

# 深層学習を用いた作業員行動のモデル化による物流倉庫内 シミュレーション構築

鈴木 優真<sup>1,a)</sup> 高木 魁利<sup>1</sup> 浅井 悠佑<sup>1</sup> 永田 吉輝<sup>1</sup> 片山 晋<sup>1</sup> 浦野 健太<sup>1</sup> 米澤 拓郎<sup>1</sup>  
河口 信夫<sup>1,2</sup>

**概要：**E コマースの拡大等の要因により，物流の需要が拡大している．それに伴い，商品が中継される倉庫内の業務が複雑化しており，作業の効率化が求められる．適切な施策を検討するための有効なアプローチとして，デジタルツイン技術の活用が注目されている．現場を再現したシミュレーション上での様々な施策の試行により，休みなく運用されている現場に介入しない検証が可能となる．実用可能なシミュレータを構築するには，現場の使われ方や作業員の動きの特徴を正確に捉える必要がある．本研究では，現場の実データ（作業員や物の位置情報）をもとに，深層学習を用いて作業の特徴を抽出し，そのモデルを用いてシミュレータを駆動させる手法を検討した．本研究では，実データの整備に先立ち，ルールベースで構築したシミュレータでデータを生成して実験を行った．このシミュレーションデータを用いて深層学習モデルを訓練し，作業員の行動や物品フローの特徴をモデルに学習させた．次に，訓練されたモデルの出力を用いてシミュレータを駆動させ，明示的にルールを与えずにシミュレータを動作可能な点を確認した．この一連のプロセスを通じて，実データに基づくシミュレータの構築の実現可能性を示した．

**キーワード：**深層学習 シミュレータ データ駆動モデリング

## 1. はじめに

物流の需要が拡大している現在，商品が中継される物流倉庫内の業務は複雑化しており，作業の効率化が求められている [1]．しかし，物流倉庫は絶えず運用されているため，現場での適切な効率化施策の検証は困難である．この課題の解決に有効なアプローチとして，デジタルツイン技術の活用が注目されている．デジタルツインとは，物理的システムの仮想表現をデジタル空間上に構築し，現実と仮想空間を情報で結びつけ，現実の挙動を仮想空間上で試行・予測する技術である [2]．物流倉庫においても，現場を再現したシミュレータ上での様々な施策の試行により，現場に介入せずに，かつ低コストで高速に施策の比較が可能となる．

現状多くの商用シミュレータにおいては，システム全体の構造をモデル化した上で，各構成要素の挙動を明示的なルールとして定義する手法が採用されている [3], [4]．こ

うしたルールベース型のシミュレーションは，シンプルな作業工程では有効に機能する一方で，複雑な環境になるにつれて，現実の作業の柔軟性のある行動をルール化することは困難になる．また，ルールが複雑になると，設計ミスや例外対応漏れが発生しやすく，シミュレーションの挙動が現実と乖離するリスクも高まる．

本研究では，こうしたルール設計に伴う課題を解決するアプローチとして，実データを用いた作業員行動のモデル化と，そのモデルに基づくシミュレーションの構築を提案する．深層学習を用いて行動の特徴を抽出し，従来のようにルールを明示せずとも作業の進行を再現可能なデータ駆動型シミュレータの実現を目指す．

図 1 に，本研究で提案する手法のフローを示す．我々が対象としている物流倉庫内の天井には，およそ 60 台のカメラで構成されたカメラ基盤が構築されている．本手法ではまず，このカメラ基盤の映像から取得した作業員や物品の位置情報などを二次元的にエンコードして学習用データを作成する．次に，そのデータを用いて作業員の行動を予測する深層学習モデルを訓練する．訓練したモデルをシミュレータに組み込み，シミュレータ内の作業員の行動をモデルの出力により決定する．このフローにより，ルールを明示せずにシミュレータを駆動できる．なお，現在はカメラ

<sup>1</sup> 名古屋大学大学院 工学研究科  
Graduate School of Engineering, Nagoya University

<sup>2</sup> 名古屋大学 未来社会創造機構  
Institutes of Innovation for Future Society Nagoya University

<sup>a)</sup> chanyou@ucl.nuee.nagoya-u.ac.jp

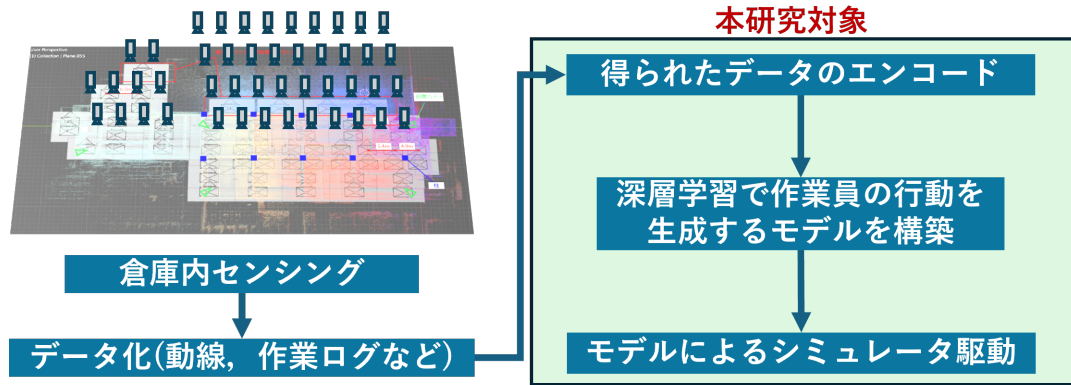


図 1: 提案手法の概要

基盤の映像を活用した作業員の動線や作業内容、物品の移動履歴などのデータ化を進めている段階である [5], [6], [7]. そのため本研究では、その前段階としてルールベースで構築したシミュレータにより倉庫内の作業シナリオを再現し、作業員や物品の位置データを生成して、実データの代替として用いた。このデータを用いて、モデルの行動生成の精度と、モデル駆動によるシミュレータでのタスクの処理完了率を指標として評価実験を行った。

