

# 汎用シミュレーションシステムの紹介

嶋田 佳明

OR手法の一つでもあるシミュレーションは、サプライチェーン、製造工程、ヘルスケア、コールセンター、交通、通信、社会システムなど解析的には分析するのが難しい複雑なシステムを分析する手法として、古くから用いられてきた。(株)NTTデータ数理システムでは、これらのモデル構築や分析を支援するソフトウェアとして、汎用シミュレーションシステム S<sup>4</sup> Simulation System (S<sup>4</sup>, エスクワトロ)を開発・販売している。本稿では、そのシステム概要や関連技術について紹介する。

キーワード：離散イベントシミュレーション、連続シミュレーション、エージェントシミュレーション、ソーシャルフォースモデル、シミュレーション最適化、強化学習

## 1. はじめに

シミュレーションは現実のシステムをモデル化（模擬）し、モデルを実行することで、現実世界における挙動を分析・予測する問題解決手法である。ここで言う、システムとはシミュレーションの対象のことである。また、モデルとは複雑な現実のシステムからシミュレーションの目的によって、特徴だけを抽出し、簡略化したものである。モデル化されたシステムは、コンピュータ上に実装され、実行される。シミュレーションを使用するメリットは、実際に試すには、コストや時間がかかるシステムや、解析的には表現できないシステムを分析できることにある。また、シミュレーションによって、条件の違いによる振る舞いを分析したり、最適な条件を探索したり（シミュレーション最適化）できる。

一般的には、コンピュータ上にモデルを実装する場合、プログラミングが必要となる。ただし、シミュレータを使用すれば、プログラミングの負荷を減らせ、ユーザはモデル構築のみに集中することができる。(株)NTTデータ数理システムでは、汎用シミュレーションシステムとして、S<sup>4</sup> Simulation System (S<sup>4</sup>, エスクワトロ)を開発・販売している。S<sup>4</sup>はPythonで開発されており、GUIで手軽にモデル構築ができる一方で、柔軟なカスタマイズ性能や、汎用性を兼ね備えているのが特徴である。また、シミュレーションに必要な不可欠なスケジュール管理は、プロセス指向アプローチで実装されている。

本稿の構成を次にまとめる。第2節では、S<sup>4</sup>が取り扱うことができるシミュレーションの種類について解説する。第3節、第4節では、S<sup>4</sup>のシステム概要とシステムの根幹となっているシミュレーション記述言語について紹介する。第5節では、シミュレーションの種類に基づいたS<sup>4</sup>のモデル構築機能について紹介し、その中から特に歩行者モデルのモデリング機能について言及する。第6節、第7節ではシステムを最適化する為の、シミュレーション最適化機能や強化学習機能を紹介する。

## 2. シミュレーションの種類

シミュレーションには、フライトシミュレーションや地球シミュレーション、耐震シミュレーションといったものから、半導体や物理・化学シミュレーションなどさまざまがある。その中でも、S<sup>4</sup>が対象としているシミュレーションは、離散イベントシミュレーション（離散系シミュレーション、離散型シミュレーション、離散事象シミュレーションなどとも言われる）、連続型シミュレーション（システムダイナミクス）、エージェントシミュレーションでモデル化されるものである。

離散イベントシミュレーションとは、システムの状態が変化するイベントが離散的に起こるようなシミュレーションで、事象の発生を乱数によって表現することで、待ち行列に代表されるシステムを分析するモデル化手法である。応用例として、銀行のATMにおける順番待ちや、コールセンターにおけるオペレータ待ち、製造工程における仕掛り在庫、ネットワーク分析などが挙げられる。また、待ち行列の分析以外にも、電気製品の故障解析や、製品のライフサイクル分析など、幅広く応用が可能なシミュレーションである。

連続型シミュレーションは、別名システムダイナミ

しまだ よしあき  
(株)NTTデータ数理システム  
シミュレーション&マイニング部  
〒160-001 東京都新宿区信濃町 35 番地 信濃町煉瓦館 1 階

クスとも呼ばれるシミュレーションで、複数の相互に関係しあう変数の連続的な時系列変化を分析する手法である。システムの内部構造をモデル化して、その挙動をシミュレートすることで、その動的特性を解明することができる。システムの構成要素となる状態量の関連性を常微分方程式の形でモデル化する。応用例としては、エネルギー資源と発電量の関係や、生態系の分析、社会システム分析などがある。

