Vol.2022-DPS-191 No.44
Vol.2022-MBL-103 No.44
Vol.2022-ITS-89 No.44
2022/5/27

ロボット周辺の非占有面積を用いた複数ポリシー切替によるナビゲーション手法の検証

天野 加奈子[1]　加藤 由花[1]

概要：近年，多様な動的環境を対象とした，深層強化学習による自律移動ロボットのナビゲーション手法が研究されている．しかし，学習を行うシミュレーション環境と実際の環境には差異が存在するため，学習された方策を直接現実世界に適用することは安全面および効率面で最適ではない場合がある．これに対し，複数の行動決定方法を組み合わせるアプローチが提案されているが，既存手法では狭い環境や混雑した状況下において危険性の高い状況の見落としや移動速度の低下が発生する．これらの問題を解決するため，我々はロボット周辺の非占有面積を用いた状況判定方法およびリセットポリシーを導入した複数ポリシー切り替えによるナビゲーション手法を提案してきた．本稿では，リセットポリシーのアルゴリズム設計と提案手法の実装を行い，シミュレーション環境におけるナビゲーション実験によって提案手法の性能を検証する．そして，従来手法ではロボットの振動や停止が起こりやすい狭い環境において，提案手法を用いることで成功率が向上することを示す．

1. はじめに

近年，労働力不足や対面サービスによる感染症拡大といった社会課題を背景に，人間に代わって案内や清掃等を行うサービスロボットの需要が高まっている．サービスロボットは，人間との接触や稼働範囲が限られる環境で活動する産業用ロボットとは異なり，人間の生活空間内で活動する．そのため，サービスロボットは人の動きや状況の変化に柔軟に対応しつつ，与えられた任務を遂行することが求められる．中でも移動ロボットの場合は，多様な環境下において安全かつ効率的に移動可能であることが必要とされる．

現在まで，人が存在する動的環境を対象とした自律移動ロボットナビゲーションについて，多くの研究が行われている．従来のアプローチの一つは，観測した歩行者の軌跡から将来の歩行経路を予測し，その予測経路を回避するようナビゲーションを行う方法である．しかし，人が密集した環境では，いずれの行動も衝突可能性が高いと判断し，ロボットが振動や停止をする「Freezing Robot Problem」（FRP）が発生することが知られている．FRP の解決にはエージェント同士の協調行動を考慮することが有効であると考えられている [1]．また，経路予測を行う際には各歩行者の検知および追跡が必要となるが，ロボットに取り付けたセンサのみを用いる場合，混雑した環境では歩行者が交差することでデータの欠損が発生し，十分な精度を得られない可能性がある．これに対し，近年では歩行者が存在する環境において柔軟なナビゲーションを可能にすることを目的に，深層強化学習（DRL）を用いたナビゲーション手法が研究されている．DRL 手法では，ロボットが学習環境の中で膨大な試行錯誤を繰り返すことで行動を学習するため，センサ値や移動の誤差といった不確実性のある環境においてロバストな制御が可能である．また，センサ計測値または周囲の人間の情報を入力とし，エージェント同士の協調行動を暗黙的もしくは明示的にモデル化するため，周囲の人間の動きを考慮した行動を学習可能だと考えられる．一方，安全性の観点から学習はシミュレータ上で行われることが一般的だが，シミュレータと実世界の環境には差異があるため，学習した行動を実際のロボットに適用する際に安全性および効率性の保証が困難という問題がある．そこで，DRL 手法の柔軟性を生かしつつ，実世界においても安全性や効率性の高い行動計画を行うため，複数のポリシーを切り替えるアプローチが提案されている．Sathyamoorthy ら [2] は FRP に対処するためのアルゴリズム Frozone を考案し，人の密度によって Frozone と DRL ポリシーを切り替えるアプローチを提案している．この手法では，密度が一定以下のとき Frozone を用いることで，特定の状況において歩行者に対する親和的な行動や FRP の抑制を安定して行うことを可能にしている．しか

[1] 東京女子大学 大学院理学研究科
Suginami, Tokyo 167–8585, Japan

Vol.2022-DPS-191 No.44
Vol.2022-MBL-103 No.44
Vol.2022-ITS-89 No.44
2022/5/27

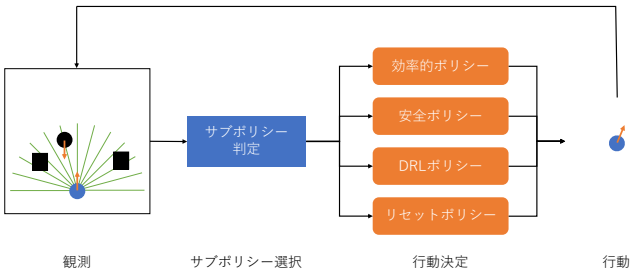


図 1: 提案手法の概要

し，Frozone は保守的な行動を生成するため，人の少ない場合に非効率な行動をとる可能性がある．また，密度が高い場合は DRL ポリシーをそのまま利用するため，シミュレータと実世界のギャップに対処できない可能性がある．Fan ら [3] は障害物との最短距離に応じ，DRL ポリシー，DRL ポリシーの入出力を制限する Safe DRL ポリシー，PID 制御を切り替える手法を提案している．この手法は移動障害物が多い環境であっても機能するが，静止障害物や壁に囲まれた狭い環境ではロボットが停止や振動を繰り返すといった状況に陥りやすいという問題がある．