本研究の貢献は、以下のようにまとめられる。

- ルールベースで構築したシミュレータから取得したデータを用いて、作業員行動の特徴を深層学習モデルで表現可能であると示した。
- 学習したモデルにより、ルールを明示的に設計しないシミュレーションの駆動を確認した。
- 実環境データの活用に向けた、シミュレーションと深層学習の連携手法の有効性を検証した。

## 2. 関連研究

### 2.1 マルチエージェントシミュレーションの活用

実環境における複数の要素が相互に影響し合う挙動を模倣・再現する手法として、マルチエージェントシミュレーション (MAS: Multi-Agent Simulation) は広く用いられている。各エージェントが独立した意思決定主体として振る舞う MAS は、個別行動の集積によって全体としての現象を再現可能であり、複雑な社会的・物理的な相互作用のモデル化に適している。実利用の一例として、物流倉庫における出荷作業が挙げられる。出荷時には、作業員が指定された商品を倉庫内の棚から取り出すピッキング作業が発生する。これにおいては、棚のレイアウトや商品の配置、作業員の動線が作業効率に大きな影響を与える。こうした複雑な要因の評価には、MAS ベースのシミュレーションが有効であり、実際に企業によるシステム実装事例 [3] が存在する。また、研究の分野でも MAS は、複雑な相互作用を含む社会的・行動的な現象の定量的分析に活用されている。たとえば、イベント会場からの分散退場による駅構

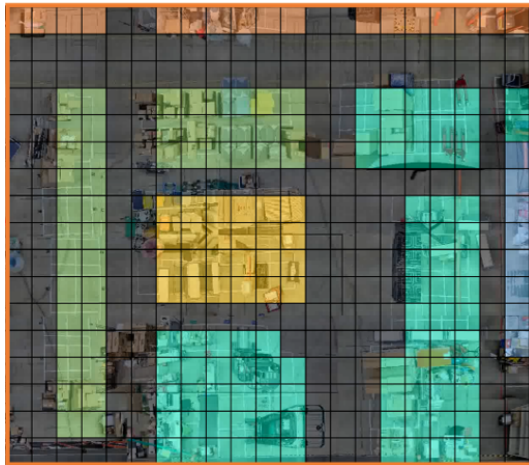
内の混雑影響を評価する研究 [8] や、鬼ごっこにおける戦略の違いによる有利・不利を検討する研究 [9] などがある。

### 2.2 機械学習を用いたマルチエージェントの行動モデリング

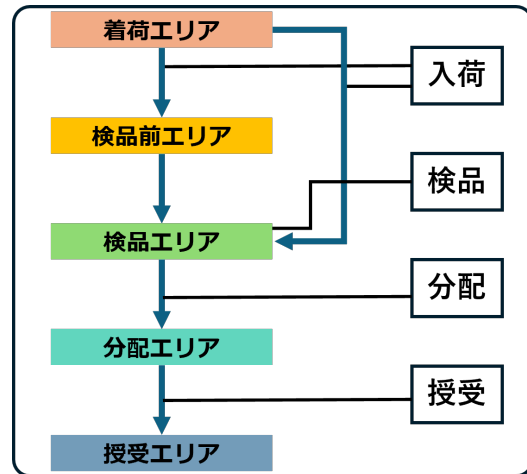
近年では、ルールベースの行動設計に代わって、実データに基づき行動を学習する機械学習アプローチも注目されている。Alahi らの研究では、混雑した環境における人間の動きを対象とし、各人間の軌道を LSTM (Long Short Term Memory) を用いてモデル化した [10]。従来の系列モデルでは独立に各人間の動きを予測する構成が一般的であったが、本手法では複数の人間の LSTM 間で情報共有をする Social Pooling 層を導入して、衝突回避や協調的な移動等の相互作用を考慮した行動予測を実現している。藤井らは、チームスポーツにおける選手の行動データを用いて、部分的な観測と生体的な制約を考慮した分散型の行動モデリング手法を提案した [11]。本手法では、各エージェントが独立に周囲の環境を観測しながら行動を生成する。系列生成モデルとしては変分再帰ニューラルネットワーク (VRNN) を用いている。損失関数に運動コストにかかわる項を追加して、現実的な行動生成を実現している。また、物流分野において、人間とロボットが混在するマルチエージェント環境で戦略的な協調を強化学習で学習させる研究がある。Krnjaic らは、倉庫内のピッキングタスクにおいて、ピッカーと AGV (自動搬送車) が協力して作業する状況を階層型のマルチエージェント強化学習 (MARL) により最適化する手法を提案した [12]。この研究では、マネージャエージェントが倉庫内の領域をタスクとして分割し、各ワーカーエージェントに割り当てる構造で、行動空間の分解と協調の促進を実現している。