エージェントシミュレーションは、一定のルールに従って、自律的に行動するエージェントの振る舞いをモデル化し、エージェント同士の相互作用から現れる、複雑な社会現象を分析する。エージェントシミュレーションでは、エージェント間のつながりや、エージェントの状態や行動に対して、ある程度複雑な条件を課すことができるため、連続型シミュレーションよりも現実世界に近いモデル化を行うことができる。また、エージェント同士が互いに影響した結果、マクロ的な性質が変化する創発と呼ばれる現象を観察できるのも特徴である。応用例としては、交通シミュレーションや、人流シミュレーション、避難シミュレーションなどがある。

### 3. システム概要

S<sup>4</sup>は2010年に離散系シミュレータとして、Ver1.0がリリースされ、その後のバージョンアップを経て、現バージョン Ver5.2 では、連続型シミュレーションやエージェントシミュレーションが取り扱えるようになっている。

(株) NTT データ数理システムでは、数理計画法パッケージ Numerical Optimizer (旧 NUOPT) を古くから開発しているが、数理計画法では表現できない複雑なシステムの最適化や不確定要素がある最適化の領域をカバーするために、シミュレーションシステムを開発したのが原点にある。

次にシステム構成であるが、S<sup>4</sup>は大きくモデル構築やシミュレーション結果の分析を行う GUI 部分と、シミュレーション記述言語 psim 言語部分に分かれている(図1)。psim 言語については、4 節で説明する。GUI で作成されたモデルは、実行時に psim 言語 (Python) のプログラムに変換され実行される。変換されたプログラムは、Python が実行される環境であれば動作する為 (実行には S<sup>4</sup> のライセンスが必要になるが)、システムに組み込むことや、独自の UI からシミュレーションを実行することも容易である。また、GUI を用いず、Python プログラムとして、psim 言語だけで実装して

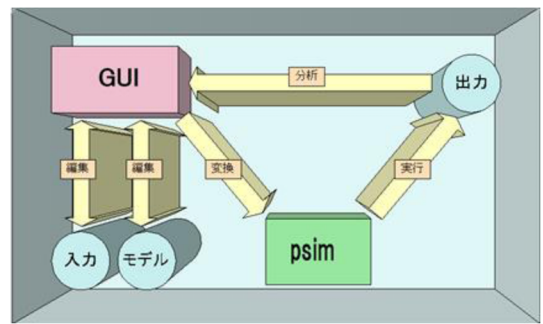


図1 システム構成

いくことも可能である。GUI も Python (wxPython) で開発されている為、見た目もとても洗練されている。

### 4. psim 言語

psim 言語は Python 上で動作するプロセス試行のシミュレーション記述言語で (株) NTT データ数理システムが独自に開発した。プロセス指向のイベント処理エンジンであるため、シミュレーションで発生する大量のイベントを効率的にスケジューリングすることができるのが特徴である。プロセス指向とは、プロセスを基盤とするイベント処理方法のことで、スケジューラは、プロセスの継続を管理し、ある時間の待ち受けがなければ、次の待ち受けまで時間を進める。よって、イベント発生が疎の場合には、高速にシミュレーションが進む。

プロセス指向アプローチの実装には、Coroutine を使って実装することが多い。Coroutine とはプロセスの実行中にプロセスをいったん中断し、プロセスを再開する仕組みである。psim 言語では Python の generator 機能を使って Coroutine を実装している。

また、psim 言語は、これまでのシミュレーション記述言語になかった、複雑な待ち受けや状態推移を、柔軟かつ簡潔に記述することができるように設計されている [1]。その他にも乱数発生やデータの分布推定機能、その他数多くの Python による API をもつ。

### 5. モデル構築機能

S<sup>4</sup>は GUI からモデル構築を行うことができるが、シミュレーションの種類によって、モデル構築の方法は異なる。ここからは、シミュレーションの種類別に S<sup>4</sup> のモデル構築機能を紹介していく。