本稿では，我々が提案した，ロボット周辺の非占有面積を用いたシナリオ判定方法とリセットポリシーを導入した複数ポリシー切り替えアプローチによるナビゲーション手法 [4] の性能をシミュレーションによって検証する．提案手法の概要を**図 1** に示す．本手法は，ポリシー切替におけるシナリオ判定の際に，ロボット周辺の移動可能な領域の面積を用いることで，衝突可能性が高い状況や低い状況のほかに，ロボットが身動きできなくなる可能性が高い状況を判別することを可能にする．そして，各状況において，効率ポリシー，安全ポリシー，リセットポリシー，DRL ポリシーという 4 つのサブポリシーを切り替えることによって，多様な環境に適応したナビゲーションを行う．リセットポリシーは，ロボット周辺の非占有領域が大きい方向へと進むように設計され，潜在的な衝突可能性が高い状況やロボットが身動きを取れない状況を脱するために利用される．今回，提案手法のナビゲーション性能を検証するため，リセットポリシーのアルゴリズムを設計し，複数のシミュレータ環境においてナビゲーション実験を行った．その結果，従来手法ではロボットの振動や停止が起こりやすい狭い環境において，提案手法を用いることで成功率が向上することを示した．

2. 関連研究

我々の提案手法では，多様な動的環境において安全性と効率性を両立する柔軟な行動決定を可能にするため，DRL ポリシーを含む複数ポリシーを切り替えるハイブリッドアプローチを採用している．本章では，関連研究として深層強化学習を用いたナビゲーション手法と状況に応じて複数のポリシーを切り替えるハイブリッドアプローチによるナビゲーション手法について取り上げる．

2.1 深層強化学習によるナビゲーション

近年，動的環境において，安全かつ効率のよいロバストなロボットナビゲーションの実現を目的とし，エージェント間の相互作用や実世界の不確実性を考慮する深層強化学習を用いたナビゲーション手法が研究されている．深層強化学習によるナビゲーション手法はエージェントレベルとセンサレベルのアプローチに分けられる．エージェントレベルのアプローチ [5] は，歩行者等の各エージェントの位置や速度を入力として用いる．そのため，明示的に各エージェント間の相互作用を学習に取り入れることができ，無駄な行動が少なく効率的な移動が可能である．一方で，各エージェントの位置や速度の情報を利用するため，移動エージェントの検知・追跡モジュールが別途必要となる．また，その精度がロボットの行動決定に影響を及ぼす可能性がある．センサレベルのアプローチ [6] は，センサによる計測値をロボットの制御指令値に直接マッピングする End-to-end のポリシーを学習するアプローチである．これは，歩行者の検知等を明示的に行う必要がないため，ロバストな制御が可能である．一方，エージェントレベルのアプローチと比較して学習が難しいということがデメリットとして挙げられるが，センサデータの表現の工夫 [7] や段階的な学習 [6] による性能向上が図られている．

また，一般的に，強化学習はモデルの学習に膨大な時間を要し，学習途中では危険性のある行動を選択する可能性がある．そのため，学習はシミュレータ上に構築された学習環境で実行される．このシミュレータ上の環境と実環境では，障害物の配置やエージェントのモデル等，様々な差異が存在する．そのため，シミュレーションで学習したモデルを直接実機に適用する際，目的とする動作が得られない可能性がある．この問題は Sim-to-Real ギャップと呼ばれる．これに対処するため，Rusu らによるプログレッシブネットアーキテクチャ [8] や Sadeghi らによる訓練セットのレンダリング設定を高度にランダム化する学習方法 [9] が提案されている． 2.2 節でも紹介する Fan らのハイブリッド制御アプローチ [3] は，DRL ポリシーを含む複数のポリシーを状況に応じて切り替えることにより，上記の問題に対処しようとするものである．

本研究の提案手法においても，DRL ポリシーの利点を生かしつつ，DRL ポリシーでは性能を発揮できない状況で他のポリシーを利用するため，複数ポリシーを切り替えるハイブリッドアプローチを採用する．今回，提案手法における DRL ポリシーとして，Long らによる 2D-LiDAR をセンサとして使用した End-to-end の衝突回避ポリシー [6] を用いることを前提とする．