## 3. 提案手法

本章では、物流倉庫を対象とする、深層学習を用いたモデル駆動のシミュレータの構築について説明する。前章で述べたように、従来の研究では、マルチエージェント環境



(a) 倉庫内のエリア構成



(b) エリアに対応した業務内容と流れ

図 2: 倉庫内のエリア構成と業務内容

における個別エージェントの行動モデリングに焦点を当て、LSTM や VRNN を用いた系列予測や、強化学習による戦略学習などが提案されてきた。本研究では、こうした個々の行動予測の枠組みを基に、シミュレーション全体の進行をモデルで駆動することを目的とする。

個々の判断と行動の集積による全体の振る舞いをモデル化するためには、その前提となる行動意図の推定をモデルに組み込む必要があると考えた。そこで、本研究では行動選択を 2 段階に分け、まず作業員の注目すべき物品の位置を推定し、その後に具体的な行動を決定するという構成をとる。

### 3.1 対象倉庫とその業務

本研究が対象とするのは、倉庫内の業務のうち、外部から到着した物品を受け取り、検品を行った後、適切な位置へ運搬する作業である。倉庫内は、図 2a に示すように、複数のエリアに区分けされている。エリアごとに役割が異なり、図 2b に示すように各エリアで対応した業務が行われる。作業内容は以下の手順で進行する。作業員には、どの手順の作業を担当するか役割が割り当てられる。

- (1) 入荷：外部から到着した物品を着荷エリアで受け取り、検品エリアに運ぶ。検品エリアの占有状況によっては検品前エリアに運ぶ。
- (2) 検品：検品エリアの物品を検品する。
- (3) 分配：検品を終えた物品を物品の特徴に応じて検品後エリアに運ぶ。
- (4) 授受：検品後エリアから授受エリアに運び、物品を次の作業に受け渡す。

また、シミュレーション構築にあたって、庫内のエリアは図 2a の様にセルに区分けたグリッドで表現する。倉庫のシミュレータのマップは図 3 に示すように、このグリッド分割に基づいて作成した。

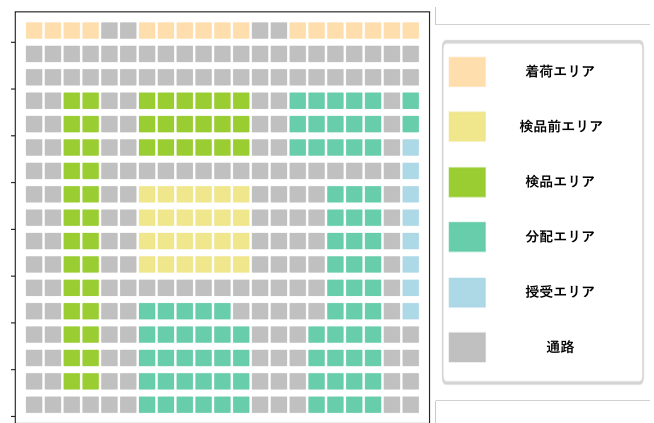


図 3: シミュレータマップ

### 3.2 入力データと学習データセットの構成

後述する深層学習モデルの入力には、取得した倉庫内のデータを空間的にエンコードしたテンソルを用いる。このテンソルは多チャンネルの画像のような形式で、シミュレータのマップと同じ形状を持つ二次元配列の層である。モデルの入力には、以下の情報が含まれる。

- 各物品の位置（状態に応じて、検品前／検品中／検品後に分類）
- 各作業員の位置（入荷／検品／分配／授受の担当を区別）
- 各作業員がそのステップで実行した行動の種別  
この行動は、以下の 17 種類に分類される：
  - － 移動（上・下・左・右の 4 方向）
  - － 物品の持ち上げ（上・下・左・右の 4 方向）
  - － 物品の設置（上・下・左・右の 4 方向）
  - － 検品（上・下・左・右の 4 方向）
  - － 静止
- 各作業員が作業対象とする物品の位置

上記の情報から、表 1 で示す構成のテンソルと、行動の正

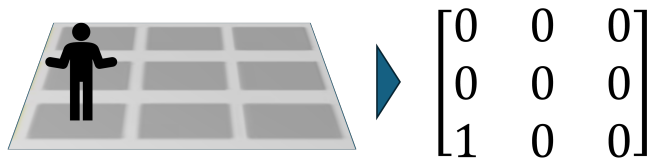


図 4: エンコードのイメージ

表 1: 入力テンソルにおける各チャネルの意味

チャネル	内容
1	検品前の物品の位置
2	検品中の物品の位置
3	検品後の物品の位置
4	作業員の現在位置
5	作業員が作業対象とする物品の位置
6-9	他作業員の現在位置 (6: 入荷, 7: 検品, 8: 分配, 9: 授受)
10-13	作業員自身の役割情報 (ワンホット) (10: 入荷, 11: 検品, 12: 分配, 13: 授受)
14	Hold 状態情報 (作業員が物品を保持しているか)

