

最適制御策の効率的探索技術に関する研究

A Study of Efficient Technique for Finding Optimal Control Strategy

清武 寛 幸島 匡宏 松林 達史 戸田 浩之
Hiroshi Kiyotake Masahiro Kohjima Tatsushi Matsubayashi Hiroyuki Toda

NTT サービスエボリューション研究所
NTT Service Evolution Laboratories

Congestion events such as traffic jams and long queues at popular events must be treated as serious problems due to the strongly negative impacts of traffic paralysis and human stampedes. Therefore, this paper addresses the control strategies needed to smooth traffic flows and limit admission rates to reduce crowding. To find an effective strategy that can reduce congestion, Multi Agent Simulators are frequently used in the stages of evaluation and planning. However, control strategies are usually decided based on the experiences of experts and may not be truly optimal. In this paper, we propose a method that can find the optimal control strategy automatically.

1. はじめに

交通渋滞や、テーマパークにおけるアトラクションの長い待ち時間などの、いわゆる混雑は問題視されており、国や企業による対策が長年求められてきた。これに対し、道路やテーマパークの拡張等のハード面での対策や、信号の点灯時間変更、ユーザの誘導方法の変更等のソフト面での対策が考えられている。しかしながら、このような制御策によって対策を行うことにより、逆に混雑を引き起こしてしまうリスクや金銭的成本が発生するため、十分な対策は行えていない。実際の現場で制御を行うことが困難な場合に、疑似的に制御結果を取得する手段としてマルチエージェントシミュレータ (Multi Agent Simulator : MAS) [11] がある。MAS とは、人や車などの行動対象に対し移動速度や相互作用等のミクロのパラメータを定義し、交通渋滞や待ち行列などマクロな現象を理解するものである。MAS は、一般道やテーマパークを再現するものなどが研究・開発されている [7][8]。MAS を用いて発見された制御策が用いられた事例も報告されている [1]。

MAS による制御策探索を行う場合、考えうるすべての制御策を実施することが出来れば、最適な制御策を把握することが出来る。しかし、全探索には制御策の組み合わせ数が問題となる。例えば、テーマパークの混雑を緩和することを目指し、来訪者にアプリを用いた巡回ルート推薦という制御策を考えたとする。この場合、ユーザの増加に従い、巡回ルートの組み合わせ数は指数的に増えてしまう。そのため、すべての制御策を実行し、最適なものを選ぶという戦略は現実的に不可能である。

本稿では、テーマパークの混雑緩和を行う事を目的とし、ユーザ毎の巡回ルート変更という制御策の探索を MAS を用いて行なう。そこで、MAS に入力する制御策と出力される制御結果の関係を関数として扱い、数理最適化の手法を用いることを検討する。今回は、関数の最適化手法の一つであるベイズ最適化 (Bayesian Optimization : BO)[3] を利用する。本技術の全体像を図 1 に示す。制御策をベイズ最適化によって探索する場合、(I)MAS を実行し、(II) 得られた結果をもとに制御策と制御結果の関数形を予測し、(III) 真の関数値が低くなるような可能性が高い制御策を決定し、探索していくことで最適制御策を発見する。

この探索をより正しく行うためには、関数形の高精度な予

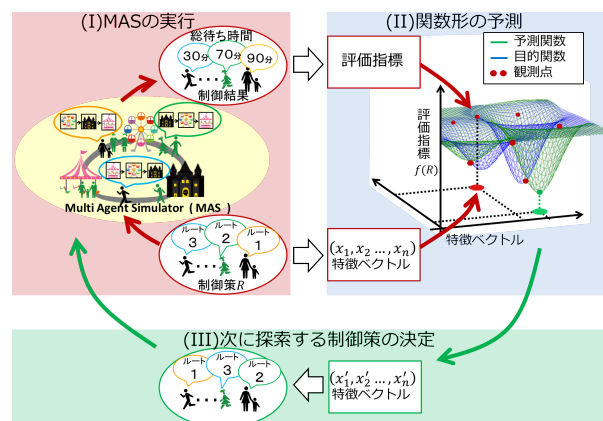


図 1: 提案手法のフレームワーク

測が必要である。この関数形を正しく予測するためには、入力となる制御策の特徴ベクトルへの変換方法が重要となる。ユーザ数の増加に従ってベクトルの次元が増えるような表現方法では関数の予測が正しく行われな。また、低次元に表現した場合、表現方法によっては特徴ベクトルに対応する制御策が多数存在してしまい、最適制御策の発見が難しくなる。