2.2 ハイブリッドアプローチによるナビゲーション

複数の行動決定手法を状況に応じて切り替えることで，DRL ポリシーのデメリットをカバーしようとするアプローチが研究されている．Sathyamoorthy ら [2] は FRP に対処するためのアルゴリズムである Frozone を設計し，人の密度によって Frozone と DRL ポリシーを切り替えるアプローチを提案している．この手法では，密度が一定以下の場合において Frozone を用いることで，DRL ポリシーの行動を保証できないという欠点が生じる範囲を狭めており，密度が高い場合には DRL ポリシーを用いることで，混雑した状況においてもロバストな行動を生成できるという DRL ポリシーの利点を活かしている．しかし，Frozone は回避する領域を必要以上に大きくとる可能性があり，非効率な行動が生成される場合がある．また，密度が高い状況では DRL ポリシーを直接利用するため，その際の Sim-to-Real ギャップ問題は解決できていない．Fan ら [3] は，実環境に DRL ポリシーを適用する際の安全性と効率性向上を目的とし，ロボットと障害物との最短距離に応じて DRL ポリシー，DRL ポリシーの入出力を制限する Safe DRL ポリシー，PID 制御を切り替える手法を提案している．この手法は，移動障害物が多く混雑した環境において単一の DRL ポリシーよりも高いナビゲーション性能を示し，最高速度や大きさが異なるエージェントが存在する環境にもロバストに対応することが可能である．しかし，周囲に静止障害物が多い環境や壁が近い通路のような環境ではロボットの振動が起こりやすいことが課題として挙げられる．

我々の提案するナビゲーション手法は，混雑した環境や通路等を含む多様な環境において安全性と効率性を高めることを目的としており，複数のポリシーを切り替えるハイブリッドアプローチを用いる．従来手法の課題である狭い環境での振動や衝突を回避するため，ポリシーの切替判定時にロボット周辺の非占有領域の面積を用いて，潜在的衝突可能性や FRP 発生可能性の高い状況を判別する．潜在的衝突可能性や FRP 発生可能性の高い状況と判定された場合，新たに導入したリセットポリシーによって危険性が高い状況の回避やフリーズ状態からの脱出を行う．

3. 準備

本章では，提案手法を構成する要素技術を紹介する．3.1 節では深層強化学習について説明し，3.2 節では効率ポリシーとして用いる PID 制御について説明する．

3.1 深層強化学習

今回，提案手法を構成するサブポリシーの DRL 手法として，Long らが提案した End-to-end のマルチロボットシステムのための分散型衝突回避ポリシー [6] を用いる．選択理由は，2D-LiDAR をセンサとして用いるため，シミュレーションと実世界における観測の差異が比較的少なく，環境中のエージェントを明示的に検知する必要がないためである．本節ではこの手法における問題設定や学習アルゴリズムについて説明する．

なお我々の目的は，マルチロボット環境に限らず，群衆間など多様な動的環境において安全性と効率性を両立したロボットナビゲーションを行うことである．Long らの分散型衝突回避ポリシーはマルチロボットシステムを対象とするが，各ロボットは自分以外のロボットと位置等の情報を共有しておらず，非協力なエージェントが存在する場合にも対応可能といった汎化性能が示されているため，ロボット以外の移動物体が存在する動的環境にも応用可能であると考えられる．

Long らはポリシーを学習するためのマルチシナリオ・マルチステージ学習の枠組みを提案しており，ポリシーはその枠組みの中で方策勾配に基づく強化学習アルゴリズムにより，シミュレータ上の環境で複数のロボットに対して同時に学習される．Long らはマルチロボット衝突回避問題を部分観測マルコフ決定過程（Partially Observable Markov Decision Process; POMDP）として定式化している．N 台のロボットのモデルは全て半径 R を持つ円形の非ホロノミック差動駆動ロボットであることを前提とし，i 番目のロボット（$1 \leq i \leq N$）は，各時刻 t において，環境の状態 s_i^t から，確率的に観測 $\boldsymbol{o}_i^t$ を得る．そして，障害物に衝突せずに現在位置 $\boldsymbol{p}_i^t$ からゴール $\boldsymbol{g}_i$ へ向かうための制御指令値 $\boldsymbol{a}_i^t$ を決定し，実行する．強化学習では，ロボットは行動によって報酬 r_i^t を獲得し，それによって行動全体で得られる報酬の総和の期待値を最大化するように方策 π の更新を行う．

本手法では，各ロボットの観測 $\boldsymbol{o}^t$ は $\boldsymbol{o}^t = [\boldsymbol{o}_z^t, \boldsymbol{o}_g^t, \boldsymbol{o}_v^t]$ のように 3 つの要素で構成される．$\boldsymbol{o}_z^t$ はロボットに取り付けられた 2D-LiDAR センサによる測定値，$\boldsymbol{o}_g^t$ はロボットのローカル極座標系におけるゴールの座標，$\boldsymbol{o}_v^t$ はロボットの速度である．$\boldsymbol{o}_z^t \in \mathbb{R}^{3\times512}$ は，角度範囲 180 deg，最長距離 4 m のセンサで観測される 512 本のレーザスキャンの 3 ステップ分で構成される．制御指令 $\boldsymbol{a}^t = (v^t, w^t)$ は，ロボット前方の速度 $v^t \in (0.0, 1.0)$ とロボット中心の角速度 $w^t \in (-1.0, 1.0)$ を組み合わせたものであり，次の観測 $\boldsymbol{o}^{t+1}$ を受け取るまでの時間 Δt 内で実行される．報酬関数は，

$$r_i^t = ({}^g r)_i^t + ({}^c r)_i^t + ({}^w r)_i^t \tag{1}$$

のように設計されている．右辺の各項はそれぞれ，ゴールへ近づくことに対する報酬，衝突に関する報酬，スムーズな行動に対する報酬であり，

$$({}^g r)_i^t = \begin{cases} r_{arrival} & \text{if } \|\boldsymbol{p}_i^t - \boldsymbol{g}_i\| < 0.1 \\ \omega_g \left(\left\|\boldsymbol{p}_i^{t-1} - \boldsymbol{g}_i\right\| - \|\boldsymbol{p}_i^t - \boldsymbol{g}_i\| \right) & \text{otherwise} \end{cases} \tag{2}$$

