

2013年度 卒業論文

テーマパークを科学する
－MASが導く行動パターン－

2010年度入学

180100008 東 由梨

主担当教員 大田 靖 助教

副担当教員 山内 信幸 教授

2013年12月20日

同志社大学文化情報学部文化情報学科

2013年度 卒業論文

テーマパークを科学する
－MASが導く行動パターン－

同志社大学文化情報学部文化情報学科

180100008

主担当教員 大田 靖 助教

副担当教員 山内 信幸 教授

2013年12月20日

要 旨

近頃日本のテーマパークは、映画やカラオケ、旅行などの娯楽産業の多様化によって市場の伸び悩みが深刻化しており、テーマパークを取り巻く経営環境は年々厳しくなる一方である。しかし、そのような状況の中でも東京ディズニーリゾートの経営は好調であり、最多の待ち時間が400分となる人気のアトラクションも存在している。加えて、新しいアトラクションも続々と登場し、東京ディズニーリゾートは今後より一層混雑することが予想される。そうなれば、入場者は満足にテーマパークを巡ることが困難になってしまう。このような混雑の発生に対して全体の混雑度を減少させ、訪れる人を満足させる方法を考えることをテーマパーク問題という。

本研究では、テーマパーク問題を解決するために混雑の発生状況を仮想空間上に作り出し、それをもとに混雑を避け効率よくパーク内を巡る方法を提案することを目的とした。この目的を達成するために、本研究では *artisoc* というシミュレーションソフトを使用して東京ディズニーシー参考としたテーマパークのモデルを再現した。さらに先行研究でのモデルをもとに、入場者エージェントの潤滑な移動方法、各アトラクションエージェントの選好度の見直し、さらにパレード・ショーの時間や船による利用者団体の移動方法などを新たに考案しモデルの作成を進めた。

シミュレーション実験は入場者および観測者の混雑情報所持率（混雑情報を所持している人の割合）を60%に固定した上で1日の行動シミュレーションを1回とするシミュレーション実験を45回行った。また入場者とは別に5パターンの観測者のグループを作成し、滞在時間のちがいをみるために5回の試行を行った。観測者のグループにはそれぞれ100人ずつ存在するためこの実験で2500人分のデータを収集することができた。またシミュレーション経過時間によって朝、昼、夕、夜の4つの時間帯に分け、各アトラクションに並んでいる人数のデータも収集した。シミュレーション結果の値として出力された5つの観測者グループ一人一人の滞在時間のデータでは分散分析を行い、各アトラクションの行列人数のデータでは主成分分析を行った。今回のシミュレーション実験の結果と統計的な分析結果から、入場者が満足してテーマパークを巡る戦略としてゲートからの距離が遠いアトラクションから巡る方法が有効であるということがわかった。朝の行列人数が少ない時間帯にこれらのアトラクションを目指して行動し、夕方から夜にかけて他の人気アトラクションの待ち行列に並び、昼の時間帯は比較的待ち時間が少ないアトラクションを混雑情報と照らし合わながら巡っていくという方法である。人気のアトラクションは昼・夕方の時間帯に行列人数が増加する傾向にあるため、比較的行列人数が少ない朝あるいは夜の時間帯に人気アトラクションに並ぶことが有効な方法であるということがわかった。

目次

第1章	はじめに	1
1.1	テーマパークの現状	1
1.2	テーマパーク問題	2
1.3	東京ディズニーシーについて	2
1.3.1	テーマポートとアトラクション	2
1.3.2	待ち時間	2
1.4	先行研究	3
第2章	研究目的	5
第3章	研究方法	6
3.1	研究方法	6
3.1.1	artisoc について	6
3.1.2	artisoc の要素	7
3.2	テーマパークモデルの作成	7
3.3	エージェントの構成	9
3.3.1	入場者エージェント	9
3.3.2	観測者エージェント	9
3.3.3	入場者の設定	10
3.3.4	優先方向について	12
3.3.5	入場者エージェントの持つ属性と情報について	13
3.3.6	入場者のアトラクションへの行動方法	14
3.3.7	入場者エージェントにおいて使用する変数・関数	14
3.3.8	アトラクションエージェント	14
3.3.9	アトラクションの設定	15
3.3.10	アトラクションの持つ属性と情報	16
3.3.11	アトラクションエージェントにおいて使用する変数・関数	16
3.3.12	ゲートの設定	17
3.3.13	船エージェント	17
3.3.14	パレードエージェント	19
3.3.15	Universe のルール	19
3.3.16	Universe において使用する変数・関数	19

第 4 章	シミュレーション実験	20
第 5 章	シミュレーション結果	21
第 6 章	分析・結果	23
6.1	主成分分析	23
6.1.1	全アトラクションにおける主成分分析	23
6.1.2	人気アトラクションにおける主成分分析	24
6.1.3	人気アトラクションを除いたアトラクションにおける主成分分析	24
6.2	分散分析	26
第 7 章	考察	27
第 8 章	おわりに	29
	謝辞	30
	参考文献	31
付 録 A	FILO	A-1
A.1	FILO	A-1
付 録 B	使用した変数・関数	B-1
B.1	変数表	B-1
B.2	変数表	B-2
B.3	関数表	B-4

第1章 はじめに

私は夢を与えてくれるテーマパーク¹が好きだ。遊びに来ている人の顔はみんな楽しそうでたくさんの人が現実を忘れて遊んでしまう。私自身もその一人で、ずっとここにいたいとさえ思うほどである。しかし友人の中にはテーマパークに遊びに行っていない、人が多くてあまりアトラクションに乗れないから時間がもったいない、などテーマパークに興味がない人がいることもまた事実である。私はこのような人々にもテーマパークの魅力を知って楽しんでほしいと思い卒業研究において、テーマパークを訪れる人が快適かつ満足してテーマパーク内を巡る方法を研究することにした。

本論文の構成は次のようになっている。第1章ではテーマパーク問題と先行研究について述べ、第2章では、本研究の目的について述べていく。第3章では、本研究の目的を達成させるために行う artisoc とシミュレーションモデルの構築について詳しく述べる。第4章では、シミュレーション実験の方法について述べる。続く第5章では、シミュレーション実験の結果を表やグラフを用いて述べる。第6章では、第5章で述べたシミュレーション結果をもとに行う分析方法と結果について述べる。そして第7章では分析結果からみる考察について述べる。最後の第8章ではすべての章のまとめを述べる。

1.1 テーマパークの現状

テーマパークは映画、カラオケや旅行など娯楽産業の多様化によって市場の伸び悩みが深刻化しており、長引く不況の影響により経営困難に陥る企業もでてきている。さらに2011年の震災の影響で多くの施設が一時的な休業を余儀なくされている。テーマパークビジネスではリピーターを獲得することが一番の課題であるが、今存在しているテーマパークも集客が難しいという問題に直面し、テーマパークを取り巻く経営環境は年々厳しくなる一方である。そのような状況のなか株式会社オリエンタルランドが経営する東京ディズニーリゾートは好調である。東京ディズニーリゾートでは2012年度の年間入場者数が2750万人を超え、過去最多の待ち時間が400分となるアトラクションも存在するほど人気が高い。新しいアトラクションも続々と増え、東京ディズニーリゾートは今後より一層混雑することが予想される。リピーターを獲得するためには、訪れる人々の満足度を高めることが必須であると考えられるため、本研究は非常に意義のあるものであるといえる。

¹ある特定のテーマをもとに構成された大型レジャー施設。本研究では複数のアトラクションにより構成された複合型のことをいう。

1.2 テーマパーク問題

本研究で対象とするテーマパークは東京ディズニーリゾートに代表されるような複数のアトラクションにより構成されたテーマパークを指す。東京ディズニーリゾートではアトラクションだけでなくパレード、イベントなどを行い空間全体を演出して娯楽を提供するため、来場者の平均滞在時間も長くなる。テーマパーク内では入場者がそれぞれの目的や意志で各自が乗りたいアトラクションを目指して行動する。しかし、アトラクションによって人気度が異なるため、集中的に混雑する場所が生まれてしまう。そしてこの混雑がテーマパーク全体の利用効率を悪化させ、入場者の満足度を妨げる原因にもなりうる。このような混雑の発生に対して入場者の選好や目的を考慮しつつ全体の混雑度を減少させ訪れる人を満足させる方法を考えることをテーマパーク問題とよぶ。

本研究では2012年に「トイ・ストーリー・マニア!」という超人気アトラクションができたばかりの東京ディズニーシーにおけるテーマパーク問題について考える。

1.3 東京ディズニーシーについて

1.3.1 テーマポートとアトラクション

東京ディズニーシーはテーマポートとよばれる7つのテーマにより構成されている。これらは海にまつわる伝説や物語をもとにされている。7つのテーマポートとはメディテレーニアンハーバー、アメリカンウォーターフロント、ポートディスカバリー、ロストリバーデルタ、アラビアンコースト、マーメイドラグーン、ミステリアスアイランドである。待ち時間²からみる上位人気アトラクションは一位から順に、3D映像を駆使したシューティングアトラクションである「トイ・ストーリーマニア!」、地底を駆け抜けるライドコースター「センター・オブ・ジ・アース」、フリーフォールタイプのアトラクションである「タワー・オブ・テラー」、映画『インディ・ジョーンズ』をもとにした魔宮ツアーの「インディ・ジョーンズ・アドベンチャー：クリスタルスカルの魔宮」などがある。上記のなかでも特にアメリカンウォーターフロントは「トイ・ストーリー・マニア!」と「タワー・オブ・テラー」という2大アトラクションが存在しているためよく混雑するテーマポートである。

1.3.2 待ち時間

東京ディズニーシーのテーマパーク内には各アトラクション待ち時間の書かれたボードが掲示されていたり、東京ディズニーリゾートオフィシャルサイトではリアルタイムの待ち時間が発信されている。テーマパークを訪れる人はその掲示ボードを見たり、スマートフォンや携帯電話でオフィシャルサイトにアクセスすることで各アトラクションの待ち時間を知ることができる。またアトラクションに並ぶ人の中からランダムに一人選出して「待ち時間調査