#### 5.1 離散イベントシミュレーションモデル

S<sup>4</sup>で離散イベントシミュレーションモデルを構築する場合、S<sup>4</sup>があらかじめ提供している、「アイテム」、

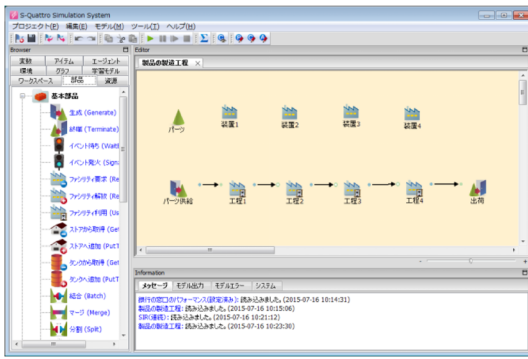


図2 離散イベントシミュレーションモデル構築画面



図4 連続部品編集画面

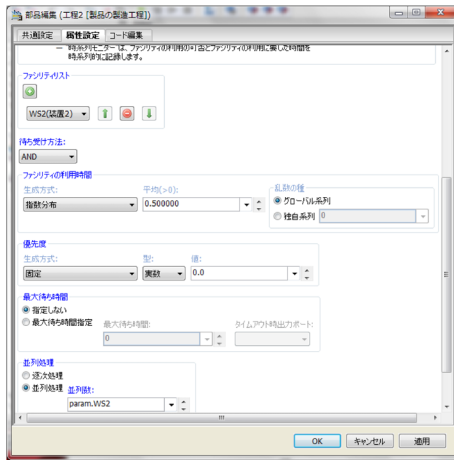


図3 部品編集画面

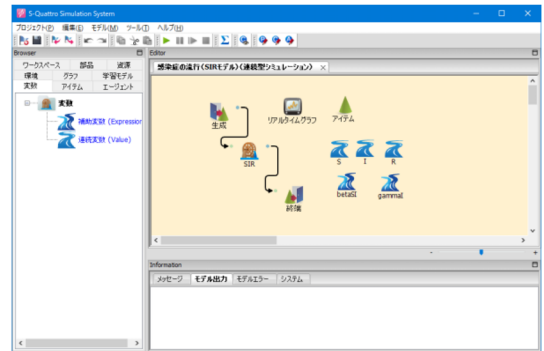


図5 連続型シミュレーションモデル構築画面

「資源」, 「遅延」, 「ストア」などのシミュレーションの構成要素となる部品を配置し, それらをフローで連結することでモデルを表現する (図2) [2]. シミュレーションのフロー上は, 「アイテム」が流れ, 「アイテム」が各部品に離散的に移動していく. アイテムが各部品に到着することが, 各部品の処理開始の契機となる.

$S^4$  は汎用シミュレータのため, このような部品の意味は作成するモデルによって変わる. 銀行の窓口のモデルであれば, 「アイテム」は「お客」, 「資源」は「窓口」, コールセンターのモデルではれば, 「アイテム」は「問い合わせ」, 「資源」は「オペレータ」や「電話回線」, 製造工程であれば, 「製品」がアイテム, 「資源」は「機械」や「作業員」の意味で使われる. 配置した部品をダブルクリックすると, パラメータの編集画面 (図3) が開き, 部品の挙動をコントロールするパラメータを設定することができる. たとえば「資源」には, 使用する資源の数や, 資源の専有時間などを設定することができ, それらを確率分布から生成するように設定す

ることできる.

各部品のプログラムはユーザが自由に編集することができ, オリジナルの部品の挙動を Python でカスタマイズすることができるようになっている.

## 5.2 連続型シミュレーションモデル

連続型シミュレーションのモデルでは, 状態量を示す変数間の関連性を常微分方程式でモデル化する.  $S^4$  には状態量を表す「連続変数」部品と常微分方程式を設定する「連続」部品がある. 連続部品には常微分方程式を設定する (図4).

離散イベントシミュレーションモデルと同様, 「連続」部品の処理開始の契機となるのは「アイテム」の到着となるように設計されている (図5). これによって, 離散イベントシミュレーションと連続型シミュレーションのハイブリッドシミュレーションも可能となっている. たとえば, 石油タンカーの輸送シミュレーションをハイブリッドシミュレーションモデルで表現すると,