Vol.2022-DPS-191 No.44
Vol.2022-MBL-103 No.44
Vol.2022-ITS-89 No.44
2022/5/27

$$({}^{c}r)_i^t = \begin{cases} r_{collision} & \text{if } \left\|\boldsymbol{p}_i^t - \boldsymbol{p}_j^t\right\| < 2R \\ & \text{or } \left\|\boldsymbol{p}_i^t - \boldsymbol{B}_k\right\| < R \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

$$({}^{w}r)_i^t = \omega_w \left|w_i^t\right| \quad \text{if } \left|w_i^t\right| > 0.7 \tag{4}$$

と定義される．$\boldsymbol{B}_k$ は静止障害物である．各パラメータは $r_{arrival} = 15$，$\omega_g = 2.5$，$r_{collision} = -15$，$\omega_w = -0.1$ と設定されている．

方策ネットワークは，入力 $\boldsymbol{o}^t$ から $\boldsymbol{v}_{mean}^t$ を出力する 4 層の隠れ層をもつニューラルネットワークである．ロボットの行動 $\boldsymbol{a}^t$ は，ガウス分布 $N(\boldsymbol{v}_{mean}^t, \boldsymbol{v}_{logstd}^t)$ からサンプリングされる．ここで $\boldsymbol{v}_{logstd}^t$ は，学習中にのみ更新される対数標準偏差である．

強化学習アルゴリズムには，Proximal Policy Optimization（PPO）[10] をマルチロボットシステムに拡張したものを用いる．学習は 2 段階に分かれており，第 1 段階ではロボットのみ存在する環境で学習を行い，第 2 段階ではロボットの他に障害物が存在するような複雑さを増した環境で学習が行われる．

3.2 PID 制御

PID 制御はフィードバック制御の代表的手法であり，出力結果と目標値の差とその微分および積分を利用して制御を行う手法である．操作量 $u(t)$ は

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau)\, d\tau + K_d \frac{de(t)}{d\tau} \tag{5}$$

のように定まる．$e(t)$ は目標値 $r(t)$ と出力値 $y(t)$ の偏差

$$e(t) = r(t) - y(s) \tag{6}$$

である．式 (5) の右辺の各項は，出力と目標値の偏差，偏差の積分，偏差の微分に関する項である．パラメータは比例ゲイン K_p，積分ゲイン K_i，微分ゲイン K_d であり，これらは制御対象に合わせてチューニングを行う必要がある．

4. 提案手法

本章では，我々が提案する独自のポリシー判定方法とリセットポリシーを導入した複数ポリシー切り替えアプローチによるナビゲーション手法 [4] について述べる．

4.1 概要

本手法によるナビゲーションの流れを図 1 に示す．ロボットには，自身に取り付けられた 2D-LiDAR センサによる測定値，現在の速度，およびロボットのローカル極座標系におけるゴール位置が毎時刻与えられることを前提とする．ロボットは，2D-LiDAR センサによる測定値に基づいて，サブポリシーの選択を行う．次に，選択された各サブポリシーに従い，センサの値，現在の速度，ゴール位置を用いて，ロボットがとるべき行動を決定する．最後にロボットが行動を実行する．この一連の流れをロボットがゴールに到達するまで繰り返す．

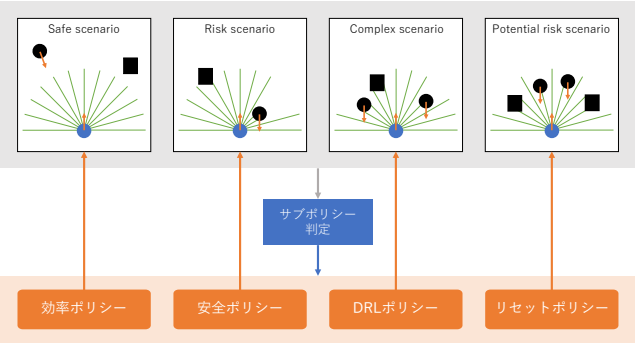


図 2: シナリオの分類

表 1: サブポリシーとシナリオの対応

名称	シナリオ
効率ポリシー	衝突可能性が低い
安全ポリシー	衝突可能性が高い
リセットポリシー	潜在的な衝突可能性や FRP 発生可能性が高い
DRL ポリシー	上記 3 つ以外

また，一般的に，ロボットナビゲーションは大域的経路計画と局所的経路計画に分けて行われる．大域的経路計画では事前に与えられる地図をもとにスタートからゴールまでの経由点を出力し，局所的経路計画では地図上にない障害物や歩行者を回避しつつ，大域的経路計画で導出された経由点を辿ってゴールへ向かうようにロボットの行動決定を行う．本手法は局所的経路計画手法に分類できるが，これ以降，大域的経路計画モジュールの利用を前提とせず，経由点ではなく直接ゴール位置を入力とするものとして説明する．