²『ディズニーランドディズニーシー混雑予想カレンダー』<http://www15.plala.or.jp/gcap/disney/realtime.htm>より

カード」というものを配布していることがある。これは私自身も協力したことがあるが、最後に並んだ時にこの調査カードを渡されアトラクションに乗車するときにキャストに返却するというものである。このシステムにより待ち時間を算出していると考えられる。このように東京ディズニーリゾートではアトラクションの待ち時間に対してさまざまな取り組みがされている。

1.4 先行研究

「テーマパークの楽しみ方—シミュレーション分析を通して—」

渡辺（2012）は artisoc というシミュレーションソフトを用いて、東京ディズニーランドをもとにシミュレーションモデルを構築して分析を行い、アトラクションに並ぶ行列人数の変化をみることでテーマパーク内の混雑を緩和させ、入場者を満足させるテーマパークの巡り方を提案することを目的として研究を行っている。

本研究では、東京ディズニーランドをテーマパークのモデルとして取り上げ優先搭乗券³ありと優先搭乗券なしの2つのモデルを作成し、シミュレーション実験を行っている。優先搭乗券は、あるアトラクションで一枚発行するとある一定時間を超えるまでは別のアトラクションの優先搭乗券を発行することができないという特徴をもつ。この券には優先して搭乗できる時間帯が記載されており、この時間を過ぎてしまうと優先搭乗券は無効となってしまう。渡辺では、入場者の平均滞在時間⁴が短いほど効率よくアトラクションを移動したことになると定義している。平均滞在時間は各入場者エージェント⁵があらかじめ定めた満足度の値を上回ると帰宅すると設定している。分析に用いたデータは2つのモデルにおいて、混雑情報所持率を0%～100%まで20%毎に変化させていき、そのときの平均滞在時間と各アトラクションエージェントの行列人数である。

この研究では優先搭乗券ありモデル、優先搭乗券なしモデルのどちらにおいても混雑情報所持率が増加するにつれて、各アトラクションエージェントの行列人数の減少がみられ、混雑情報を所持することがテーマパーク内の混雑度を減少させていると結論づけている。しかし一方で、優先搭乗券なしモデルでは、混雑情報所持率が60%までは混雑情報を所持することで入場者の満足度を高めることができるが、混雑情報所持率が60%以上になると混雑を所持することの効果は期待できなくなっている。一方、優先搭乗券ありモデルでは、混雑情報を入場者が所持すればするほど、満足度を高めることができる。これは混雑情報を所持していても人気のアトラクションに並ぶ時間は変わらないため、優先搭乗券を使って人気のアトラクションに少ない待ち時間並ぶ方が効率よくテーマパークを巡ることができるということを示している。すなわち優先搭乗券のシステムが混雑緩和に大きく貢献していると考えられる。さらに得られたデータを用いて主成分分析を行い、時間帯によって混雑するアトラクションを特定しており、それらのアトラクションを避けることで効率よくテーマパークを巡る方法を

³少ない待ち時間でアトラクションを体験できるシステム。実施アトラクション付近で入手することができる。

⁴各入場者の中で帰宅した人の滞在時間を平均したもの

⁵シミュレーションソフト artisoc において表現される入場者のこと

提案している。渡辺は課題点として、入場者エージェントの潤滑な移動方法の考案、各アトラクションエージェントの選好度の見直し、入場者の休憩や食事時間の考慮などを挙げている。

第2章 研究目的

本研究の目的は、テーマパーク問題を解決するために混雑の発生状況をシミュレーションにより再現し、それをもとに混雑を避け効率よくパーク内を巡る方法を提案することである。本研究では大規模なテーマパークでの利用者の行動をテーマとしており、上記の目的を達成するためには現地調査だけで研究を進めていくことは非常に難しい。研究方法は多々あると考えられるが、本研究の目的は、利用者がテーマパークを効率よく巡ることでテーマパークを楽しみさらに魅力を感じてもらうことである。例えば現地調査では時間や距離的な問題もあり、ほんの数回の調査しかできないため再現性に乏しく、さらに調査者が訪れた日のデータをもとに研究を進めていくことになるため、どの日でも有効であるような効率よくパーク内を巡る方法を断定することはできない。これらの問題を解決するためには利用者の属性やテーマパークの環境などあらゆる場面を想定する必要があると考えられる。そこで本研究では仮想的にテーマパークを作り、その中での利用者の行動をシミュレーションにより再現し、研究を行うことにする。このような研究方法の利点は利用者の人数や入場者の歩行速度などの条件を容易に変更することが可能であり、さらに現地調査よりはるかに多くの回数の実験検証ができることである。そのため本研究ではできる限り多くの試行実験を繰り返すことで、訪れる人が楽しく効率よくテーマパークを巡る方法を提案したいと考えている。また、この最適な巡回方法を提案することができれば、今までテーマパークを訪れることを敬遠していた人たちにもテーマパークに遊びに行きたいと思わせることができると考える。さらにはリピーターをいかに獲得するかが鍵になっているテーマパーク業界にも少ばかりの貢献ができるのではないかと考えている。

本研究では先行研究でのモデルをもとに、入場者エージェントの潤滑な移動方法、各アトラクションエージェントの選好度の見直し、さらにパレード・ショーの時間や船による利用者団体の移動方法などを新たに考案し、より現実的なテーマパークモデルの作成を目指しテーマパーク問題に取り組んでいく。

第3章 研究方法

3.1 研究方法

本研究では、大きく分けて3段階の方法で研究を進めていく。

まず初めにテーマパークの構成に関する調査を行った。シミュレーション上に作成するテーマパークの大きさ、テーマパークを訪れる人間やテーマパークに存在するアトラクションのルールを定義していく。そして次にシミュレーションモデルの構築を行う。本研究では先行研究のモデルをもとにテーマパークを東京ディズニーシーに設定し、先行研究よりさらに現実世界に近いシミュレーションモデルの作成をする。最後にシミュレーション実験の結果と分析を行った。構築したシミュレーションモデルを用いて、条件を様々に変更して複数回試行を行う。得られたデータに対して統計的分析により混雑の様子や利用者の行動の特徴を明らかにする。

3.1.1 artisoc について

本研究では構造計画研究所が開発したシミュレーションソフト artisoc を使用している。artisoc はマルチエージェントシミュレーション (MAS)¹を行うために適しているソフトであり、特に人口社会としての社会現象のモデルを作りシミュレーションを実行することを念頭に開発されたものである。そのため本研究では仮想空間上にテーマパークを設置しその中にいる人の行動を観察することができる。さらにシミュレーションの実行過程のリアルタイムでの観察も容易である。また artisoc は使用するプログラミング言語が他の言語に比べ比較的容易でプログラムの直観的な操作が可能であるため、比較的簡単にシミュレーションモデルを作成することができる。シミュレーション過程や結果、グラフなどの出力も容易に行え、さらに組み込み関数などのサポート機能も充実しているため多少のプログラミングの知識があれば扱いやすいものとなっている。したがって本研究のように大規模なテーマパークでの実験検証において artisoc を用いたシミュレーション実験を行うことは非常に有効な方法である。

¹コンピュータ上に人口社会を構築し、その中に存在する人やモノを動かし相互にどのような影響を及ぼすかを観察すること

3.1.2 artisoc の要素

artisoc は大きく分けるとエージェントと空間という2つの要素で成り立っている。エージェントとはシミュレーションの中で、一定のルールに従って行動・相互作用する人や構造物である。空間とはエージェントや変数を配置するものである。X軸とY軸の座標で表される格子状のマス(セル)で空間を表現しており、配置はこの座標に従う。この artisoc を用いて、仮想的にテーマパークを作成する。

3.2 テーマパークモデルの作成

本研究においては東京ディズニーシーをモデルとしたテーマパークモデルを作成する。



図 3.1: y.park

図 3.1 を参考にしながら,Excel で東京ディズニーシーの形状を再現する。このマップを excelmap とする。この excelmap では人が通ることができる場所(セル)を1,人が通ることができない場所(セル)を0で描き表している。このマップを artisoc で読み込むと人は0の場所は通ることができず,1の部分のみ通ることができるという指示を与えることができる。また, artisoc のテーマパークの大きさは200セル×200セル(縦×横)であり,入場者は1

ステップに1ドット²進むことができる.これは実際の大きさが約800m×約800mであること,時速4kmで人が移動することなどに対応して決定した.

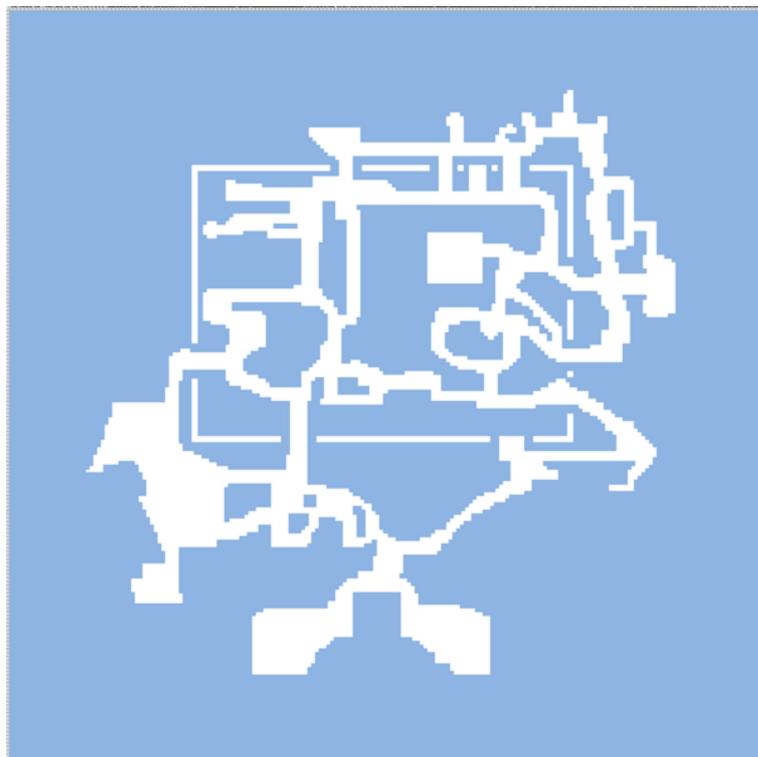


図 3.2: excelmap

²各座標の頂点

3.3 エージェントの構成

y.park は入場者, 観測者, アトラクション, ゲート, 船, パレードという 6 種類のエージェントによって構成されている. これら各エージェントの説明と各エージェントの基本的なルールを以下に示す.