本研究では、ユーザのテーマパーク到着時刻という情報に着目し、制御策である巡回ルートの特徴ベクトルに表現する方法を提案する。具体的には、ユーザー一人一人を区別せず、同時時間帯に到着したユーザをまとめて考えることで特徴ベクトルに表現する。この表現方法により、ユーザの増加による特徴ベクトルの高次元化を回避することが出来る。また、時間帯毎に数えることで考えうる巡回ルートの数を減らすことが出来る。

評価実験では、混雑を測る評価指標を設定し、評価指標値が小さくなる巡回ルートの探索を行った。まず、本提案であるベクトルの次元数を下げたことの有効性を確認するため、ユーザ数を増やして問題の規模を大きくした場合の実験を行った。探索において、ユーザ毎の巡回ルートをランダムに決定するランダムサーチ、関数最適化の一手法である遺伝的アルゴリズムと比較を行い、本手法の有効性を確認した。実際の制御を必要とする現場では、平均待ち時間といった単一の評価指標ではなくユーザ毎の待ち時間の分散値等も考慮した巡回ルートが必要

である．そのため，複合的な評価指標に対しても実験を行い，本手法の有効性を確認するとともに，得られた巡回ルート of 定性評価を行った．

2. 関連研究

テーマパークの混雑緩和を行うために巡回ルートを推薦する方法として，最も空いているアトラクションに誘導する方法 [9] やユーザの興味に応じた推薦方法 [12] 等が考案されている．しかし，これらの推薦方法，“待ち時間を減らすための巡回ルート”や“アトラクション搭乗数を増やすための巡回ルート”のように，単一の評価指標を最適化する推薦方法であるため，待ち時間とアトラクション数という両方を考慮した評価指標への適応は難しい．

任意の評価指標に対して適応できる巡回ルートの推薦方法として，朝日ら [10] は，一部のユーザの巡回ルートを現状の最適なルートから変更し，MAS を実行することで評価指標が小さくなる巡回ルートを発見していくという方針を考案している．しかしながら，巡回ルートを少し変更する度に MAS を実行する必要があり大規模な場合の適応は難しい．

また，MAS の入出力関係を関数として扱い，最適化を行う場合，一つの手法として遺伝的アルゴリズム [6](Genetic Algorithm: GA) が考えられる．MAS に対しては，評価指標値の小さい巡回ルートの保存 (選択)，ユーザ間の巡回ルートの入れ替え (交叉)，確率的に別の巡回ルートに変更 (突然変異) を行い，次に探索する巡回ルートを決定することで適応が可能となる．しかしながら，MAS のように関数値を得るための時間的コストが大きい問題への適応は難しい．

関数値取得に時間がかかるような関数の最適化手法としてベイズ最適化 [3] が近年注目されている．ベイズ最適化は，機械学習や深層学習のハイパーパラメータチューニング，材料工学 [4] 等様々な分野で成功を収めている．しかしながら，その適応には入力ベクトルの次元が低次元であることを想定することが多く，今回のようにシミュレータのエージェント数に依存して入力ベクトルの次元が高く成り得る問題への単純な適応は難しい．入力が高次元に成り得る問題へのベイズ最適化の適応は，目的関数に加法性などの仮定を置き，高次元空間上で実施するものが提案されている [5][2]．MAS は個々のエージェントの振る舞いが複雑に作用し，目的関数に仮定を置くことは難しいため，提案手法では，入力を低次元のベクトルに変換し，低次元空間上でベイズ最適化を行う事を検討する．