サブポリシー選択フェーズでは，**図 2** に示すように，最適なポリシーが異なると考えられるシナリオを 4 つに分類し，各シナリオに適したポリシーを選択する．サブポリシーとシナリオの対応を**表 1** に示す．シナリオ判定を行う際の指標として，ロボットからロボットに最も近い障害物までの距離，およびロボット周辺の非占有領域の面積を用いる．ロボット周辺の非占有領域の面積をシナリオ判定に用いることによって，潜在的な衝突可能性や FRP 発生可能性が高い状況を判別でき，リセットポリシーを適用することで，それらの状況を回避することが可能になると考えられる．

4.2 サブポリシー

本手法では，混雑した動的環境において柔軟な行動選択が可能な DRL ポリシーを学習環境と異なる実世界に適用するため，複数のサブポリシーを組み合わせるハイブリッドアプローチを採用している．本手法において選択可能な

Vol.2022-DPS-191 No.44
Vol.2022-MBL-103 No.44
Vol.2022-ITS-89 No.44
2022/5/27

サブポリシーは以下の 4 つとする．

効率ポリシー　未知の実環境における効率性向上を目的とし，障害物を考慮せず高い速度を維持してゴールへ向かう．これは Fan らのハイブリットアプローチにおける PID ポリシーに相当する．本手法においても効率ポリシーとして 3.2 で紹介した PID 制御を用いる．

安全ポリシー　未知の実環境における安全性向上を目的とし，ロボットの速度を制限することで衝突を回避する．これは Fan らのハイブリットアプローチにおける Safe DRL ポリシーに相当する．Safe DRL ポリシーは DRL ポリシーの入出力に制限をかけたものであり，本手法においても同様の方法を採用する．

リセットポリシー　衝突可能性が高いシナリオを回避することや停止または振動状態から脱することを目的とし，ロボット周辺の非占有領域の面積が大きい方向へと進む．今回設計したリセットポリシーの詳細は 4.4 にて説明する．

DRL ポリシー　安全性・効率性の双方を考慮した柔軟な行動選択の実現を目的とし，シミュレータ上で学習された DRL ポリシーを用いる．本稿では，3.1 で説明した Long らによる End-to-end のマルチロボットシステムのための分散型衝突回避ポリシー [6] を用いることを前提とする．

DRL ポリシーとして，Long らの衝突回避ポリシーを選択した理由は，2D-LiDAR センサによる観測値を入力として利用しており，カメラを用いる場合と比較して学習環境と実世界とのギャップが少なく，移動物体の検知・追跡モジュールが不要で混雑した環境においても安定した性能を発揮できると考えられるためである．また，非協力的なエージェントも回避可能であるといった汎化性能が示されており，マルチロボット環境に限らず一般的な動的環境においても適用可能と考えられる．

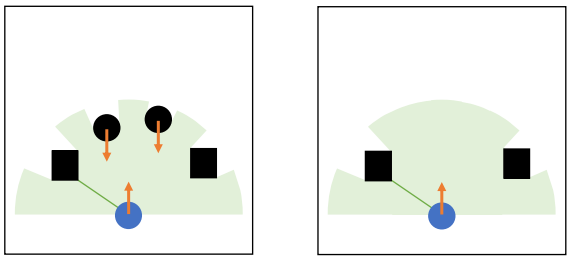
リセットポリシーは，その他 3 つのサブポリシーでは解決できない，ロボットが立ち往生となった状況を脱するために導入される．これにより，従来手法の課題であった狭い環境におけるロボットの振動の回避を目指す．また，安全ポリシーの頻度が多い場合，ロボットの速度の低下によって効率的に移動できないことや，速度変化が多く周囲の人間にとって不自然な動きになることが考えられる．そのため，安全ポリシーが必要とされる状況自体を回避することもリセットポリシーの目的の 1 つとしている．

4.3　シナリオ判定

図 2 にある 4 種類のシナリオの判定方法をアルゴリズム 1 に示す．$\min(l_i^t)$ は時刻 t におけるロボット i のセンサによる観測値の最小値，すなわち最も近い障害物との距離である．$\|p_i^t - g_i\|$ はロボットとそのゴールの距離，s はロボット周辺の非占有領域の面積である．ロボット周辺の非占有領域は，ロボットのセンサの範囲内で障害物が存在しない領域のことを指しており，**図 3** の薄い緑色の部分である．

Algorithm 1 シナリオ判定

```
if min(l_i^t) > R_safe or min(l_i^t) > ||p_i^t − g_i|| then
    Safe scenario (efficient policy)
else if min(l_i^t) ≤ R_risk then
    Risk scenario (safe policy)
else if S ≤ S_risk then
    Potential risk scenario (reset policy)
else
    Scenarios other than the above (DRL policy)
end if
```



(a) 潜在的危険性が高い状況　**(b)** 潜在的危険性が低い状況

図 3: ロボットと障害物間の最小距離では区別できない状況の例．青い円はロボット，黒い円や四角は他の移動物体や静止障害物を表し，オレンジの矢印は移動方向を表す．緑色の線はロボットと最も近い障害物の間に引かれており，(a)，(b) で同じ長さである．薄い緑はロボットの観測範囲を表す．