3.3.1 入場者エージェント

本研究では 3 種類の属性ごとに入場者エージェントを作成する.

(1) 入場者エージェント 1

一般的な入場者である. 初めてテーマパークを訪れる入場者も含む.

(2) 入場者エージェント 2

年間パスポートを持っている人や, これまでに 10 回以上訪れたことのある入場者である.

(3) 入場者エージェント 3

夕方 6 時以降アフター 6 パスポートを使って入場する入場者である.

テーマパーク内にはこれら 3 種類の属性を持つ入場者を発生させ, 実際のテーマパーク環境を作成する.

入場者エージェントはアトラクションに乗ると満足度を獲得する. そしてこれらの入場者エージェントはそれぞれにおいて帰宅決定満足度³の値を変えている. 実地調査により, 混雑時に乗ることができるアトラクションの一日の平均数は 5~7 個であると考えられるため入場者エージェントの満足度の値を 5 に設定した. それに伴って入場者エージェント 2 の満足度は 3, 入場者エージェント 3 の満足度は 2 と設定した. またこれらの入場者はテーマパーク内を徒歩だけでなく, 船を使って巡回することもできる. さらにパレードやショーの開始時間にその付近にいる場合は立ち止まって観賞するように設定している.

3.3.2 観測者エージェント

入場者エージェントに加えて, 入場者の行動をみるために観測者エージェントとして 5 つの属性を持つ観測者エージェントを作成する. これらの観測者エージェントはそれぞれの属性において 100 人ずつ生成されるようにした.

- (1) 観測者エージェント 1: 人気のアトラクションを巡る
- (2) 観測者エージェント 2: 空いているアトラクションから巡る
- (3) 観測者エージェント 3: 距離が遠いアトラクションから巡る
- (4) 観測者エージェント 4: 距離が近いアトラクションから巡る
- (5) 観測者エージェント 5: パレードやショーも楽しんで巡る

³入場者の満足度がこの値を超えると入場者は帰宅する. 後に詳しく述べる.

観測者エージェントは5パターンの行動方法をとる。観測者エージェント1についてはアトラクション番号2,3,5,8,9,24のアトラクション効用値⁴のみ適用するように設定し、それ以外のアトラクション効用値を0とすることで人気のアトラクションのみを巡るようにした。観測者エージェント2においては通常のアトラクション効用値の値を反転させた。すなわち上記で挙げたアトラクション効用値の値を下げ、比較的すいているアトラクション効用値の値を上げることで対応した。観測者エージェント2はすいているアトラクションから巡るため比較的多数のアトラクションに乗ることができると考えられる。そのため帰宅決定満足度の値を8と設定した。次に観測者3と観測者4であるが、アトラクションを目指して行動する際のルールにおける距離の比重を変えることで設定した。この行動ルールは総合効用値という値にしたがって決定されるが、総合効用値についての説明は後述する。最後に観測者エージェント5はパレードやショーが開始される時間にその付近にいる場合、立ち止まって観賞するというルールを加えた。これら5パターンそれぞれの行動を観察し、最適な巡回パターンを提案する。

3.3.3 入場者の設定

入場者はy.park上のゲートから発生させる。入場者の行動ルールは以下の通りとする。また東京ディズニーシー(2013年時点)ではファストパス発券可能なアトラクションを8つ設定している。そのためy.parkでも同様に、8つのアトラクションを優先パスの発券が可能なアトラクションとして指定することにした。ファストパス発券可能なアトラクションでは少ない待ち時間でアトラクションに乗ることができるファストパス専用ルートと、ファストパスを持っていない入場者が並ぶ通常ルート(東京ディズニーシーではこのことを「スタンバイ」という)が存在する。

入場者はファストパスを所持しているとファストパスの発券を受けてから750ステップ以上1500ステップ以下のとき搭乗するアトラクションがファストパス発券可能なアトラクションかどうかを判断し、この条件を満たすときファストパス専用ルートに並ぶことができる。しかし1500ステップを超えると優先パスは無効となってしまう。

⁴アトラクション人気度

入場者の行動ルール

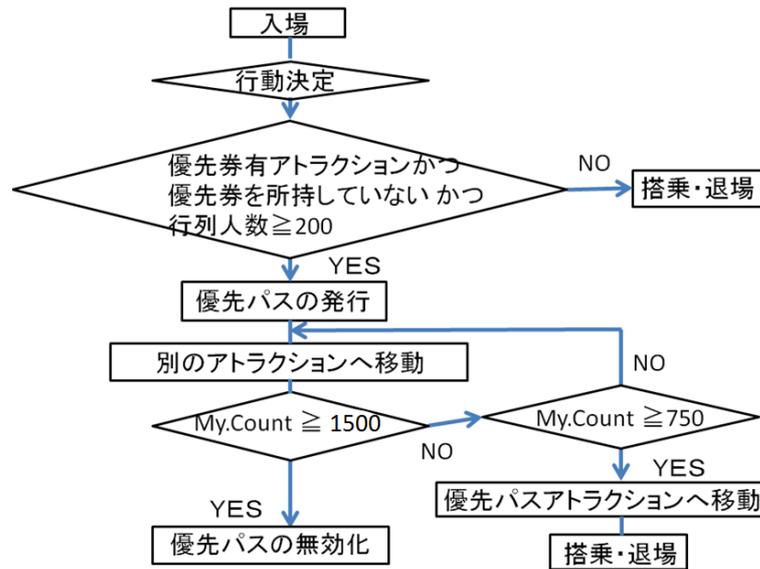


図 3.3: 入場者エージェントフローチャート

- Step1** 入場者が入場する.
- Step2** アトラクションを選択し, 移動する.
- Step3** アトラクションに到着する.
- Step4** (優先搭乗券を所持していない) かつ (優先搭乗券発券可能なアトラクションである) かつ (行列人数 > 50) であるなら **Step5** へ. そうでなければ **Step14** へ.
- Step5** 750 ステップ後の優先搭乗券の発券を受ける.
- Step6** 他のアトラクションを選択し移動する.
- Step7** 入場者数が最大収容数以下のとき **Step8** へ.
最大収容者数より多ければ行列に並ぶ.
- Step8** アトラクションに搭乗し, 退出する.
- Step9** 帰宅決定満足度が満足度以下なら **Step17** へ. そうでなければ **Step10** へ.
- Step10** 優先搭乗券発券を受けてからのステップ数が 1500 以上であるなら **Step11** へ.
そうでなければ **Step6** へ.
- Step11** 優先搭乗券を無効化する.**Step16** へ.
- Step12** 優先搭乗券の発券を受けてからのステップ数が 750 以上であるなら **Step13** へ.
そうでなければ **Step6** へ.
- Step13** 優先搭乗券の発券を受けたアトラクションに戻り, 優先的な行列に並ぶ.
- Step14** 入場者数が最大収容数以下のとき **Step15** へ, 最大収容数より多いとき行列に並ぶ.
- Step15** 優先搭乗券所有者の場合は優先搭乗券を無効にし, アトラクションに搭乗し退出する.

Step16 帰宅決定満足度が満足度以下なら **Step17** へ. そうでなければ **Step2** へ.
Step17 入場者は退場する.

3.3.4 優先方向について

本モデルのように, 複雑な経路に複数人のエージェントを同時に動かすモデルにおいては, エージェントを動かすルールが複雑になりがちである. そこで, 本モデルでは優先方向という概念を取り入れる. まず各入場者が目的地へ向かって移動する際に, 障害物となる建物や壁がある場合, そのとき向いている方向を優先方向として記憶する. この記憶は保持され, FILO(First, In, Last, Out) にしたがって記憶が出し入れされ目的地へ進んでいくという仕組みである. なお FILO についての詳細な説明は付録に記載しておく.

FILO の仕組みを利用した入場者エージェントの移動ルールは以下の通りとする.

優先方向のルール

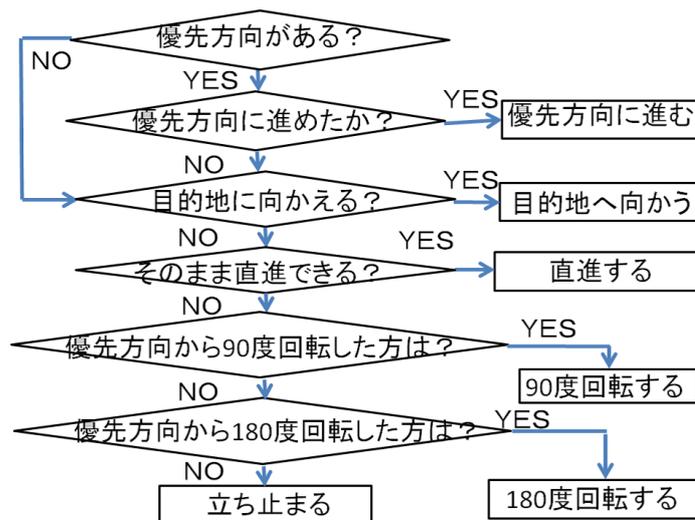


図 3.4: 優先方向フローチャート

Step1 優先方向があるなら **Step2** へ. ないなら **Step3** へ.

Step2 優先方向に進めるなら優先方向に進む. 進めないなら **Step3** へ.

Step3 目標アトラクションに向くことができるなら目標アトラクションへ向かう. できないなら **Step4** へ.

Step4 目標アトラクションを向いた方向にそのまま直進することができるなら直進する. できないなら **Step5** へ.

Step5 優先方向から 90 度回転した方向に進めるなら 90 度回転した方向へ進む. 進めない

なら **Step6** へ.

Step6 優先方向から 180 度回転した方向に進めるなら 180 度回転した方向へ進む. 進めないなら **Step7** へ.

Step7 立ち止まる.

3.3.5 入場者エージェントの持つ属性と情報について

各入場者は以下の属性と情報を持つとする.