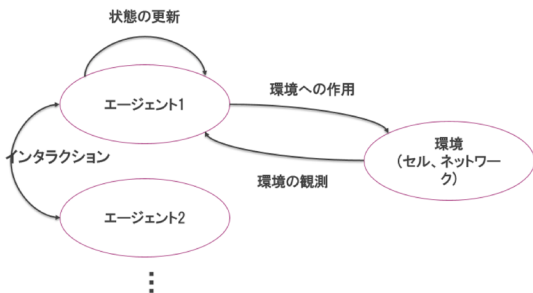


図 6 エージェントシミュレーションの設計

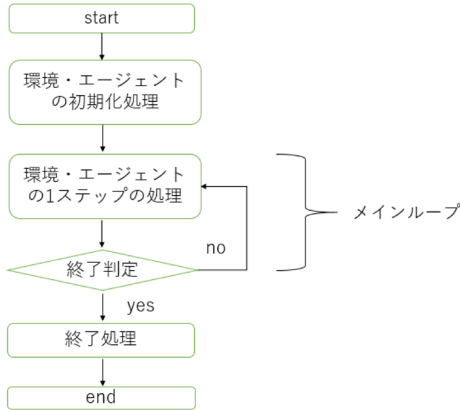


図 7 エージェントシミュレーションのフローチャート

タンカーの到着は分散イベントシミュレーションモデルで表現しておき、タンカーが石油タンクに石油を移す所は、連続部品で表現する。また、気体や液体その他、連続的に量が変化するような製品を製造しているラインのシミュレーションもハイブリッドシミュレーションモデルで表現することもできる。

### 5.3 エージェントシミュレーションモデル

エージェントシミュレーションとは、エージェントの行動をモデル化し、エージェント同士の相互作用によって起こる複雑な現象をシミュレーションするものであった。エージェントシミュレーションモデルの構成要素には、エージェントとエージェントが行動する場である環境がある。

エージェントは内部状態をもっており、シミュレーションが進むにつれ、環境や他のエージェントとのインタラクションによって、自身の内部状態や環境の状態を変化させていく (図 6)。

エージェントシミュレーションの計算は、一般的に初期化処理、メインループ、終了処理からなる。初期化処理では、環境とエージェントを生成 (初期化) し、エージェントを環境上に配置する。メインループでは、エージェントと環境の 1 ステップの動作 (ステップ処

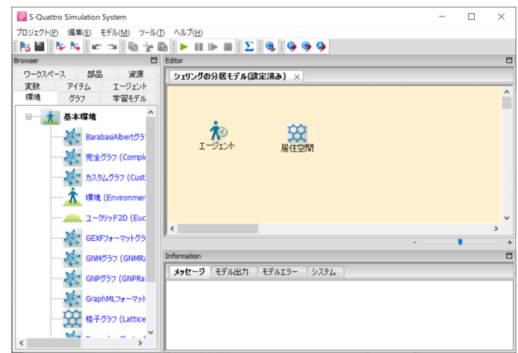


図 8 エージェントシミュレーションモデル構築画面

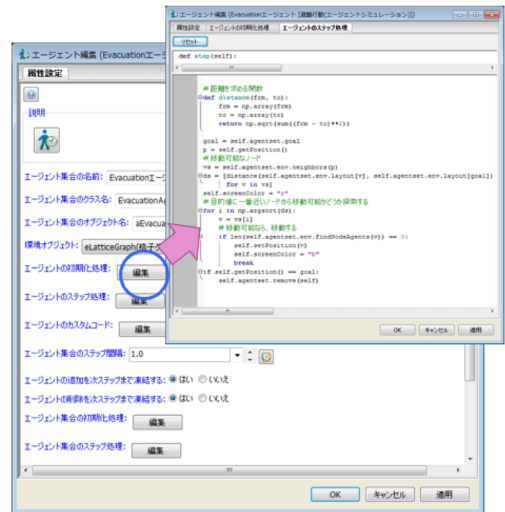


図 9 エージェントシミュレーションの編集画面

理) を繰り返す。終了処理では、結果をファイルに出力するなどの処理を行う (図 7)。シミュレータがなければ、これら一連の処理の流れをすべて一からプログラミングする必要があるが、S<sup>4</sup> が処理の流れやスケジュール管理を担い、さらにエージェントや環境などのフレームワークを用意しているため、ユーザは初期化処理やステップ処理の中身を実装するだけでシミュレーションができる。

S<sup>4</sup> でエージェントモデルを構築するには、まず「エージェント」部品と「環境」部品を配置する (図 8)。エージェント部品や環境部品を開くと、エージェントの行動ルールや相互作用をプログラムしていくための編集画面が開く。編集画面の編集ボタンをクリックするとエディタが開き、ユーザはそのエディタ上でプログラミングしていく (図 9)。プログラミング言語は Python である。