解ラベルを作成する。テンソルの各チャネルはバイナリ値 (0 または 1) を持つ。図 4 に示したエンコードのイメージのように、対象となる物体や作業員の「存在する位置」を 1 として表現して庫内の状態を空間的に記述している。チャネル 1~3 は、それぞれ検品前、検品中、検品後の物品の位置を表している。チャネル 4 は作業員の現在位置を表しており、チャネル 5 は、その作業員が運ぶべき、もしくは検品するべき、といった、作業の対象とする物品の位置を表している。チャネル 6~9 は、他の作業員の位置を、役割ごとに分けて表現している。また、チャネル 10~13 およびチャネル 14 については、他のチャネルとは異なり空間的な情報を直接表現しているわけではない。これらのチャネルは、対象作業員の属性を表すものであり、対応するチャネルにおいて、マップ上のすべてのセルが 1 となる形式 (全 1 テンソル) で表現される。例えば、対象作業員が検品担当である場合、チャネル 11 (検品チャネル) の全体が 1 となり、他の属性チャネルはすべて 0 となる。同様に、対象作業員が物品を持ち上げたあと、設置するまでは作業員は物品を保持している Hold 状態となるがその場合、チャネル 14 (Hold 状態情報チャネル) の全体が 1 となる。このように、状態をチャネルに分離して表現するエンコード法は、将棋などの盤面ゲームにおける深層学習モデルでも採用されている [13]。

なお、表 1 に示した各情報は、現在構築中のデジタルツイン基盤により実際に取得可能な構成を想定している。作業員や物品の位置はカメラ映像から取得でき、行動や状態は作業員の持つスマートフォンのセンサ情報から推定できる。

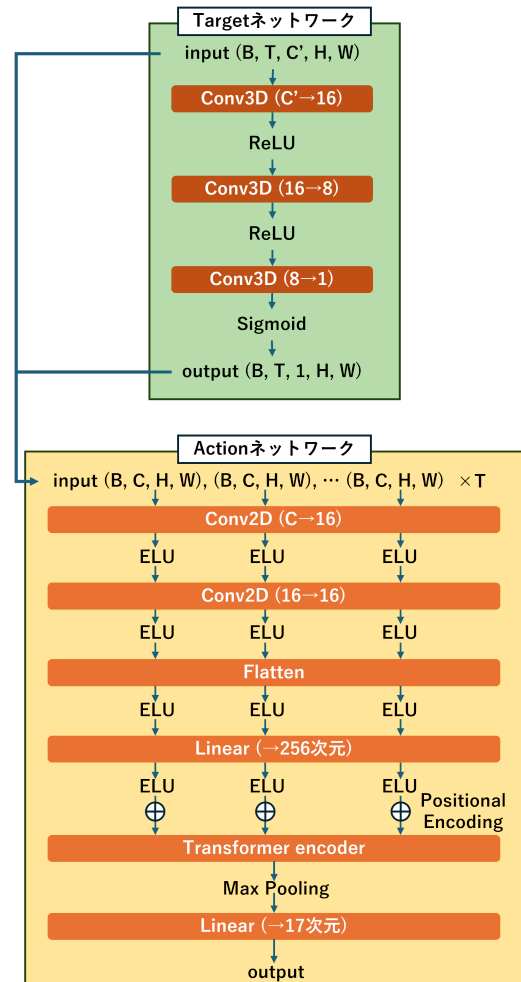


図 5: 提案モデルの全体構成図

### 3.3 モデルの構成

#### 3.3.1 モデル全体の概要

モデル全体の概要を図 5 に示す。本モデルは、倉庫内の状況に基づいて作業員の次の行動を予測する 2 段階構成のニューラルネットワークであり、Target ネットワークと呼ぶ部分と、Action ネットワークと呼ぶ部分からなる。最初の Target ネットワークで作業員が注目すべき物品の位置を予測し、その結果を含む情報を用いて、Action ネットワークが次の行動を 17 クラスから分類する。

入力は、3.2 節で述べた構成に従ったテンソルを  $T$  ステップ分履歴をスタックした、五次元テンソル  $(B, T, C, H, W)$  として与えられる。ここで、 $B$  はバッチサイズ、 $C$  はチャネル数、 $H$  および  $W$  はそれぞれマップの縦・横サイズを表す。なお、履歴が存在しない序盤のステップでは、すべての値が 0 の同じ形状のテンソルでパディングする。

#### 3.3.2 Target ネットワークの構成

Target ネットワークは、作業員が次に操作すべき対象物品の位置を予測するためのモジュールである。本ネットワークの目的は、倉庫内の状態に基づいた、作業員が注目すべき物品のマップ上での位置の推定である。入力として

は、3.2 節で述べた状態テンソル列のうち、チャンネル 5（作業員が対象とする物品の位置）を除いたテンソルを用いる。このテンソルは、 $T$  ステップ分の履歴を含んだ 5 次元テンソル  $(B, T, C', H, W)$  の形状をしており、ここで  $C'$  は 14 チャンネル中、チャンネル 5 を除いた 13 チャンネルである。

このネットワークでは、Conv3D 層を 3 層重ねて時系列的かつ空間的な特徴を同時に抽出する。それぞれの Conv3D 層の後には ReLU 活性化関数を適用し、最終層ではセルごとに注目度を表すスコアを出力するため、sigmoid 関数を用いる。出力テンソルの形状は  $(B, T, 1, H, W)$  であり、これを次の Action ネットワークへの入力に用いる。