現在の状況

入場者エージェントの行動ルールとして 3 つの状況を設定する. まず現在の状況が 1 なら目標アトラクションへ移動する. 目標アトラクションとは入場者の行動の目的地となるアトラクションのことを指す. 現在の状況が 2 ならアトラクションエージェントの中に滞在している. すなわち入場者がアトラクションの行列に並び, アトラクションに乗り込んでいる状況である. 行列がなければそのままアトラクションに乗り込み退出する. 最後に現在の状況が 3 なら帰宅する. この場合, 入場者エージェントはゲートを目指して歩いている状態である. 現在の状況が 3 になる条件は入場者の満足度が帰宅決定満足度を超えたとき, あるいは現在のステップ数が 10000 ステップを超えたときである. テーマパーク内にいる入場者が 10500 ステップ目に閉園ゲートをくぐることができるように帰宅の命令を出すステップ数を 10000 ステップと定めた.

このように入場者エージェントはこの 3 つのうちいずれかの状況から行動選択をする.

各アトラクション選好度

各アトラクションにあらかじめ選好度を設定する. 選好度は 2012 年度の東京ディズニーシーの全アトラクションの待ち時間 1 年間分を平均したものを各々集計した値及びインターネットにおいてアンケート調査を行った値に基づいて決定する. なお, 値は 0.0~1.0 の間で設定している.

混雑情報

各アトラクションに並んでいる行列人数とその待ち時間の情報である. 実際に東京ディズニーリゾートでもモバイルサイトでリアルタイム待ち時間が発信されている. 本モデルでは, 入場者は携帯電話およびスマートフォンでこのサイトを閲覧することができることを想定している.

満足度

搭乗したアトラクションの効用値に基づいて決定される。アトラクションに乗ると満足度にアトラクション効用値の値が随時足される。人気のアトラクションほどアトラクション効用値が高いため、人気のアトラクションにたくさん乗れば満足度の値はすぐに高くなる。一方、あまり人気のないアトラクションでは数を多く乗らなければ満足度はなかなか高くない。

帰宅決定満足度

帰宅するために基準となる満足度の数値である。入場者はこの値を超えるとゲートへ向かい帰宅するという設定をした。

3.3.6 入場者のアトラクションへの行動方法

次に、入場者がどのように目標アトラクションを決めるかについて説明する。入場者は入場したとき及びアトラクションを退出するときに、目標アトラクションを決定するか、帰宅するかを選択する。これを行動決定とし、このルールは以下の総合効用値⁵に基づいて決定される。総合効用値に関して [1] を参考に以下の定義式を示す。

$$\text{総合効用値} = \text{アトラクション効用値} - \alpha \times (\text{現在いる場所からアトラクションまでの距離}) - \beta \times (\text{アトラクションの行列人数})$$

[1] 式から、総合効用値が最も高いところを選択して入場者は目標アトラクションを決定し、行動する。また帰宅する場合は、全てのアトラクションエージェントに搭乗するか、満足度が帰宅決定満足度を上回った場合とする。

3.3.7 入場者エージェントにおいて使用する変数・関数

本モデルでは入場者エージェントの変数を 27 個作成する。変数の中には混雑情報所持率、退場数、平均滞在時間などが含まれる。入場者エージェントで使用する変数および組み込み関数、ユーザー定義関数の詳細は付録にまとめて記載する。

3.3.8 アトラクションエージェント

実際の東京ディズニーシーを参考に 26 個のアトラクションエージェントを作成した。以下にアトラクションエージェントのルールを述べていく。また 8 つの優先パス発券可能なアトラクションはアトラクション番号 2,3,6,8,9,14,24,25 のアトラクションである。

⁵アトラクション効用値、アトラクションまでの距離および混雑情報の所持の有無に基づいて決定される。 α 、 β の数値を変えることで入場者の行動ルールを変更することができる

3.3.9 アトラクションの設定

アトラクションのルール

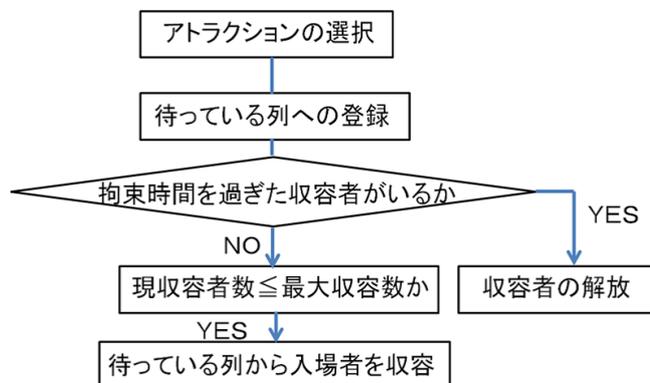


図 3.5: アトラクションエージェントフローチャート

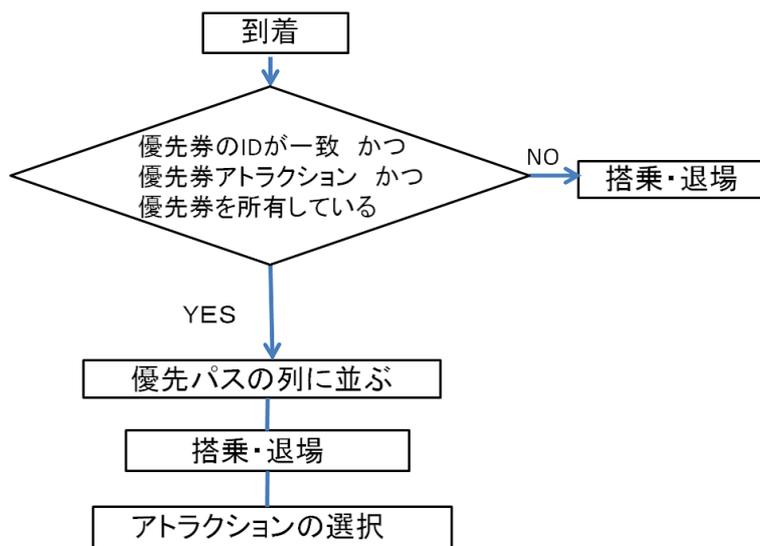


図 3.6: ファストパスアトラクションエージェントフローチャート

- Step1 アトラクションエージェントを選択する。
- Step2 アトラクションエージェントへ到着した後、「待ち行列」へ登録する。
- Step3 「拘束ステップ数」を過ぎた収容者がいるなら Step4 へ。いなければ Step5 へ。
- Step4 収容者を解放する。Step5 へ。
- Step5 収容者数 ≤ 最大収容者数なら Step6 へ。そうでないなら Step9 へ。
- Step6 (優先券所持者 > 0) かつ (ランダム値 ≤ 0.8) なら Step7 へ。そうでないなら Step8 へ。

Step7 「待ち行列」から最も「待ちステップ数」が大きい優先搭乗券所持者をアトラクションへ収容する.**Step5** へ.

Step8 「待ち行列」から最も「待ちステップ数」が大きい優先搭乗券所持者以外の入場者をアトラクションへ収容する.**Step5** へ.

Step9 他のアトラクションに対して **Step2** から **Step5** を繰り返す.

3.3.10 アトラクションの持つ属性と情報

各アトラクションは以下の属性と情報を持つとする.

待ち行列

各アトラクションに並んでいる入場者の数.

待ちステップ数

待ち行列に並んでいるステップ期間(アトラクション到着時を0とする)を意味する.待ちステップ数が大きい入場者エージェントからアトラクションエージェントに搭乗する.

拘束ステップ数

入場者がアトラクションに搭乗しているステップ数.実際の各アトラクションの所要時間を参考にしている.

最大収容数

各アトラクションエージェントが収容することのできる最大人数のことを指す.実際のアトラクション最大定員を参考にしている.

収容者

各アトラクションエージェントに搭乗している入場者エージェントの数を指す.

3.3.11 アトラクションエージェントにおいて使用する変数・関数

本モデルではアトラクションエージェントの変数を10個作成する.アトラクションエージェントの変数には最大収容数・拘束時間・待ち行列・収容者・待ちステップ数などが含まれる.アトラクションエージェントで使用する変数および組み込み関数,ユーザー定義関数の詳細は付録にまとめて記載する.

3.3.12 ゲートの設定

「入場者エージェントの生成」と「入場者エージェントの退場」を行う場所である。本モデルではゲートエージェントを1つ作成した。本研究の目的は混雑状況における入場者の行動を観察することであるため入場者の生成については時間帯を考慮する必要がある。なぜなら入場者の数を毎ステップ一定の割合で生成し続けると退場者の数より多くなってしまい、テーマパーク内に入場者があふれることになってしまうからである。したがって以下のように生成するとする。

- (1) 1ステップ⁶：初期入場者として1ステップ目に入場者を3000人、入場者2を1000人生成する。
- (2) 入場者は5/16の確率で一定数ずつ生成される。生成確率の値は先行研究を参考に定めたものである。
- (3) この間は入場者は生成されないようにする。3750ステップはちょうど13:00にあたるため、この時間帯以降からテーマパークに入場する入場者は少ないと考えた。そこで本研究のシミュレーションモデルでは入場者の人数調整のため生成の一時停止時刻を13:00と設定した。
- (4) 東京ディズニーシーにはアフター6という入場パスポートが存在する。アフター6パスポートは通常よりも価格が安価なため18:00以降からの入場する入場者も多い。そこで18:00にあたる7500ステップ目に500人を生成することで対応した。

そして現在の状況が3、すなわち帰宅の状態にある入場者がゲートに向かうと入場者は消滅するように設定した。

3.3.13 船エージェント

東京ディズニーシーでは時計回りに各テーマポートを経由しながらパークを一周する「ディズニーシー・トランジットスチーマーライン」という小型蒸気船のアトラクションがある。入場者はパークの景色を船から眺めることができるのはもちろん、テーマポート間の移動手段としても使うことができる。y.parkでも同様に船エージェントとして、テーマポート間を移動するエージェントを作成した。船は図3.7の赤色の部分を運行する。緑色の楕円形の場所は船の乗降場である。乗船する入場者の行動はわかりやすくフローチャートでまとめた。