3. 問題の定式化

本章では，今回解くべき問題の定式化を行ったあと，本稿で用いた MAS について述べる．

3.1 解くべき最適化問題

本稿では，複数の場所 (道路の交差点やテーマパークのアトラクション等) を経由する各ユーザに対して，その場所の巡回ルート R と移動開始時刻 T_n が与えられた際に，評価指標 (平均の待ち時間や移動時間等) $f(R)$ を最小にするような各ユーザの巡回ルートを求める問題を考える．つまり，

$$R^* = \arg \min_{R \in \{\text{巡回ルート全体}\}} f(R)$$

という最適化問題を考える．今回はテーマパークシミュレータを用いて実験を行った．

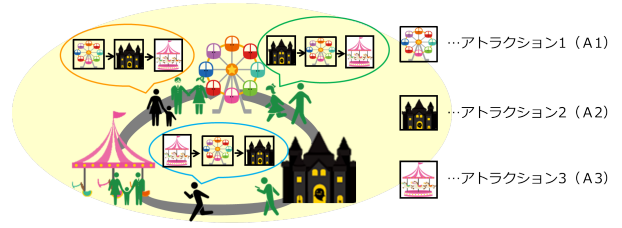


図 2: テーマパークシミュレータのイメージ図

ユーザ ID(n)	到着時刻 (T_n)	巡回ルート (r_n)
1	0	1 → 2 → 3
2	11	2 → 1 → 3
3	15	2 → 3 → 1
4	19	2 → 3 → 1
5	24	3 → 2 → 1

表 1: 入力制御策となる巡回ルートの例

3.2 テーマパークシミュレータの設定

本稿で用いるテーマパークシミュレータのイメージ図を図 2 に示す．シミュレータの実行時間の単位をステップと呼ぶことにする．アトラクション総数を K とし，来園ユーザ数を N とする．また，ユーザは全アトラクション K 個のうち体験アトラクション数 p 個を選択し，搭乗する．ユーザの到着間隔 t は平均 λ の指数分布に従うものとした．これは， λ ステップあたりに平均一人来るものとして考えることが出来る．今回のシミュレータ設定は，ユーザ数 $N = 50,500$ の 2 種類，アトラクション数 $K=3$ ，体験アトラクション数 $p=3$ とした．

シミュレータの入力 (制御策)：シミュレータの入力はユーザ全員分の巡回ルートとする．表 1 に，ユーザ数 $N=5$ ，アトラクション数 $K=3$ ，体験アトラクション数 $p=3$ の時の巡回ルートの例を示す．巡回ルートは数字と矢印によってあらわす．例えば， $A1 \rightarrow A2 \rightarrow A3$ の順にアトラクションに乗る場合の巡回ルートは“1 → 2 → 3”と表記する．

シミュレータの出力 (制御結果)：シミュレータの出力は各ユーザの待ち時間を要素とするベクトル W とする．

評価指標値：MAS の出力値を用いて，混雑を定量的に評価するような評価指標に変換する必要がある．評価指標としてはユーザの好みのアトラクションを推薦するもの等も考えられているが，今回はユーザの平均待ち時間 W_{ave} ，ユーザの待ち時間の標準偏差値 W_{std} を用いた以下の評価指標とする．

$$f(R) = W_{ave} + \alpha W_{std}$$

α は評価指標において平均待ち時間と待ち時間の標準偏差値のどちらを重視するか調整する定数である．

ユーザの行動ルール：ユーザはあらかじめ設定された到着時間に入園する．そして，入力された巡回ルートに従ってテーマパーク内を回遊するものとする．アトラクションに待ち列が出来ていた場合，最後尾に並び，先頭に来たら搭乗する．また，単純化のためアトラクション間の移動時間やユーザの移動速度などは考慮しない．