ロボット周辺の非占有領域の面積 s は，$\min(l_i^t)$ では区別できない潜在的な衝突可能性や FRP 発生可能性が高いシナリオを判別するため，本手法において導入される．例えば図 3 のように，最も近い障害物との距離 $\min(l_i^t)$ は同じだが，非占有領域である薄い緑色の部分の面積 s は異なるといった状況が存在する．**図 3a** は**図 3b** の状況に比べてロボットが移動する方向に障害物が多く，非占有領域が小さいため，図 3a は図 3b よりも衝突可能性が高い状況であると判断する．s は，センサのレーザの本数を m，センサの角度範囲を $[r_{min}, r_{max}]$ として，

$$s = \sum_{j=1}^{m} \frac{{l_j}^2}{2} \cdot \frac{r_{max} - r_{min}}{m} \tag{7}$$

のように求める．ここで，l_j は，その時刻で観測される m 本のレーザのうち j 本目で計測された値のことである．

4.4　リセットポリシー

本節では，今回設計したリセットポリシーについて述べる．リセットポリシーの機能は，ロボット周辺の非占有領域の面積が大きい方向へと進むことである．そこで，**図 4**

Vol.2022-DPS-191 No.44
Vol.2022-MBL-103 No.44
Vol.2022-ITS-89 No.44
2022/5/27

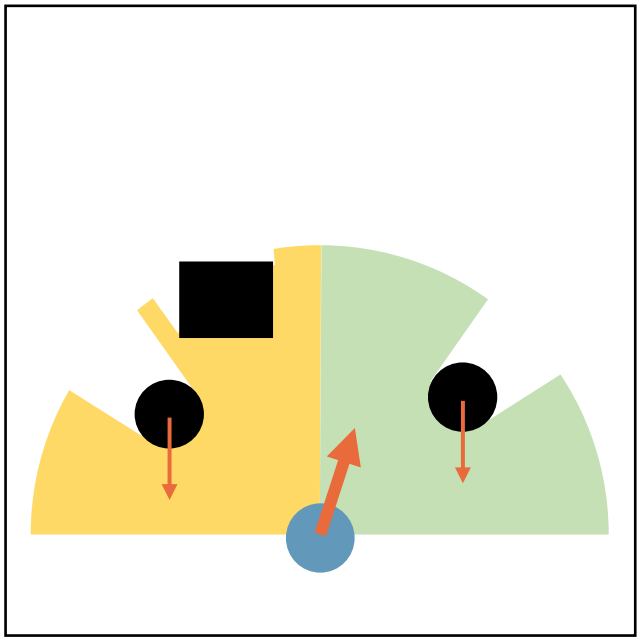


図 4: リセットポリシーによる行動決定

のように，センサの範囲を範囲中央から左右に分割し，左右それぞれにおける非占有面積 s_l，s_r を式 (7) によって算出する．これらを用いて，ロボットの行動 a^t を

$$a^t = \begin{cases} (V_{reset}, -W_{reset}) & \text{if } s_r > s_l \\ (V_{reset}, W_{reset}) & \text{otherwise} \end{cases} \tag{8}$$

のように決定する．これにより，ロボットは非占有領域の面積が大きい方向へと進むことを可能にし，振動や停止状態を脱することを目指す．

5. 実験

今回，提案手法のナビゲーション性能を検証するため，シミュレータ環境にて実験を行った．本章では，実験方法および実験結果について述べる．

5.1 環境

本実験では，ロボット用ソフトウェアプラットフォームである ROS[*1]および 2 次元ロボットシミュレータである Stage[*2]を用いる．本研究は歩行者が存在する動的環境を対象としているが，今回は移動障害物をすべてロボットとし，複数のロボットと静止障害物をシミュレータ上の環境に配置して実験環境を構築し，ナビゲーション実験を行う．実験環境内の全てのロボットは，2D-LiDAR を搭載し，同一のナビゲーション手法によってそれぞれのスタート地点からゴール地点まで移動するものとする．センサの範囲は 180 deg，6.0 m とする．各ロボットは自分以外のロボットと位置等の情報を共有しないため，ロボットを他の障害物と明示的に区別することはない．そのため，ロボットをロボット以外の移動物体（人間等）と置き換えることができ，マルチロボット環境に限らず，より一般的な動的環境における衝突回避ナビゲーション問題に応用可能であると考えられる．ロボットが他のロボットや障害物と衝突した場合は，ロボットはその場で停止し，障害物として残存するものとする．

[*1] http://wiki.ros.org
[*2] http://wiki.ros.org/stage

5.2 指標

ナビゲーション性能を表す指標として，

成功率 全試行中，1 度も衝突せずに時間内にゴールへ辿りついたロボットの割合．

衝突率 全試行中，他のロボットや静止障害物に衝突して停止したロボットの割合．

タイムアウト率 全試行中，衝突しないものの時間内にゴールできなかったロボットの割合．

平均移動時間 1 度も衝突せず時間内にゴールしたロボットがゴール到達までに要した時間の平均．

平均速度 1 度も衝突せず時間内にゴールしたロボットの速度の平均．

を用いる．なお，今回は環境中にロボットが複数存在するが，各指標は環境中の全てのロボットのデータを用いて算出するものとする．タイムアウトとする制限時間は，実験環境の大きさや実際にナビゲーションを行った様子から，100 s と設定した．