⁶シミュレーション開始時を1ステップとする

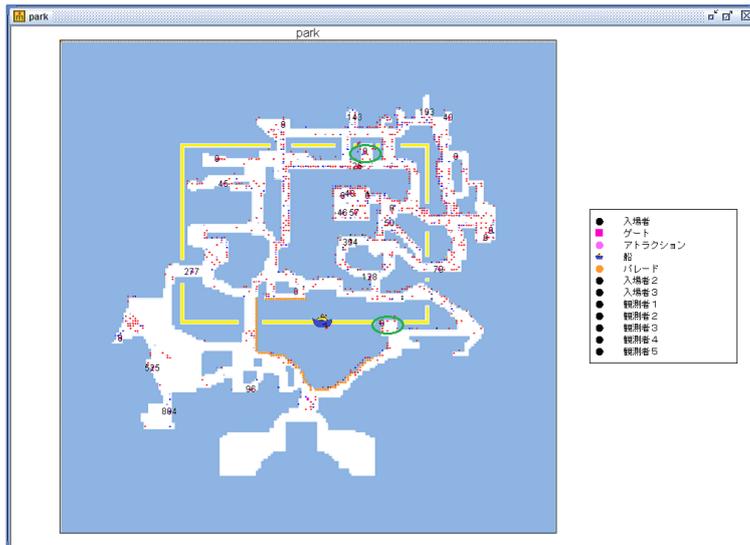


図 3.7: 船エージェントが進む様子

入場者は船の乗り場に到着すると入場者のもつ ride という変数が 1 になる. つまり ride=1 のとき入場者は乗船している状態である. 「船に乗る」という状態は, ride=1 のとき常に入場者の x 座標と y 座標が船の x 座標と y 座標の周辺が変わるというルールを書き, 入場者と船が共に動くようにして表現した. なお船は入場者 20 人以上を乗船客として捉えたと進むように設定している. 乗り場に到着すると, 乗船していた入場者をその場で降船させまた新たな乗船客が集まるまでその場で待機する.

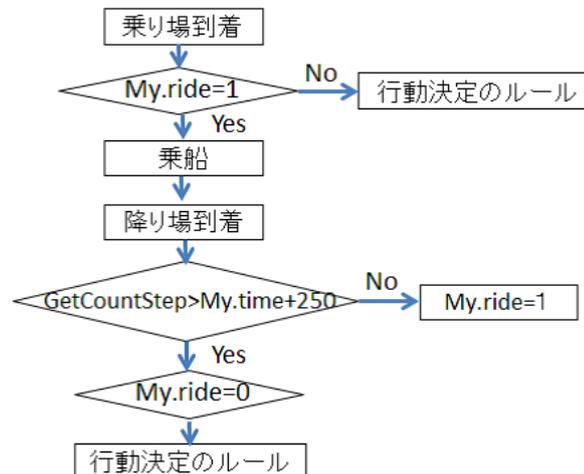


図 3.8: 乗船のルール

3.3.14 パレードエージェント

東京ディズニーシーでは一日に数回メディテレーニアンハーバーの中央にある大きな海でショーが行われる。y.park ではパレードを一日に5回開催すると設定した。パレードエージェントは海の周囲を囲むように設置した。入場者はパレードが開催する時間帯にパレードエージェントの近くにいるとパレードを観賞するために立ち止まる。パレードの開催時間はそれぞれ1125ステップ、3000ステップ、4812ステップ、7375ステップ、8937ステップからとし、10～20分の上演とする。時間および上演時間は実際の東京ディズニーシーをを参考にした。パレードを観賞していた入場者はパレードが終わるとまたもとの通り動き出す。

3.3.15 Universe のルール

Universe ではモデル全体を動かすためのルールを作成する。ここに書き込まれたルールはモデル全体の変数に適用する。主に入場者、観測者の生成ルールや行動決定のルールが書かれている。また滞在時間や帰宅者などのデータをファイルとして出力するためのルールなどもここで設定する。

3.3.16 Universe において使用する変数・関数

本モデルでは Universe の変数を8個作成した。モデル全体の退場数・平均滞在時間などの属性を変数で指定する。Universe で使用する変数と組み込み関数の詳細は付録に記載する。

第4章 シミュレーション実験

本研究では入場者および観測者の混雑情報所持率を60%に固定し、シミュレーション実験を行う。この60%という値は先行研究 [1], [2] で混雑情報が60%を超えると混雑情報の所持は入場者の満足度を高める効果に関係しないという結果をふまえて設定した。

また1日の行動シミュレーションを想定しているため本研究では東京ディズニーシーの混雑時の営業時間に合わせて8:00~22:00の14時間(10500ステップ)を1回とするシミュレーションを行う。データは時間帯ごとの各アトラクションの行列人数と観測者エージェントの平均滞在時間のデータをファイルに出力するというかたちで収集した。試行回数は45回である。また観測者5パターンの滞在時間のちがいをみるために、観測者5パターン一人ひとりごとの滞在時間のデータを集めるために5回の試行を行った。これにより1474人分のデータを収集した。

図4.1はシミュレーションモデルの実行画面である。

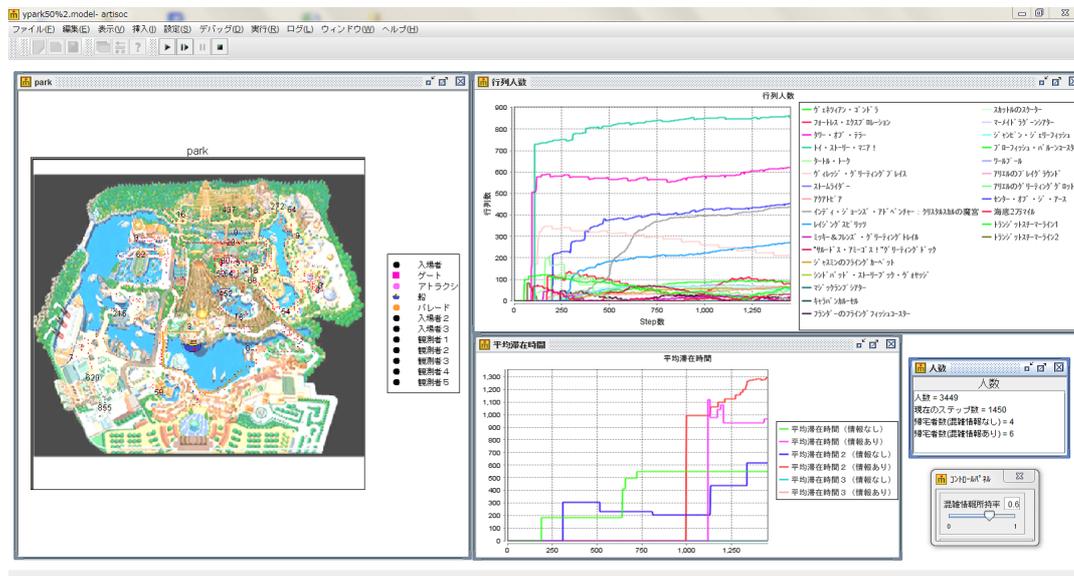


図 4.1: シミュレーション実行画面

第5章 シミュレーション結果

図 5.1 はシミュレーション実験により得られたデータのうち、各アトラクション行列人数の平均を表¹にしたものである。

name	M	A	E	N
0.ゴントラ	78	42	1	1
1.フォートレス	6	2	0	0
2.タワテラ	551	733	706	580
3.トイマニ	764	813	732	722
4.タートル	0	0	0	0
5.ダッフィー	264	42	0	0
6.ストーム	1	0	0	0
7.アクアヒュー	42	67	9	4
8.インディ	333	559	549	496
9.レイシング	214	418	419	402
10.ミキミクグフィ	12	15	0	0
11.トナルト	59	19	0	0
12.ジャズミン	41	61	5	1
13.シントハット	0	0	0	0
14.マジック	0	0	0	0
15.キャラハン	0	0	0	0
16.フランダー	11	1	0	0
17.スカットル	69	74	6	2
18.マーメイト	0	0	0	0
19.ジェリーフィッシュ	66	63	1	1
20.ハルーン	71	73	11	3
21.ワールプール	73	74	10	4
22.アリエルフレ	3	0	0	0
23.アリエルクリ	28	22	0	0
24.センター	404	630	645	594
25.2万マイル	65	45	0	0
26.トランジット1	0	0	0	0
27.トランジット2	0	0	0	0

図 5.1: アトラクション行列人数

¹M は朝 (9:00), A は昼 (13:00), E は夕 (17:00), N は夜 (20:00) とする

図 5.2 は観測者別の平均滞在時間を棒グラフで表したものである。

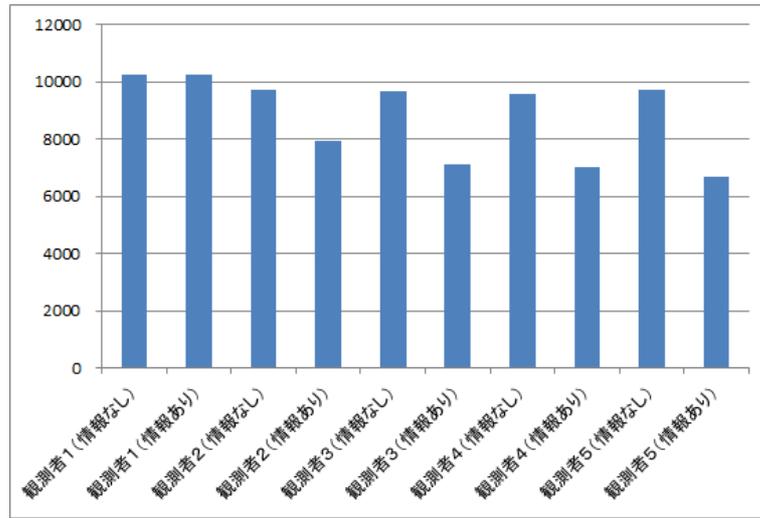


図 5.2: 観測者別平均滞在時間

第6章 分析・結果

シミュレーション結果から得られたデータを用いて主成分分析と分散分析を行う。

6.1 主成分分析

混雑情報所持率 60 %の行列人数と時間帯による主成分分析を行う。時間帯は開園してからすぐの 750 ステップ (9:00) 時, お昼過ぎの 3750 ステップ (13:00) 時, 夕方の 6750 ステップ (17:00) 時, 閉園時間 2 時間前の 9000 ステップ (20:00) 時の時間でみた。