### 3.3.3 Action ネットワークの構成

Action ネットワークは、作業員が次取るべき行動を分類するモジュールである。入力は、3.2 節で構成したテンソルに、Target ネットワークによって予測された対象物の位置を挿入したものを用いる。本ネットワークでは、まず各時刻のテンソルに対して Conv2D を適用し、空間的な特徴を抽出する。Conv2D 層は 2 層構成で、それぞれ ELU 活性化関数を用いて非線形変換を行う。得られた各特徴マップは Flatten され、全結合層によって 256 次元の特徴ベクトルに変換される。時系列方向の関係性を捉えるため、ベクトル列に Vaswani らの手法に従って Positional Encoding を加え [14]、Transformer Encoder（3 層、head 数 4）に入力する。Transformer の出力は、時系列方向に最大値プーリングを行って 1 つのベクトルに集約する。最終的に全結合層を通じて、17 クラスの行動分類を行い、作業員の次の行動を推定する。

### 3.3.4 モデルの学習とハイパーパラメータ

モデルは、3.3.1 節で述べた 5 次元テンソルを用いた教師あり学習を行う。学習の流れは以下の通りである。

- (1) 入力テンソルからチャンネル 5（作業員が対象とする物品の位置）を除いたテンソルを Target ネットワークに入力し、対象物の位置マスクを予測する。
- (2) 予測されたマスクをチャンネル 5 として元のテンソルに挿入し、Action ネットワークに入力する。
- (3) Target ネットワークの出力に対しては、正解マスクとの間で各セルそれぞれでバイナリクロスエントロピー損失 (BCELoss) を計算する。
- (4) Action ネットワークの出力に対しては、正解行動との間でクロスエントロピー損失 (CELoss) を計算する。
- (5) これら 2 つの損失を加重平均 (1:2) で合成し、全体の損失を求める。
- (6) 逆伝播を行い、Adam オプティマイザによってモデルパラメータを更新する。

学習時のハイパーパラメータを表 2 に示す。

### 3.4 モデルによるシミュレータの駆動

学習済みのモデルをシミュレータに組み込み、モデルの

表 2: 学習におけるハイパーパラメータ

モジュール	項目	値
Conv3D (Target)	出力チャンネル数	16, 8, 1
	カーネルサイズ	(3,3,3)
	ストライド	1
	パディング	1
	活性化関数	ReLU, Sigmoid
Conv2D (Action)	出力チャンネル数	16, 16
	カーネルサイズ	(3,3)
	ストライド	1
	パディング	1
	活性化関数	ELU
Transformer エンコーダ	層数	3
	ヘッド数	4
	隠れ次元	256
	Dropout 率	0.1
その他	Optimizer	Adam
	バッチサイズ	32
	損失関数	BCE, CE
	学習率	$10^{-4}$
	エポック数	15

予測によるシミュレータの駆動を行う。モデル駆動のシミュレータに用いる環境は、ルールベースのシミュレータと同様のレイアウトおよび構成要素を持つ。

シミュレーションは、以下の手順で進行する。

- (1) シミュレーション開始時に作業員がランダムな位置に配置される。
- (2) ステップごとに、3.2 節で述べた構成のテンソル（チャンネル 5 は除く）を各作業員に対して作成する。
- (3) 作業員ごとに、テンソルをモデルに入力し、次の行動を予測する。
- (4) モデルの出力に基づいて、作業員が行動を取る。  
ただし、以下のような明らかなルール違反行動に関しては、シミュレータ側で実行をキャンセルし、作業員は静止する。

- マップ外への移動
- 他の作業員が保持している物品を持ち上げようとする行動
- 物品を置けない場所への設置行動

上記の手順を 1 ステップごとに繰り返してモデル駆動によるシミュレーションが進行する。これにより、従来のルール記述を用いずに、作業員が適切な物品を選択し、搬送・検品・授受を行う一連の業務を再現できるかが検証できる。

## 4. 評価実験

提案手法の有効性を確認するため、モデル駆動のシミュレーションを実行し、その出力を評価した。評価は 2 つの観点から行った。ひとつは、モデルによるターゲット物品



の予測精度と行動選択の分類性能をそれぞれ個別に評価する観点である。もうひとつは、モデルが出力した行動によって実際にシミュレータを駆動した際に、作業シナリオ全体が正しく完了するかどうかという観点である。本章では、それぞれの評価実験の詳細と結果を示す。

#### 4.1 評価環境とデータセット

学習に用いるデータセットの整備には、ルールベースのシミュレータを用いた。本シミュレータは、3.1節で述べた倉庫業務を再現するために、Pythonのエージェントベースシミュレーションフレームワークである Mesa[15]を用いて弊研究室で構築し、研究に利用しているものである[16], [17]。

本シミュレータは、複数のエージェントが個別に動作しながら、倉庫内の物品の処理をステップバイステップで進行させるマルチエージェントシステムである。各エージェントは、作業員や物品などの役割を持ち、それぞれが独立したルールに従って動作して、全体の作業が進行するように設計されている。作業員エージェントには実際の倉庫業務に倣って入荷、検品、分配、授受の担当が割り当てられ、それぞれに応じた行動を取る。また、物品エージェントは検品などの処理段階に応じて状態が更新される。これらの動作は図3の様に、実際の倉庫環境を模したマップ上で行われる。以下に、シミュレーション進行の概要を示す。