4. 提案手法

本章では，3.1 節で述べた最適化問題をベイズ最適化によって解く手法を提案する．MAS に対するベイズ最適化は入力と

	ナীব表現	低次元表現	提案表現
変換後の特徴ベクトル	ユーザIDルートID $\begin{matrix} 1 & \begin{pmatrix} 1 \\ 3 \\ 4 \\ 6 \end{pmatrix} \\ 2 & \\ 3 & \\ 4 & \\ 5 & \end{matrix}$	ルートi ユーザ数 $\begin{matrix} 1 \rightarrow 2 \rightarrow 3 & \begin{pmatrix} 1 \\ 0 \\ 1 \\ 2 \\ 0 \\ 1 \end{pmatrix} \\ 1 \rightarrow 3 \rightarrow 2 & \\ 2 \rightarrow 1 \rightarrow 3 & \\ 2 \rightarrow 3 \rightarrow 1 & \\ 3 \rightarrow 1 \rightarrow 2 & \\ 3 \rightarrow 2 \rightarrow 1 & \end{matrix}$	ルートi 到着時間毎のユーザ数 $\begin{matrix} 1 \rightarrow 2 \rightarrow 3 & \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \\ 1 \rightarrow 3 \rightarrow 2 & \\ 2 \rightarrow 1 \rightarrow 3 & \\ 2 \rightarrow 3 \rightarrow 1 & \\ 3 \rightarrow 1 \rightarrow 2 & \\ 3 \rightarrow 2 \rightarrow 1 & \end{matrix}$
次元数	N	KP_p	$KP_p \times J$
低次元	$\times$	$\bigcirc$	$\bigcirc$
可逆性	$\bigcirc$	$\times$	Δ

図 3: 表 1 の巡回ルートから特徴ベクトルへの変換

なる制御策と出力となる制御結果の評価指標値の関数形を近似し、その情報をもとに探索を行っていくアルゴリズムであるため、関数形を正しく近似することが必要となる。そのためには、入力となる制御策をどのようなベクトルに変換するかが重要となる。以下、本提案である各ユーザの巡回ルートの特徴ベクトルに変換する手法について述べ、その後、特徴ベクトルを用いたベイズ最適化について述べる。

4.1 入力ベクトルの表現方法 (提案手法)

ベイズ最適化を行うために、入力となる巡回ルートをベクトルに変換する方法について図 3 を用いて説明する。例えば、巡回ルートの一つの表現方法として、ルートに番号を付け、各ユーザがどのルートを選んだか表現する (ナীব表現) という方法がある。このナীব表現では、次元数がユーザ数と同数となり、一日 5 万人のユーザが来場する大規模なテーマパークでは 5 万次元のベクトルとなってしまふ。このように高次元で表現してしまうと、近似精度が著しく低下してしまうため、低次元で表現する必要がある。

しかし、低次元で表現した場合、対応する巡回ルートが多数存在することが問題となる。例えば、各巡回ルートを選んだユーザ数という形に表現した (低次元表現) とする。この表現ではベクトルの次元数が巡回ルートの総数となり、ユーザ数に依存しない形になる。しかし、ユーザの到着時刻や到着順序を考慮出来ないため、特徴ベクトルに対応する巡回ルートが多数存在してしまい、最適な巡回ルートの発見が難しくなる。つまり巡回ルートへの可逆性に乏しくなってしまう。

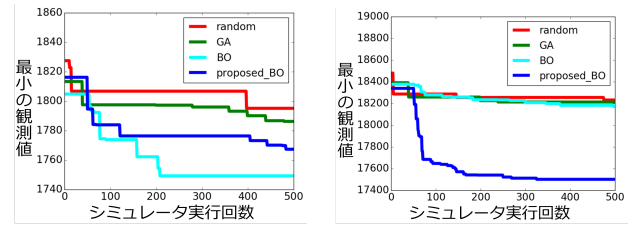
そこで、巡回ルートを以下のような特徴ベクトル $x = (x_{ij})$ に表現する方法を提案する。