6.1.1 全アトラクションにおける主成分分析

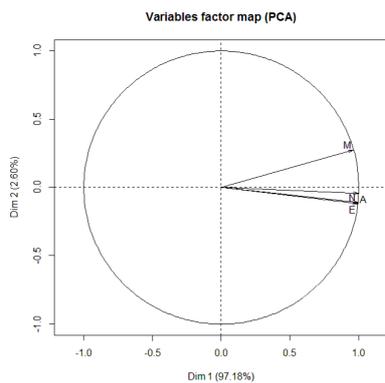


図 6.1: 全アトラクションバイプロット

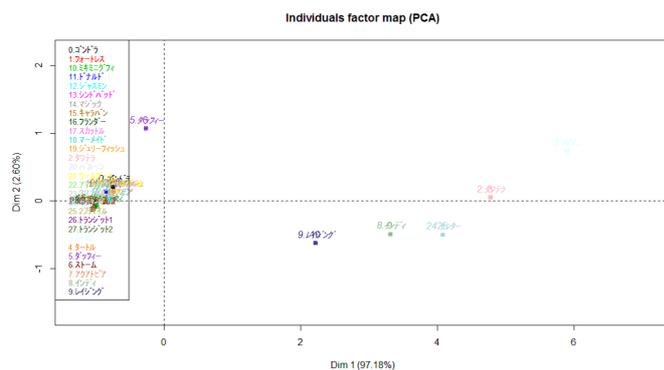


図 6.2: 全アトラクション散布図

6.1.2 人気アトラクションにおける主成分分析

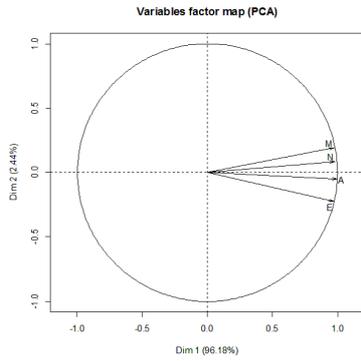


図 6.3: 人気アトラクションバイプロット

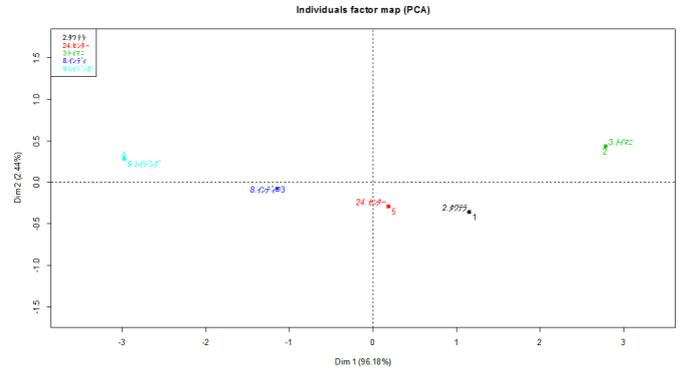


図 6.4: 人気アトラクション散布図

6.1.3 人気アトラクションを除いたアトラクションにおける主成分分析

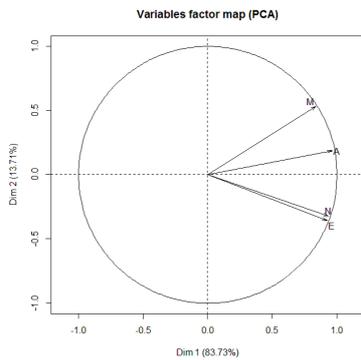


図 6.5: 人気アトラクション以外のバイプロット

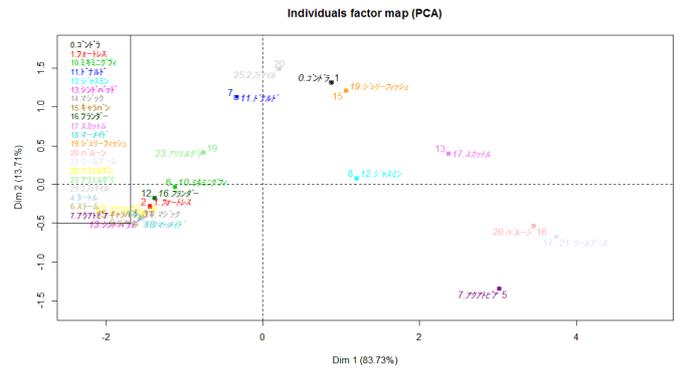


図 6.6: 人気アトラクション以外の散布図

人気アトラクションの待ち時間推移は以下の図の通りである。

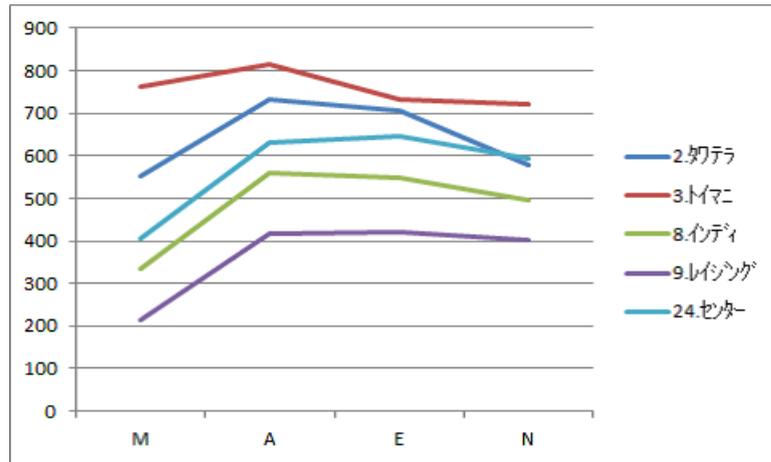


図 6.7: 人気アトラクションの折れ線グラフ

6.2 分散分析

観測者 1 から観測者 5 の滞在時間について一元配置の分散分析を行った. その差が有意であるとみられたため, 多重比較を行った. スマートフォンや携帯電話が普及している世の中でテーマパークを訪れる際に混雑情報を所持していないことはあまり現実的ではないということ を考慮し, 混雑情報を所持している観測者のデータのみを使用した. 多重比較の結果は以下の図のようになった.

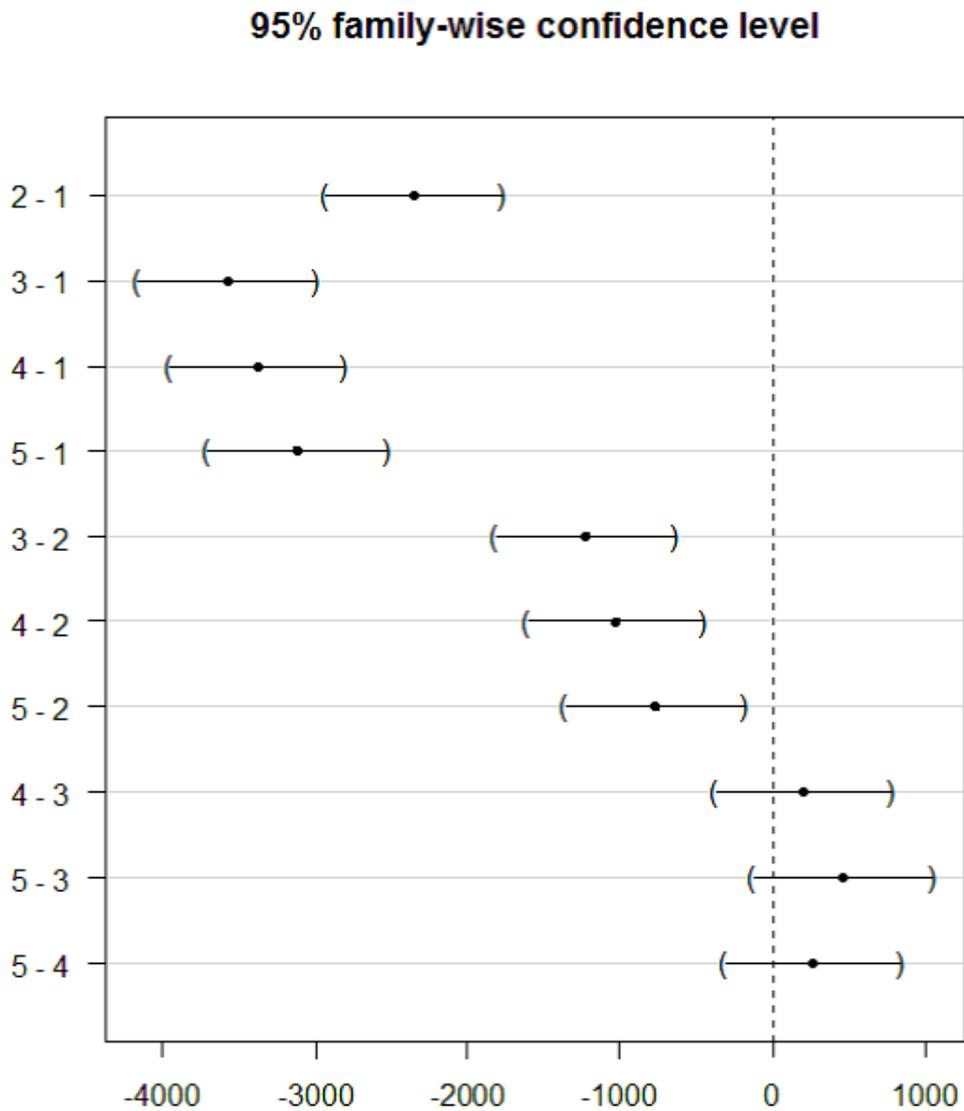


図 6.8: 分散分析による多重比較

第7章 考察

アトラクション行列人数について

主成分分析の結果からわかる人気アトラクションの特徴について以下にまとめる。

【アトラクション番号2】

朝・昼・夕・夜すべての時間帯において行列人数が多い。特に朝の時間帯が混雑している。

【アトラクション番号3】

昼・夕の時間帯に行列人数が増加する。朝・夜の時間帯は比較的行列人数は少ない。

【アトラクション番号8】

昼・夕・夜に行列人数が増加する。朝の時間帯は比較的行列人数は少ない。

【アトラクション番号9】

昼・夕・夜で行列人数は一定して多い。朝の時間帯は比較的行列人数は少ない。

【アトラクション番号24】

昼・夕・夜に行列人数が増加する。

アトラクション番号2は人気アトラクションの中でも特に人気が高く、朝の時間帯には一番混雑しているアトラクションである。その一方アトラクション番号8、アトラクション番号9は朝の時間帯が比較的に行列人数は少ない。この2つのアトラクションはゲートから最も距離が遠いところに位置している。