- (1) シミュレーション開始時に、指定された人数の作業員がランダムに倉庫内へ配置され、それぞれに入荷・検品・分配・授受のいずれかの役割が割り当てられる。
- (2) あらかじめ設定された時刻に指定個数の物品が着荷エリアに出現する。
- (3) 入荷担当の作業員が着荷エリアから物品を回収し、検品エリアまたは検品前エリアに運搬する。
- (4) 検品担当の作業員が、検品エリアにある物品に対して検品を行う。検品には事前に設定された所要時間がかかる。
- (5) 分配担当の作業員が、検品を終えた物品を物品の種類に応じて検品後エリアへと運搬する。
- (6) 授受担当の作業員が、検品後エリアの物品を授受エリアに運搬する。
- (7) 授受エリアに運ばれた物品は、あらかじめ定められた間隔で自動的に回収される。

このような一連の処理フローが、シミュレーション全体を通して複数の物品に対して繰り返し発生する。このフローの中で、3.2節で述べたデータが各タイムステップごとに記録される。

#### 4.2 学習データセット

評価には、ルールベースシミュレータから取得したシミュレーションデータを用いた。300ステップを1エピソード

表 4: 行動種別ごとの精度

表 3: Target 予測の各指標		行動種別	
指標	値 (%)	正解率 (%)	
正解率 (Accuracy)	99.92	全体	94.45
適合率 (Precision)	48.80	移動	86.93
再現率 (Recall)	21.80	検品	99.82
F1 値 (F1-score)	30.14	持ち上げ	97.60
		設置	61.39
		静止	98.59

ピソードとし、合計 3,000 エピソードのデータを取得した。ただし、エピソード終盤において作業員の静止状態が最後まで続くものは除外した。除外処理後のステップ数の合計は 6,789,165 件であった。各ステップごとに、3.2節で述べた形式の状態テンソルおよび行動ラベルを記録した。マップサイズは  $H = 17$ ,  $W = 21$ , 作業員は 8 人（入荷・検品・分配・授受に各 2 名）で構成した。取得データは、学習用・検証用・テスト用にそれぞれ 6:2:2 の割合で分割した。各作業員が実行した行動に応じて、17 クラスの行動ラベルのいずれかが付与されている。学習時に含むステップ数  $T$  は、 $T = 5$  とした。

#### 4.3 Target 予測および行動分類

Target 予測に関する評価結果を表 3 に示す。これは、Target ネットワークの出力に対して 0.5 を閾値として二値化して、セルごとに正解マスクと比較して各指標を計算したものである。全体のセル数に対する正解率は 99.92% と高い値を示しているが、注目すべき対象物品のセルはごく一部に限られるため、適合率や再現率は相対的に低く、F1 スコアは 30.14% であった。これは対象物の存在領域が非常に局所的であり、クラス不均衡が影響しているものと考えられる。

行動分類について、種別ごとの正解率を表 4 に示す。全体の正解率は 94.45% と高く、特に検品や、持ち上げ、静止の行動は高い精度を示している。一方で、設置の行動の正解率については相対的に低く、61.39% に留まっている。これは、設置動作において複数の妥当な設置位置の候補が存在しているからだと考えられる。ルールベースシミュレータではそのような候補の中からランダムに設置位置が選ばれており、同じ状態に対して一意に正解が定まらない構造になっている。そのため、モデルが出力した設置位置が適切であったとしても、教師データとの不一致として扱われる場合がある。このように、行動の評価においては正解率だけでは捉えきれない側面が存在するため、次節では、モデルが出力した行動によって実際にシミュレータを駆動した際に、作業シナリオ全体が正しく進行するかを検証する。