5.3 比較手法

本実験では，提案手法のナビゲーション性能を従来手法と比較する．比較手法は

DRL 単体 3.1 節で説明した Long らによる衝突回避 DRL ポリシー [6]．

DSE DRL ポリシー，安全ポリシー，効率ポリシーの 3 種類を切り替える．

DSER 提案手法．DRL ポリシー，安全ポリシー，効率ポリシー，リセットポリシーの 4 種類を切り替える．

の 3 つとする．

DSE は，アルゴリズム 1 から 5，6 行目の Potencial risk scenario の判定を除いたフローでサブポリシー選択を行う．これは，判定に障害物との最小距離を用いる点，および DRL ポリシーと安全・効率性を高めるポリシーを組み合わせるという点において，Fan らのハイブリッドアプローチ [3] と同様である．

安全ポリシーは，Fan らのハイブリッドアプローチにおける Safe DRL ポリシーと同様に，DRL ポリシーの入力と出力を制限したものを用いる．入力に関しては，パラメータ P_{scale} によって，センサによる観測値 o_z^t を $\hat{o}_z^t = \frac{o_z^t}{p_{scale}}$ のようにスケーリングし，出力される速度および角速度は $[-V_{max}, V_{max}]$ の範囲にクリッピングする．また，今回の実装における効率ポリシーは，アルゴリズム 2 によって制

Vol.2022-DPS-191 No.44
Vol.2022-MBL-103 No.44
Vol.2022-ITS-89 No.44
2022/5/27

Algorithm 2 効率ポリシーの実装

$v = v^{t-1} + 0.1$
if $v > 1.0$ **then**
  $v \leftarrow 1.0$
end if
$\theta = \arctan\left(\frac{g_x}{g_y}\right)$
if $\theta < -0.2$ **then**
  $a \leftarrow (v, -0.5)$
else if $\theta > 0.2$ **then**
  $a \leftarrow (v, 0.5)$
else
  $a \leftarrow (v, 0.0)$
end if

表 2: パラメータの設定

V_{max}	0.5
V_{reset}	0.1
W_{reset}	0.5
P_{scale}	1.05
R_{safe}	4.0
R_{risk}	1.0
S_{risk}	18.0

御司令 $a^t = (v^t, w^t)$ を決定するものとする．

各パラメータの値を**表 2** に示す．これらは，予めナビゲーションのテストを行った結果に基づいて決定した．

5.4 実験シナリオ

本節では，ナビゲーション実験を行うシナリオについて説明する．今回，提案手法のナビゲーション性能を検証するため，次の 3 通りのシナリオにて実験を行った．

円形 直径 20 m の円形内にロボット 12 台と静止障害物 9 個配置したシナリオ．各ロボットのゴールは，初期位置から円の中心を通った反対側である．シナリオの様子を**図 5a** に示す．

通路 1 通路幅 6 m の階段状の通路にロボットを 8 台配置したシナリオ．各ロボットの初期位置とゴール位置は環境中心に対して対称であり，スタートからゴールまでの直線距離は全ロボットで同じである．シナリオの様子を**図 5b** に示す．

通路 2 最大通路幅 6 m，最小通路幅 3 m の通路にロボットを 8 台配置したシナリオ．各ロボットの初期位置とゴール位置は環境中心に対して対称であり，スタートからゴールまでの直線距離は全ロボットで同じである．シナリオの様子を**図 5c** に示す．

各シナリオは，従来手法ではロボットの停止や振動が発生しやすいと考えられる複数の移動障害物と静止障害物が混在する環境や狭い環境を模したものである．円形のシナリオでは，大きな部屋の中に障害物が点在する場合のナビゲーション性能を検証する．2 通りの通路のシナリオは，ロボットの移動可能範囲や視界が制限されるような障害物で囲まれた状況を表している．特に，通路 2 は幅が狭く，ナビゲーションがより困難な状況を想定している．