$$x_{ij} = \# \{n \mid t_j \leq T_n < t_{j+1}, r_n = i\}$$

ここで、 $\#$ は集合の個数、 T_n はユーザ n の到着時間、 t_j はシミュレーション内時刻、 r_n はユーザ n の巡回ルートを表す。つまり全員分の巡回ルートを、ある時間幅の中で到着し、同一ルートを選んだユーザ数を要素とするベクトルに変換する。アトラクション数 K 個の中から p 個の体験アトラクションを選ぶ組み合わせの数は KP_p であるから、ユーザ数 N 、入力時刻の時間分割数を J とすると、 $(KP_p)^N$ 個の巡回ルートを $(KP_p) \times J$ 次元の特徴ベクトルに変換できることになる。作り方から、特徴ベクトルの次元数がユーザ数 N に依存しないことがわかるため、ユーザ数が増え、大規模になった場合にも対応できる。

4.2 MAS に対するベイズ最適化

MAS に対するベイズ最適化は、入力制御策となる巡回ルートと評価指標値の関係を表す目的関数がガウス過程に従うと



(a) N=50 人

(b) N=500 人

図 4: シミュレータの規模の違いにおける各手法の比較

仮定し、MAS によって観測点が得られる毎に事後分布を更新し、その情報を用いながら最適化を行う。

巡回ルートの特徴ベクトルへ変換: 初めにいくつかの制御策を MAS を用いて実行し、制御策 R と制御結果 W の組を取得する。そして、この制御策 R を提案手法によって特徴ベクトル x に変換する。以後、特徴ベクトル x と混雑の評価指標値 $y = f(R)$ の組を $D_L = \{(x_l, y_l) \mid l = 1, 2, \dots, L\}$ と書く。

目的関数の予測: 次に、目的関数の予測値と真の値からの予測値の分散値を推定する。ある入力制御策 R を変換した特徴ベクトル x における予測値 μ_x と分散値 σ_x は、 D_L を用いて $\mu_x = \mathbf{k}^T \mathbf{C}^{-1} \mathbf{y}$, $\sigma_x = c - \mathbf{k}^T \mathbf{C}^{-1} \mathbf{k}$ と表される。ここで、 $\mathbf{C}$ は入力変数の近さを定義するカーネル k のグラム行列であり、 $\mathbf{y} = (y_1, y_2, \dots, y_L)$, $\mathbf{k} = (k(x, x_1), \dots, k(x, x_L))$, $c = k(x, x)$ である。

獲得関数最大化: 次に、最小値をとる可能性を定量的に表す獲得関数 $acq(x)$ を用いて、次に探索する制御策に対する特徴ベクトルを決定する。獲得関数は、評価指標の予測値 μ_x が低く、予測誤差 σ_x の大きい入力に対して大きな値を取るよう設計される。特徴ベクトル x が存在する特徴ベクトル空間 X の中で、獲得関数値が最も大きくなる特徴ベクトル $x_{\text{next}} = \arg \max_{x \in X} acq(x)$ を決定する。

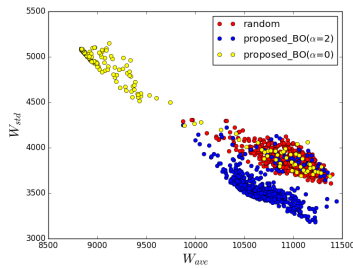
特徴ベクトルを巡回ルートへ変換: x_{next} は実数値ベクトルであるが、巡回ルートに対応する特徴ベクトルは整数値を要素に持たなければならない。そのために、整数値のベクトル x^{int} であり、 x_{next} に最も近い、

$$x_{\text{next}}^{\text{int}} = \arg \min_{x^{\text{int}}} |x_{\text{next}} - x^{\text{int}}|^2$$

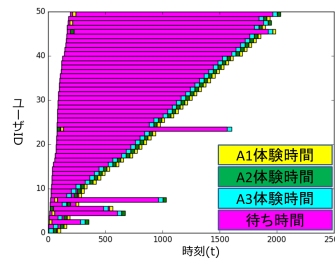
を決定する。そして、 $x_{\text{next}}^{\text{int}}$ に対応する制御策 R を次の入力とする。この $x_{\text{next}}^{\text{int}}$ に対応する入力制御策は複数存在するが、今回はその中からランダムに一つ選び、次のシミュレーションに対する制御策 R_{next} とした。そして、新たに得られた観測点 $(x_{\text{next}}^{\text{int}}, f(R_{\text{next}}))$ を D_L に追加し、再度関数の予測を行うことで探索を行う。