#### 4.4 モデル駆動シミュレーションによるタスク実行

モデルによって出力された行動をもとにシミュレータを

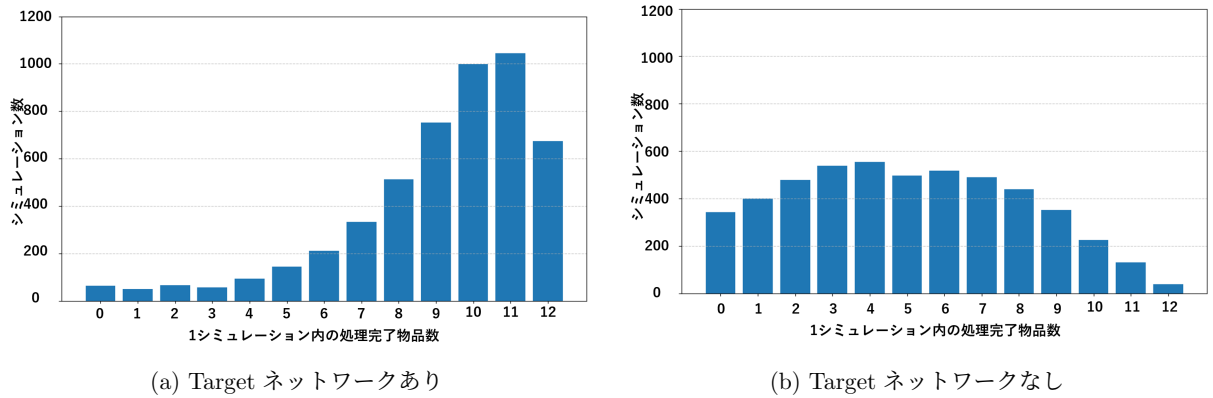


図 6: シミュレーションごとの処理完了物品数の分布

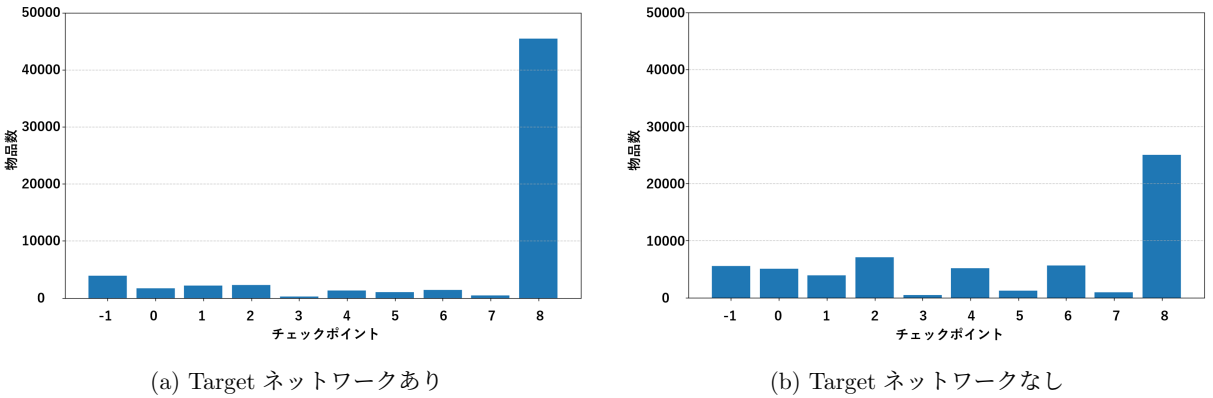


図 7: 各物品の到達チェックポイントの分布

表 5: 物品処理のチェックポイント

チェックポイント名	内容
1	着荷エリアから物品が持ち出された
2	検品エリアに物品が設置された
3	検品が開始された
4	検品が終了した
5	検品エリアから物品が持ち出された
6	分配エリアに物品が設置された
7	分配エリアから物品が持ち出された
8	授受エリアに物品が設置された

表 6: モデル駆動シミュレーションにおける完遂率

モデル構成	処理完了率 (%)	全処理達成率 (%)
提案モデル (Target あり)	75.27	12.70
比較モデル (Target なし)	42.17	0.62

駆動し、物品の処理が完了するかどうかを検証した。処理の完了は、シミュレーション内に 8 つのチェックポイントを設けて測った。チェックポイントの内容を表 5 に示す。これらのチェックポイントを過不足なく、順番に通過したものを完了とみなす。

評価では、倉庫内に各役割 (入荷・検品・分配・授受) を担当する作業員を 2 名ずつ、合計 8 名を初期配置した。物品はシミュレーション中に一定間隔で断続的に到着し、700

ステップの間に合計 12 個が着荷する設定とした。この環境で、700 ステップまでのシミュレーションを 5,000 回実行しモデルの動作を評価した。また、提案モデルの有効性を検証するために、Target ネットワークの有無による 2 構成を比較した。比較対象では、チャンネル 5 (作業員が注目すべき物品の位置) を除いたテンソルを入力とし、Action ネットワークのみで行動を予測する。

表 6 に、提案モデルおよび比較対象 (Target ネットワークなし) における結果を示す。処理完了率とは、シミュレーション中に到着した物品のうち、検品などの必要な工程を経て授受エリアまで到達した物品の割合を指す。一方、全処理達成率は、1 回のシミュレーション内で到着したすべての物品 (12 個) が処理されたケースの割合である。提案モデルでは、到着した物品のうち 75.27% が処理を完了できた。また、5,000 回のうち 12.70% のシミュレーションでは 12 個すべての物品が処理された。一方、Target ネットワークを用いない構成では、処理された物品の割合は 42.17% にとどまり、完全に処理されたシミュレーションはわずか 0.62% であった。

次に、モデル駆動によるシミュレーションの詳細な挙動を確認するため、シミュレーションごとの処理完了物品数の分布および各物品が到達したチェックポイントの分布を分析した。それぞれの結果を図 6、図 7 に示す。図 6 は、

1 シミュレーションあたりに処理が完了した物品数の分布である。Target ネットワークを用いた構成 (図 6a) では、12 個すべての物品を処理できたケースが多く、加えて 10 個以上処理できたシミュレーションの割合も高い。一方、Target ネットワークなしの構成 (図 6b) では、処理できた物品数が 5 個以下に偏る傾向が見られ、タスクの完遂において明確な差が存在した。図 7 は、各物品が最終的にどのチェックポイントまで到達したかを集計したものである。-1 は、チェックポイントを過剰に通過して失敗したことを示す。Target あり構成 (図 7a) では、多くの物品が最終チェックポイント (8) まで到達しており、途中で処理が止まったケースも少なく、タスク全体が一貫して進行していたことが分かる。一方 Target ネットワークなしの構成 (図 7b) では、特定のチェックポイントで停滞している物品が目立つ。これらは、検品の開始やエリア間の搬送など、適切なタイミングと対象の物品の選択が必要な場面である。これらの結果から、Target ネットワークによる注目物品の明示的な推定が、適切な行動推定に寄与していることが分かる。