また、環境部品にはセル空間や、連続空間、ネット



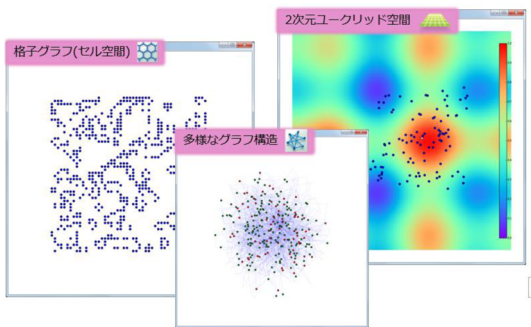


図 10 エージェントシミュレーションの環境例



図 12 歩行者シミュレーション用 3D アニメーションの例

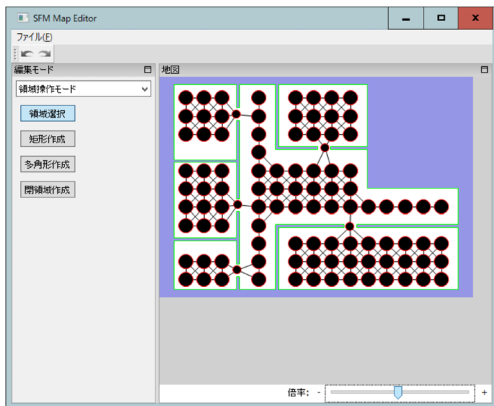


図 11 エージェントシミュレーションの地図エディタ

ワーク空間があり、シミュレーションの状況はアニメーションで可視化することができる (図 10)。

## 5.4 歩行者モデル

エージェントシミュレーションで表現されるモデルには、避難行動や施設内の回遊行動などの歩行者モデルがある。S<sup>4</sup> はこれら歩行者モデルに関する機能が充実しているため、ここでは歩行者モデルに着目して紹介する。

### 5.4.1 ソーシャルフォースモデル

歩行者の動きをモデル化する方法はさまざまあるが、S<sup>4</sup> ではソーシャルフォースモデル (Social Force Model) を採用している。ソーシャルフォースモデルとは、群衆行動の力学ベースモデルの一つで、各歩行者は質量をもつ質点として表され、平面内で運動する粒子と考える。各歩行者は、目的地をもち、他の歩行者や障害物から相互に干渉を受けながら、それぞれが運動する。

ソーシャルフォースモデルの特徴は、歩行者同士の衝突を回避する動きが自然な形で表現されることにある。たとえば、交差点における歩行者同士のすれ違いなどは、他のモデルに比べ、よりリアルに表現できる。

### 5.4.2 S<sup>4</sup> による歩行者シミュレーション

S<sup>4</sup> で歩行者シミュレーションを行うには、歩行者が行動する為の環境を構築する必要がある。S<sup>4</sup> には歩行者モデルの環境構築を支援するための、地図エディタが備わっている。ユーザは、矩形や多角形を組み合わせ、障害物や歩行者が移動可能な領域のレイアウトを作成していく。地図エディタ (図 11) は、CAD ソフトとの連携も可能で、作成した図面を、DXF ファイル形式で読み込み、Line, PolyLine, Circle, Arc エンティティを認識することができる。その他、GeoJSON 形式のデータをインポート、エクスポートできるため、オープンソースの地図情報システムである QGIS との連携も可能となっている。

## 5.5 アニメーション

エージェントシミュレーションの結果を直感的に分析するため、アニメーションによる描画は大変重要である。S<sup>4</sup> には 2D、3D 両方のアニメーション描画機能が備わっており、ユーザは特別な設定をすることなく S<sup>4</sup> 上で即座にアニメーションを描画することができる。

2D アニメーションは、セル空間、ネットワーク空間、連続空間上のエージェントの状態を色や形で表現する他、エージェントの位置関係も描画する。エージェント同士のつながりを示すようなネットワークグラフなどの抽象的な空間から、地図空間まで汎用的に用いられる。

3D アニメーションは (図 12)、歩行者シミュレーション用に特化している。特定の歩行者に注目して画面が移動する「注目カメラ」や、エージェント視点になる「視点カメラ」といった、画面の表示方法の切り替えや、壁や床の素材の選択、壁の高さの指定などの操作が可能である。

## 6. シミュレーション最適化

はじめに述べたように、シミュレーションの目的の一つに、条件を変えた時のシステムの振る舞いを分析し、その結果、システムを最適化することがある。これをシミュレーション最適化と呼ぶ。シミュレーション最適化の長所は、解析的には解けなかったり、そもそも数式では表せないような複雑システムの最適化や、不確定事象を含む最適化を行うことができる所である。シミュレーション最適化では、シミュレーション結果が目的関数になり、シミュレーションのコントロールパラメータが、最適化における変数になる。注意すべき点は、シミュレーション結果は、確率的に求まるため、単純に単一のベストなシミュレーション結果を出したパラメータが最適解とは限らないという点である。そのため、あるパラメータにおけるシミュレーション結果に対して、統計処理を行い、シミュレーション結果の推定値を求め、それを目的関数として、最適化問題を解く必要がある。

