprising a computer-readable medium storing the computer program of claim 77.

30

79. A mobile device comprising:
 a controller;
 a wireless network interface; and
 a display;

35 the controller configured:

to retrieve, over a wireless network using the wireless network interface,
information relating to times of available slots for least one attraction in a domain;

to display the information relating to times of available slots for the at least one
attraction;

5 in response to user input, to reserve an available slot for a selected attraction as a
reserved slot;

in response to user input, to present a first code containing information about the
reserved slot for reading by a validating terminal for allowing access to a queue for the
selected attraction;

10 to generate a second code which is time dependent
to present the second code for allowing tracking of the user.

80. A computer-implemented method, comprising:

in response to user input, configuring a dynamic gamification management
15 module so as to cause the dynamic gamification management module to find, for a
plurality of attractions distributed across a domain, a cost of securing a slot to an
attraction and a value of a reward for visiting the attraction; and

in response to user input, configuring an incentive recommendation module so as
to cause the incentive recommendation module, generating incentives for affecting
20 behaviour and/or location of visitors.

81. A computer program for performing the method of claim 80.

82. A computer program product comprising a computer-readable medium storing
25 the computer program of claim 81.

83. Apparatus comprising:

at least one processor; and
memory;

30 the at least one processor configured to perform the method of claim 80.