観測者滞在時間について

図5.2からどの観測者においても、混雑情報を所持している観測者の方が混雑情報を所持していない観測者より滞在時間が短いということがわかる。混雑情報を所持していない観測者は5パターンにおいてあまり差はみられなかったため、テーマパーク内を効率よく巡るためには混雑情報を所持する、つまりスマートフォンや携帯電話などでアトラクションの待ち時間を随時調べながらアトラクション巡りをすることが絶対条件であるといえる。

分散分析の結果から、混雑情報をもっている観測者を比較すると観測者1と観測者2のアトラクションの巡り方はあまり有効ではなく、観測者3、4、5のアトラクションの巡り方が効率の良い方法であると考えられる。観測者1は滞在時間が最も長く、10000ステップを超えていることから観測者1はアトラクションに数多く搭乗し満足して帰宅したのではなく、10000ステップ経過した場合の帰宅命令により帰宅したことがわかる。観測者2は比較的

行列人数が少ないアトラクションを巡るため満足度が上がりにくい。したがって人気のアトラクションにのみこだわってパーク内を巡る方法や行列人数が少ないアトラクションのみを巡る方法は有効な方法ではないと考えられる。

また観測者1と観測者3を比較した場合の差が一番大きいという結果であったため、ゲートからの距離が遠いアトラクションから巡る観測者3の巡回方法が比較的効率の良い方法ではないかと考えられる。しかし、観測者3と観測者4、観測者5を比較したときにあまり差がみられなかったことと、観測者1と観測者4を比較したときの差が観測者1と観測者3を比較したときの差に次いで大きいため、観測者3の次に有効なのは観測者4の行動パターンであると考えられる。ゲートから近いアトラクションにはアトラクション番号2やアトラクション番号3のような人気アトラクションが存在するためであろう。また観測者5はアトラクションにこだわりがないため、うまく行列人数の少ないアトラクションや人気のアトラクションに搭乗することができていると考えられる。

以上の結果、人気のアトラクションに乗ることはできるだけ短時間で入場者の満足度を上げるための必要条件となっている。

最適な巡回方法

今回のシミュレーションモデルの結果と統計的な分析結果から、入場者が満足してテーマパークを巡る戦略として3つの行動パターンを提起する。まず、観測者3のゲートからの距離が遠いアトラクションから巡る方法である。このパターンでは朝の行列人数が少ない時間帯にアトラクション番号8とアトラクション番号9の2つのアトラクションを目指して行動し、一日のうちのできるだけ早い時間帯にこの2つの人気アトラクションに搭乗する。そして夕方から夜にかけてアトラクション番号2, 3, 24のような他の人気アトラクションの待ち行列に並ぶ。昼の時間帯はその他は比較的待ち時間が少ないアトラクションを混雑情報と照らし合ながら巡っていく。2つ目の方法は観測者4のゲートからの距離が近いアトラクションから巡る方法である。アトラクション番号2, 3のような人気アトラクションを目指して行動して搭乗し、できるだけ早い時間帯に満足度を引き上げる。その後混雑情報によるアトラクションの待ち時間を頼りに行列人数が比較的少ないアトラクションを巡る方法である。最後に観測者5のアトラクションにこだわらずテーマパークを巡る方法である。これは上述のとおり、アトラクションにこだわりがないため有効に行列人数の少ないアトラクションや人気のアトラクションに搭乗し、満足度を上げていると考えられる。

人気のアトラクションは昼・夕方の時間帯に行列人数が増加する傾向にあるため、比較的行列人数が少ない朝あるいは夜の時間帯に人気アトラクションに並ぶことが有効な方法であると考えられる。

第8章 おわりに

本研究ではシミュレーションソフト artisoc を用いて仮想空間上にテーマパークを作成し、テーマパーク問題について考え入場者が満足する効率のよいパークの巡り方について考えた。今回のシミュレーションではテーマパークを効率よく巡るための有効な方法として3パターンの巡回方法を提案することができた。

本モデルではアトラクションの人気度を実際の待ち時間をもとに算出したが、「人気アトラクション」の定義をもっと明確にした上で現地調査などで正確に算出したり、入場者が搭乘するアトラクションをどのように選択するかということをはっきりとさせて総合効用値の値を決定してシミュレーション実験をするとまた違った結果が得られることも考えられる。また今回は5つの行動パターンの観測者を作成し観察したが、他の行動パターンを新たに考案し追加させたり、行動パターンを組み合わせることでパターン数を増やすなどといったシミュレーションモデルを詳細までこだわって作ることで今回の研究よりもさらに効率的なアトラクションの巡り方を発見することができるかもしれない。

本研究を通して粘り強くプログラミングを構築する力と現実世界を仮想空間上に作り出すためにアイデアを変換させる力を身につけることができた。今後本研究をきっかけに、一人でも多くの人に夢の国でより一層楽しんでもらうことはもちろん、さらに東京ディズニーシーのテーマパーク作りにも貢献されることを願ってやまない。

謝 辞

本論文の作成にあたり, 多大なるご指導を賜りました大田靖助教, 山内信幸教授に深く御礼申し上げます. またプログラミングの知識がほとんどなかった私が途中で諦めることなく本研究に取り組むことができたのは, 共に学んだ広域数理研究室のメンバーはもちろん, 卒業研究にて共に artisoc に取り組んだメンバーからの支え, 励ましがあったからです. また, 同研究室 OB の藤井さん, 渡辺さんからもアドバイスをいただきました. 本論文の作成に関わってくださった皆様に感謝しています. 本当にありがとうございました. 最後になりましたが, 卒業に至るまでご指導いただいた諸先生方に心から感謝と御礼を申し上げたく, 謝辞にかえさせていただきます.

参考文献

- [1] 渡辺文“テーマパークの楽しみ方—シミュレーション分析を通して—”, (2012年度 同志社大学文化情報学部卒業論文)
- [2] 刀根哲也・小原和博:“テーマパークでの混雑情報と優先搭乗パスの効果に関するマルチエージェントによる検討” 知能と複雑系, 131 巻 5 号, 2007 年
- [3] 山影進『人工社会構築指南』(書籍工房早山, 2007 年)
- [4] 兼田敏之『artisoc で始める歩行者エージェントシミュレーション』(構造計画研究所, 2010 年)
- [5] 鈴木哲『東京ディズニーシー完全ガイド』(講談社, 2012 年)
- [6] オリエンタルランドグループ「東京ディズニーリゾート入園者数データ」
<<http://www.olc.co.jp/tdr/guest/1>> (2013/12/15 アクセス)
- [7] 『ディズニーランドディズニーシー混雑予想カレンダー』
<<http://www15.plala.or.jp/gcap/disney/realtime.htm>> (2013/12/15 アクセス)
- [8] Insider's Computer Dictionary スタック <<http://www.atmarkit.co.jp/icd/root/56/5783656.html>>(2013/12/18 アクセス)

付録A FILO

A.1 FILO

コンピュータにおける基本的なデータ構造の1つである。最後に入れたデータが最初に取り出される（最初に入れたデータが最後に取り出される）ようなデータ構造。概念的には、データを積み木のように積み上げる操作に例えられる。すなわちデータを上へ上へと順番に積み上げていく。この構造からデータを取り出す場合は最後に置いたデータ（一番上のデータ）が最初に取り出される。一番最初に置いたデータ（一番下のデータ）を取り出すにはその上にあるデータをすべて取り出してからでないと取り出すことはできない。

これを分かりやすくまとめると以下になる。

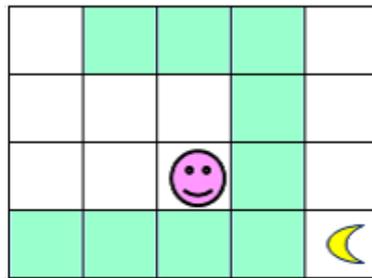


図 A.1: フローチャート 1

ピンクのエージェントが黄色の目的地へと移動するために右または下へ行こうとするが、緑の障害物が移動するのを妨げているので上へ移動する。「右」「下」は優先方向として一番下の段に保存される。（図 A.1 を参照）

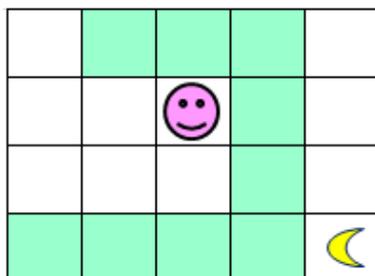


図 A.2: フローチャート 2

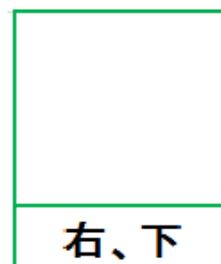


図 A.3: スタック 1

保存されている「右」「下」にはまだ移動することができないためさらに上に行こうとす

る. しかしまた緑の障害物に阻まれて行くことができないので「上」がさらに追加して保存される. そこでピンクのエージェントは左に移動する. (図 A.2, 図 A.3 を参照)

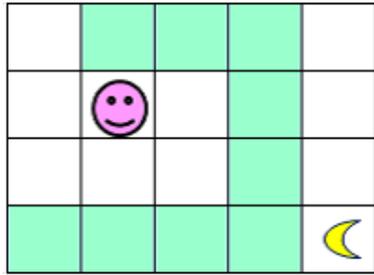


図 A.4: フローチャート 3

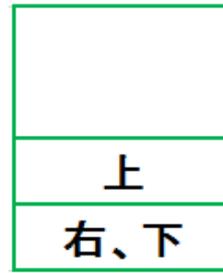


図 A.5: スタック 2

左へ移動したあと下へ移動することができるようになったが,「右」「下」が「上」よりも低い場所に保存されているので下へは移動できない. そのためさらに左へ移動する. (図 A.4, 図 A.5 を参照)