## 5. まとめと今後の展望

本研究では、物流倉庫内の業務のシミュレーションを、深層学習モデルによって駆動する手法を提案した。用いたモデルは、物品の注目対象を推定する Target ネットワークと、行動を選択する Action ネットワークの 2 段階構成からなる。ルールベースで構築したシミュレータの出力を用いて学習した結果、行動正解率は 94.45% を達成した。また、モデル駆動によるシミュレーションを実行した結果、75.27% のタスクを完遂し、明示的なルールなしでも業務再現が可能であると示した。

しかし、実用可能なシミュレータとしては十分とは言えない水準に留まっており、さらなる精度向上が必要である。また、本研究ではルールベースシミュレータによって得られたデータを用いて学習を行っており、今後は実データの導入が課題となる。

**謝辞** 本研究の一部は、国立研究開発法人新エネルギー・産業技術総合開発機構 (NEDO) の委託業務 (JPNP23003)、科研費挑戦的研究 (開拓) 22K18422、トラスコ中山株式会社に支援いただいている。

## 参考文献

- [1] 経済産業省：令和 5 年度電子商取引に関する市場調査報告書，<https://www.meti.go.jp/press/2024/09/20240925001/20240925001-1.pdf> (2024).
- [2] VanDerHorn, E. and Mahadevan, S.: Digital Twin: Generalization, characterization and implementation, *Decision Support Systems*, Vol. 145, p. 113524 (2021).
- [3] 株式会社 NTT データ数理システム：シミュレーション適用事例：倉庫内ピッキング作業の効率化，株式会社 NTT データ数理システム (参照 2025-03-21).

- [4] The AnyLogic Company: Warehouse Logistics Simulation Model, The AnyLogic Company (accessed 2025-03-21).
- [5] Asai, Y., Mori, Y., Higashiura, K., Yokoyama, K., Katayama, S., Urano, K., Yonezawa, T. and Kawaguchi, N.: Towards a Real-Time and Energy-Efficient Edge AI Camera Architecture in Mega Warehouse Environment, *2024 IEEE 3rd Real-Time and Intelligent Edge Computing Workshop (RAGE)*, pp. 1-6 (2024).
- [6] Mori, Y., Asai, Y., Higashiura, K., Katayama, S., Urano, K., Yonezawa, T. and Kawaguchi, N.: Efficient Edge AI based Annotation and Detection Framework for Logistics Warehouses, *IEEE Consumer Communications & Networking Conference 2025 (CCNC)* (2025).
- [7] 渡邊章章, 加納一馬, 片山 晋, 浦野健太, 米澤拓郎, 河口信夫: 物流倉庫におけるスマートフォンを用いた作業者のタスク認識, 電気・電子・情報関係学会 東海支部連合大会 (2024).
- [8] 中村佑輔, 石黒 慎, 落合桂一, 鈴木 喬, 山田 暁: マルチエージェント・シミュレーションを用いたイベント観客の分散退場が駅の混雑に与える影響の分析, 情報処理学会論文誌デジタルプラクティス (TDP), Vol. 6, pp. 99-107 (2025).
- [9] 安井一真, 宇都宮陽一, 奥田隆史: マルチエージェントシミュレーションを用いた鬼ごっこにおける鬼の作戦の評価, Vol. 2022, 情報処理学会, pp. 31-32 (2022).
- [10] Alahi, A., Goel, K., Ramanathan, V., Robicquet, A., Fei-Fei, L. and Savarese, S.: Social LSTM: Human Trajectory Prediction in Crowded Spaces, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016).
- [11] Fujii, K., Takeishi, N., Kawahara, Y. and Takeda, K.: Decentralized policy learning with partial observation and mechanical constraints for multiperson modeling, *Neural Networks*, Vol. 171, pp. 40-52 (2024).
- [12] Krnjaic, A., Steale, R. D., Thomas, J. D., Papoudakis, G., Schäfer, L., Keung To, A. W., Lao, K.-H., Cubuk-tepe, M., Haley, M., Börsting, P. and Albrecht, S. V.: Scalable Multi-Agent Reinforcement Learning for Warehouse Logistics with Robotic and Human Co-Workers, *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 677-684 (online), DOI: 10.1109/IROS58592.2024.10802813 (2024).
- [13] Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., Lanctot, M., Sifre, L., Kumaran, D., Graepel, T., Lillicrap, T., Simonyan, K. and Hassabis, D.: A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play, *Science*, Vol. 362, No. 6419, pp. 1140-1144 (2018).
- [14] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. and Polosukhin, I.: Attention is all you need, *Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS'17*, Red Hook, NY, USA, Curran Associates Inc., p. 6000-6010 (2017).
- [15] Masad, D. and Kazil, J.: Mesa: An Agent-Based Modeling Framework, pp. 51-58 (2015).
- [16] 高木魁利, 瀧上昂希, 片山 晋, 浦野健太, 米澤拓郎, 河口信夫: 量子アニーリングとブラックボックス最適化を用いた倉庫内業務最適化 (2024).
- [17] 瀧上昂希, 福島悠人, 浅井悠佑, 片山 晋, 浦野健太, 米澤拓郎, 河口信夫: 倉庫内業務の量子アニーリングによる最適化とシミュレーションによる評価 (2024).