5. 評価実験

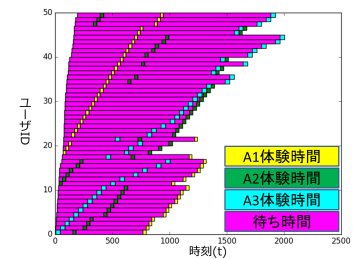
本稿では、本提案手法の有効性の確認のために行った二つの実験について述べる。まず一つ目は、提案表現の特徴ベクトルの有効性を確認するため、他の手法と比べユーザ数を増やした場合に評価指標値の小さい入力値が探索できるかという実験を行った。評価指標は $\alpha = 0$ のものとした。二つ目は、実際のテーマパークでの活用を想定し、単一ではなく、複合的な評価指標値に関して同様の実験を行った。具体的には、3.2 節で述べた評価指標値の α の値を変更した。本手法の有効性を確認し、各指標によって得られた巡回ルートの定性評価を行った。



(a) 待ち時間の平均と標準偏差値の散布図



(b) 発見された最適解 ($\alpha = 0$)



(c) 発見された最適解 ($\alpha = 2$)

図 5: それぞれの評価指標値において発見された巡回ルート

5.1 比較手法

比較手法には、ユーザが巡回ルートをランダムに選ぶランダムサーチ、関数最適化の一手法である遺伝的アルゴリズム (GA)、入力ベクトルをナイーブ表現としたナイーブ BO の 3 種類を用いた。ベイズ最適化のカーネルには、提案手法、ナイーブ BO 共にガウスカーネルを用いた。獲得関数には期待値改善 (EI) を用いた。また、GA の遺伝的オペレータにはエリート選択・ルーレット選択、2 点交叉、突然変異を用いた。各オペレータのパラメータはグリッドサーチで決定し、最も性能の良かったものを採用している。

5.2 実験結果

一つ目の実験の結果を図 4 に示す。図 4 では、横軸に探索回数、縦軸にそれまでの探索で発見された巡回ルートの最小評価指標値をプロットしている。50 人、500 人どちらのシミュレーション結果を見ても、ランダムサーチや GA に比べ、待ち時間の短い巡回ルートを発見することが出来ている。また 50 人の場合は、各手法で大きな差は見られないが、500 人の場合は提案手法によって待ち時間の短い巡回ルートの発見に成功している。尚、GA においても、十分なシミュレーション回数 (160000 回程度) を行うことにより提案手法と同程度の解を見つけることが出来ている。これは、ナイーブ表現ではユーザ数に依存して入力が高次元なる一方、提案表現では低次元で表現でき、関数近似の精度を落とさず探索出来たためと考えられる。このことから問題の規模が大きくなった場合にも有効な手法であることが示された。

次に、二つ目の実験の結果を図 5 に示す。評価指標は $\alpha = 0, 2$ の場合を用いた。図 5a には、各評価指標において発見された巡回ルートにおける平均待ち時間 W_{ave} と待ち時間の標準偏差値 W_{std} の散布図を示す。一つの丸が一つの巡回ルートに対応している。図 5b, 図 5c には各評価指標において発見された最適巡回ルートで MAS を行った場合の各ユーザの状態を示す。図 5a をみると、 $\alpha = 0$ の場合に提案手法によって発見された巡回ルートは、ランダムサーチによって得られた巡回ルートに比べ、待ち時間は短くなるものの個々の待ち時間の分散値は大きくなっている。これに対し、個々人の待ち時間の分散値も重視する $\alpha = 2$ の場合は、平均待ち時間は少し長くなるものの、ユーザ毎の待ち時間のバラつきが少なくなるような巡回ルートを探すことに成功している。実際に、図 5b, 図 5c からわかるように、 $\alpha = 0$ のときに発見された巡回ルートに比べ、 $\alpha = 2$ の場合のほうがユーザ毎の待ち時間が平準化されている様子がわかる。また、 $\alpha = 0$ のときはほぼすべてのユーザが同一の巡回ルートを選択しているが、 $\alpha = 2$ の場合は、20 番目以前と以後のユーザで選択する巡回ルートを変えており、結果として待ち時間のバラつきを小さくする事に成功している。