シミュレーション結果の推定値には、期待値が用いられることが多い。シミュレーションのコントロールパラメータを  $\theta$ 、シミュレーション結果を  $Y(\theta)$  とすると、推定値は  $E[Y(\theta)]$  となる。ここで、 $E$  は複数のシミュレーション結果の期待値を取ることを表す。

実際に最適化を行う手法であるが、単純にはすべてのパラメータにおいて、もっとも目的関数が最小（最大）になるパラメータを探索すればよいが、1回のシミュレーションに時間がかかる場合や、パラメータの数が多い場合には、計算コストがかかる。そのため、最適化アルゴリズムを用い、効率よく探索を行う必要が出てくる。 $S^4$  では最適化アルゴリズムとして、粒子群最適化法 PSO (Particle Swarm Optimization) [3] と、目的関数の解析的表現が困難な場合や目的関数の微分に関する情報を用いることができない状況下で最適解を求める DFO (Derivative Free Optimization) [4] を採用している。尚、Numerical Optimizer のアドオンモジュールに DFO モジュールがあるが、そのモジュールが  $S^4$  に組み込まれている。

## 7. シミュレーションによる強化学習

たとえば、製造ラインにおいて、リードタイムが最短となるように、ジョブを機械に割り当てることを考える。この場合、数理モデルでモデル化する方法もあるが、製造ラインが複雑になるにつれ、モデル化する

のが徐々に難しくなってくる。そのような場合は、さまざまな機械への割り当ルールをシミュレーションで試すことで最適化を図ることが考えられる。また、別の方法として、シミュレーションによる強化学習によって、製造ラインを最適化する方法もある。

強化学習とは、意思決定者が、逐次的に環境の観測と意思決定（行動）を繰り返しながら、即時報酬と呼ばれる観測ごと与えられる報酬値の割引和を最大化するような行動ルールを推定する機械学習手法である。シミュレーションによる強化学習の場合は、環境がシミュレーションモデルそのものになり、観測値と即時報酬値はシミュレーションモデルの状態から求まる値となる。行動をとった場合の環境への影響は強化学習によって提案された行動をもとに行動した結果である。 $S^4$  には、行動を行いながら学習を行うオンライン型の手法が搭載されている [5]。

## 8. まとめ

$S^4$  は離散イベントシミュレーション、連続型シミュレーション、エージェントシミュレーションが行える、汎用シミュレーションシステムである。GUIによるモデル構築や、シミュレーション記述言語の psim 言語が備わっており、歩行者シミュレーション機能や最適化機能が充実している。本稿を読まれて、興味をもたれた方は、是非一度試してほしい。また、ユーザ事例や学生研究事例も Web ページで多く公開している。こちらも併せて参照されたい [6]。

### 参考文献

- [1] 山本晃成, 雪島正敏, 高橋勲男, 山下浩, “ $S^3$  Simulation System の開発 2 シミュレーション記述言語 psim,” 日本オペレーションズ・リサーチ学会 2010 年春季研究発表会, pp. 140–141, 2010.
- [2] 雪島正敏, 山本晃成, 高橋勲男, 山下浩, “ $S^3$  Simulation System の開発 1,” 日本オペレーションズ・リサーチ学会 2010 年春季研究発表会, pp. 138–139, 2010.
- [3] 雪島正敏, 山本晃成, 高橋勲男, 山下浩, “ $S^3$  Simulation System の開発 3,” シミュレーション最適化 日本オペレーションズ・リサーチ学会 2011 年春季研究発表会, pp. 138–139, 2011.
- [4] 島田直樹, 田辺隆人, 山下浩, “数式表現によらない関数の制約付き最適化,” 日本オペレーションズ・リサーチ学会 2009 年春季研究発表会, pp. 230–231, 2009.
- [5] 成瀬俊輔, 雪島正敏, “ $S^4$  Simulation System の開発 5 意思決定を含むモデルの強化学習による最適化,” 日本オペレーションズ・リサーチ学会 2015 年春季研究発表会, pp. 136–137, 2015.
- [6] (株) NTT データ数理システム,  $S^4$  Simulation System ユーザ事例, <http://www.msi.co.jp/s4/solution/index.html> (2019 年 10 月閲覧)