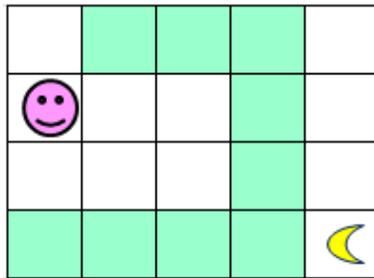


図 A.6: フローチャート 4

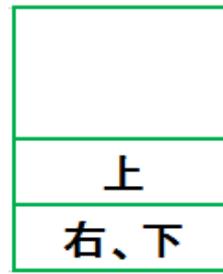


図 A.7: スタック 3

保存されている「上」への移動が可能なので, 上へ移動する. 優先方向ストックは「右」「下」が1番上の段になる. (図 A.6, 図 A.7 を参照)

付録B 使用した変数・関数

B.1 変数表

表 B.1: Universe 変数

変数名	動作
アトラクション数 (整数)	遊園地の中に存在するアトラクションの数. 本研究では 40 とする.
混雑情報所持率 (実数)	各入場者のどれくらいの割合が混雑情報を所持しているか表したもの. 0~1 の中で変化する.
退場数 (整数)	混雑情報を所持していない入場者が, ゲートから退出して帰宅する数
総滞在時間累計 (整数)	混雑情報を所持していない各入場者が, ゲートから入場してゲートから退場するまでの遊園地に滞在していた時間
平均滞在時間 (実数)	混雑情報を所持していない入場者の, 総滞在時間累計を退場数で割ったもの
退場数 (情報あり)(整数)	各アトラクションに並んでいる入場者の集団
総滞在時間累計 (情報あり)(整数)	各アトラクションに現在収容されている入場者の集団
平均滞在時間 (情報あり)(実数)	現収容入場者の人数を数えたもの
ファストパス所有者	優先搭乗パスを持っている入場者

B.2 変数表

表 B.2: 入場者エージェント変数 1

変数名	動作
ID	各エージェントの配列番号
X	空間の x 座標
Y	空間の y 座標
Layer	階層
Direction	各入場者が向く方向
現在の状況 (整数)	1, 2, 3 と設定する. 目標アトラクションへ移動することを 1, アトラクションで行動することを 2, 帰宅することを 3 とする.
目標アトラクション (エージェント)	各入場者が目指すアトラクション
待ち時間 (整数)	各アトラクションに登場するまでに並ぶ時間
滞在時間 (整数)	各入場者が遊園地にいる時間
アトラクション効用値 (実数)	各入場者がゲートから入場して, まず初めに目指すアトラクション. ランダムに 1~40 のアトラクションに振り分けられる.
混雑情報 (整数)	各アトラクションに並んでいる人数
未訪問フラグ (ブール)	実数値の少数部分を切り捨てて, 整数化する
表示色 (整数)	各入場者を表す色. 混雑情報を所持している人を赤色, 混雑情報を所持していない人を青色にする.
優先方向 (整数)	各入場者が優先的に向かう方向
PRex(整数)	各入場者が前にいた場所の x 座標
PRey(整数)	各入場者が前にいた場所の y 座標
CURX(整数)	各入場者が現在いる場所の x 座標
CURY(整数)	各入場者が現在いる場所の y 座標
入場ステップ数 (整数)	各入場者がゲートから入場してから経過しているステップ数
情報フラグ (ブール)	各入場者が混雑情報を所持しているかいないか
満足度 (実数)	既訪問アトラクションの人気度の合計
帰宅決定満足度 (実数)	0.3 と設定する. 満足度がこれ以上になれば帰宅する.
優先方向スタック (文字列)	優先方向が積み重なって入れてある入れ物
tmp 優先方向 (整数)	優先方向スタックに入っている優先方向を数値化したもの
Count(整数)	入場者が優先搭乗券の発券を受けてからのステップ数を数えるもの

表 B.3: 入場者エージェント変数 2

変数名	動作
優先搭乗券 (ブール)	入場者が優先搭乗券を所持しているかどうかを判断するもの
優先搭乗券アトラクション番号	入場者が優先搭乗券の発券を受けたアトラクションの ID 番号
Count(整数)	入場者が優先搭乗券の発券を受けてからのステップ数を数えるもの
優先搭乗券 (ブール)	入場者が優先搭乗券を所持しているかどうかを判断するもの
優先搭乗券アトラクション番号	入場者が優先搭乗券の発券を受けたアトラクションの ID 番号
stop(ブール)	パレードを観賞するときに立ち止まるかどうかを判断するもの
ride(整数)	船に乗っている状態かそうでない状態かを判断する数値
time(整数)	入場者が船に乗った時間を記録する

表 B.4: アトラクションエージェント変数

変数名	動作
ID	各エージェントの配列番号
X	空間の x 座標
Y	空間の y 座標
Layer	階層
Direction	各アトラクションが向く方向
待ち人数 (整数)	各アトラクションに並んでいる待ち行列の人数を数えたもの
待ち行列 (エージェント集合)	各アトラクションに並んでいる入場者の集団
現収容入場者 (エージェント集合)	各アトラクションに現在収容されている入場者の集団
現収容入場者数 (整数)	現収容入場者の人数を数えたもの
最大収容数 (整数)	各アトラクションが収容することができる最大人数
平均拘束時間 (整数)	各アトラクションに搭乗してから乗り終わるまでにかかる時間
優先搭乗券 (ブール)	各アトラクションに優先搭乗パスがあるかないか判断するもの.
優先搭乗券番号 (整数)	優先搭乗パス有りのアトラクションの ID 番号.
優先搭乗券所有者 (エージェント)	各アトラクションに収容される優先搭乗パス所有者のこと.

B.3 関数表

表 B.5: 組み込み関数 (Universe)

関数	動作
GetCountStep()	現在のステップ数を取得する
Openfile()	キストファイルを開く
Readfile()	テキストファイルから行を読み込む
CInt()	ブール型, 実数型, 文字列型, エージェント型の値を整数型に変換する
GetToken()	カンマ区切りの文字列から指定した位置の文字列を取得する
GetCountStep()	現在のステップ数を取得する
CDbl()	ブール型, 整数型, 文字列型, エージェント型の値を実数型に変換する
CountAgtSet()	エージェント集合型変数が保持しているエージェント数を取得する
MeasureDistance()	2点間の最短距離を計測する

表 B.6: 組み込み関数 (入場者エージェント)

関数	動作
GetCountStep()	現在のステップ数を取得する
TerminateAgt()	エージェントを完全削除する (エージェントはルール実行中であっても削除される)
AddAgt()	エージェント集合型変数にエージェントを追加する
CInt()	ブール型, 実数型, 文字列型, エージェント型の値を整数型に変換する
Cstr()	ブール型, 実数型, 整数型, 長整数型, 空間型, エージェント種別型, エージェント型の値を文字列型に変換する
Right()	文字列の右端から, 指定文字分の文字列を抽出する
Left()	文字列の左端から, 指定文字分の文字列を抽出する
Len()	文字列の文字数を求める
CountAgtSet()	エージェント集合型変数が保持しているエージェント数を取得する
CDbl()	ブール型, 整数型, 文字列型, エージェント型の値を実数型に変換する
Rnd()	0.0 以上 1.0 未満の一樣乱数値 (ランダムな値) を求める
Int()	実数値の少数部分を切り捨てて整数化する
Abs()	絶対値を求める

ユーザー定義関数（入場者エージェント）

- アトラクションへの移動 () : 入場者が各アトラクションエージェントへ向かうとき、それらへ向かう道があるかどうか判定する関数である。
- オブジェクトへ移動 () : 目的地に向かうことができるかどうか判断する関数である。図 3.4 のフローチャートのように優先方向に進めるかどうかなどを判定する。
- 目標アトラクションへ移動 () : 各入場者が目標アトラクションエージェントへ向かって移動することができ、同時にアトラクションエージェントへ到着した際にその待ち行列に並ぶようにする。
- 優先方向への移動 () : 優先方向に進むことができるかどうかを判定する関数である。移動可能チェック () 関数を使用しつつ、優先方向へ移動する為の道を作成したり、優先方向スタックに優先方向を積み重ねる。
- 直進性判定 () : 1 ステップ前に動いた方向に継続して進むことができるかどうかを判定する関数である。
- 回転判定 () : 優先方向から 90 度回転した場所に移動できるかどうかを判定する関数である。
- 回転判定 2 () : 優先方向から 180 度回転した場所に移動できるかどうかを判定する関数である。
- 移動可能チェック () : 今いる場所の周囲に動くことができるかどうかを判定する関数である。この関数は、回転判定や直進性判定に用いる。
- 移動可能チェック 2 () : 移動可能チェック () と同様に、今いる場所の周囲に動くことができるかどうかを判定する関数である。この関数は、優先方向へ移動する際に用いる。
- アトラクション効用値決定 () : アトラクション効用値を決定する関数である。
- 未訪問フラグセット () : 未訪問のアトラクションエージェントを設定する関数である。26 のアトラクションエージェント全てに対して未訪問で有り得ることを示す。
- 帰宅 () : 各入場者エージェントがゲートエージェントから退出させる関数である。
- アトラクションでの行動 () : 各入場者エージェントがアトラクションエージェントに搭乗するという関数である。各入場者の待ち時間+1, 各入場者の滞在時間+1 とすることで、搭乗したと設定する。

表 B.7: 組み込み関数 (アトラクションエージェント)

関数	動作
CountAgtSet()	エージェント集合型変数が保持しているエージェント数を取得する
RemoveAgt()	エージェント集合型変数から指定エージェントを削除する
AddAgt()	エージェント集合型変数にエージェントを追加する

ユーザー定義関数 (アトラクションエージェント)

アトラクション退場 () : 各入場者エージェントの滞在時間が各アトラクションエージェントの平均拘束時間以上になった場合にアトラクションエージェントを退出する関数である。その後、別のアトラクションエージェントへ移動するか帰宅するかの行動決定を行う。

アトラクション入場 () : 各入場者エージェントが目標アトラクションに到着した後アトラクションエージェントへ搭乗する関数である。アトラクションエージェントに1人以上並んでいる、かつ現収容入場者数が最大収容数より小さい時、待ち行列から入場者エージェントを1人ずつアトラクションエージェントへ収容する。