6. まとめ

本稿では、テーマパークシミュレータを用いて混雑緩和に効果的な巡回ルートの探索を行うため、ベイズ最適化を用いることを検討した。ナイーブなベイズ最適化では高次元の入力に対応できないという課題に対し、入力を低次元のベクトルで表現する手法を提案した。評価実験では、ランダムサーチや遺伝的アルゴリズムなどの最適化手法に比べ、ユーザ数が増えても待ち時間の短い巡回ルートを効率的に発見出来ることを確認した。また、実際の現場で想定しうる複合的な評価指標に対しても有効な手法であることを確認した。本技術により、テーマパークだけでなく、オリンピックなどで発生しうる、試合会場から駅に向かう混雑などに対しても効果的な人流誘導案を生成することが出来るようになることが期待される。

参考文献

- [1] 中国・貴陽市において、ビッグデータを活用した「渋滞予測・信号制御シミュレーション」の実証実験で渋滞緩和効果を確認。 <http://www.nttdata.com/jp/ja/news/release/2016/053101.html>.
- [2] Kirthevasan Kandasamy, Jeff Schneider, and Barnabas Poczos. High dimensional bayesian optimisation and bandits via additive models. In David Blei and Francis Bach, editors, *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pp. 295–304. JMLR Workshop and Conference Proceedings, 2015.
- [3] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical bayesian optimization of machine learning algorithms. In *Advances in neural information processing systems*, pp. 2951–2959, 2012.
- [4] Tsuyoshi Ueno, Trevor David Rhone, Zhufeng Hou, Teruyasu Mizoguchi, and Koji Tsuda. Combo: An efficient bayesian optimization library for materials science. *Materials Discovery*, Vol. 4, pp. 18–21, 2016.
- [5] Ziyu Wang, Frank Hutter, Masrour Zoghi, David Matheson, and Nando de Freitas. Bayesian optimization in a billion dimensions via random embeddings. *Journal of Artificial Intelligence Research*, Vol. 55, pp. 361–387, 2016.
- [6] 伊藤育志. 遺伝的アルゴリズムの基礎 GA の謎を解く. オーム社, 東京, 1994.
- [7] 株式会社 NTT データ数理システム. S4 simulation system 新機能紹介～エージェントシミュレーション～. https://www.msi.co.jp/userconf/2013/pdf/muc13_CR6_2.pdf.
- [8] 兼田敏之, 構造計画研究所, 名古屋工業大学. Artisoc で始める歩行者エージェントシミュレーション: 原理・方法論から安全・賑わい空間のデザイン・マネジメントまで. 構造計画研究所, 2010.
- [9] 清水仁, 松林達史, 納谷太. 混雑飽和状態の遊園地における待ち時間削減手法のシミュレーション評価. SIG-DOCMAS 研究会.
- [10] 朝日大地, 井上知洋, 筒井章博. 誘導スケジューリングによる集団状態最適化手法のシミュレーション評価. 研究報告知能システム (ICS), Vol. 2015, No. 6, pp. 1–8, 2015.
- [11] 島海不二夫, 山本仁志ほか. マルチエージェントシミュレーション: 1. マルチエージェントシミュレーションの基本設計. 情報処理, Vol. 55, No. 6, pp. 530–538, 2014.
- [12] 柳田靖, 鈴木恵二. テマパークにおける巡回戦略の比較. 情報処理学会研究報告知能と複雑系 (ICS), Vol. 2007, No. 26, pp. 15–22, 2007.