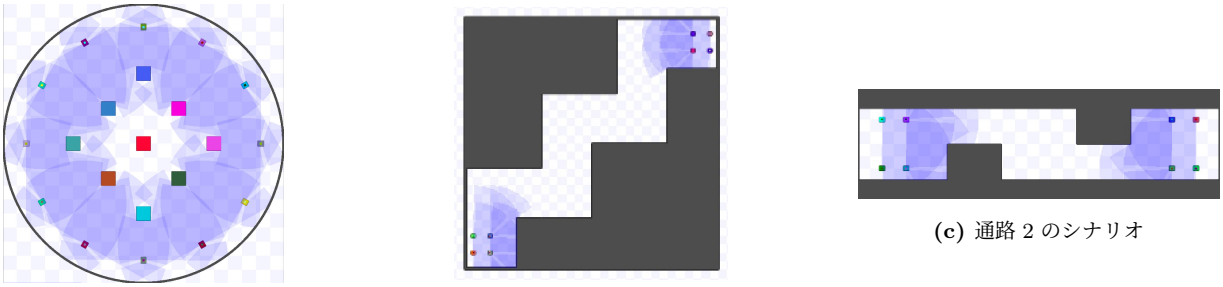
表 3: ナビゲーション性能の比較

指標	手法	シナリオ		
		円形	通路 1	通路 2
成功率	DRL	0.84	0.64	0.58
	DSE	0.96	0.72	0.50
	DSER	0.98	0.76	0.76
衝突率	DRL	0.16	0.34	0.38
	DSE	0.04	0.26	0.46
	DSER	0.02	0.24	0.23
タイムアウト率	DRL	0.0	0.021	0.046
	DSE	0.0	0.017	0.041
	DSER	0.0	0.0	0.0083
平均移動時間	DRL	21.58	40.34	31.53
	DSE	27.44	50.34	51.33
	DSER	27.42	51.21	54.61
平均速度	DRL	0.90	0.99	0.93
	DSE	0.68	0.78	0.55
	DSER	0.69	0.77	0.52

5.5 結果

各手法にて 30 回ずつ試行した結果を**表 3** に示す．円形のシナリオでは，前回の実験でも示されていた通り [4]，切替アプローチである DSE および DSER の成功率は DRL ポリシー単体の場合と比較して高くなった．なお，このシナリオにおいて，DSER のリセットポリシーが選択される条件が満たされることはなかったため，実質的に DSE と DSER では同等のナビゲーションが行われている．通路 1，通路 2 のシナリオでは，提案手法である DSER の成功率は最も高く，衝突率は最も低くなっており，周囲を壁に囲まれた狭い環境において安全性が向上している．時間内にゴールへ辿り着けない割合であるタイムアウト率も DSER が最も低くなっており，ロボットが身動きをとれない状況を判別してその状況を脱出するという提案手法における目的が達成できていると考えられる．これは，非占有領域の面積を用いてリセットポリシーへの切替を行ったことにより，狭い空間を認識した慎重な行動や前方が塞がれた場合の方向転換が可能になったためだと考えられる．

成功率等の安全面における性能が提案手法によって向上した一方で，平均移動時間や平均速度といった効率面の指標のみを見ると，DRL が最も良い結果になっている．ただし，平均移動時間および平均速度は，ナビゲーションが成功した試行における平均である．DRL ポリシー単体の成功率は全体的に他手法よりも低いことから，DRL ポリシーは比較的高い速度を維持することが可能であるが，その結果衝突に至っている可能性がある．人に近い距離で活



(a) 円形のシナリオ　(b) 通路 1 のシナリオ　(c) 通路 2 のシナリオ

図 5: 実験シナリオの様子．ロボットは中心に点（センサ）がある四角形の物体として表される．薄い青い部分はロボットのセンサ範囲を表している．濃いグレー部分およびロボット以外の図形は壁や障害物である．

動するロボットを想定するため，効率性よりも安全性の確保が優先であり，成功率が同等でない場合において効率面の指標を比較する意義は薄いと考える．

通路 1 における DSE と DSER の成功率は比較的近くなっている．リセットポリシーの目的の一つは，安全ポリシーが必要となる状況を回避して移動速度の低下を防ぐことであるが，この場合の平均移動時間および平均速度にほとんど差は見られない．そのため，この目的達成のためにはリセットポリシーの改善またはパラメータ V_{reset}，W_{reset} の調整が必要だと考える．

また，通路のような狭い環境における成功率は向上したものの，76%程度にとどまっている．この原因として，安全ポリシーによる速度制限が不十分であることや環境ごとにパラメータをチューニングする必要があること等が考えられる．

6. おわりに

本稿では，我々が提案するロボット周辺の非占有面積を用いたシナリオ判定方法とリセットポリシーを導入した複数ポリシー切り替えアプローチによるナビゲーション手法 [4] の性能の検証を行った．シミュレータによる実験では，提案手法を DRL ポリシー単体およびサブポリシー 3 種の切替手法と比較し，通路のような狭い環境において，成功率が向上することを示した．今後，シナリオ判定フローやリセットポリシー改善等を行い，更なる性能向上を目指す．また，今回の実験では環境内のロボットが全て同一のナビゲーション手法に従って移動するものとしたが，今後その制約を無くし，より実世界に近い動的環境におけるナビゲーション性能の検証を行いたいと考えている．

謝辞 本研究の一部は，JSPS 科研費 20K11776，20K12011 の助成を受けたものである．

参考文献

[1] Trautman, P. and Krause, A.: Unfreezing the robot: Navigation in dense, interacting crowds, *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, pp. 797–803 (2010).

[2] Sathyamoorthy, A. J., Patel, U., Guan, T. and Manocha, D.: Frozone: Freezing-free, pedestrian-friendly navigation in human crowds, *IEEE Robotics and Automation Letters*, Vol. 5, No. 3, pp. 4352–4359 (2020).

[3] Fan, T., Long, P., Liu, W. and Pan, J.: Fully distributed multi-robot collision avoidance via deep reinforcement learning for safe and efficient navigation in complex scenarios, *arXiv preprint arXiv:1808.03841* (2018).

[4] 天野加奈子，加藤由花：複数ポリシー切替による複雑な環境に適用可能な自律移動ロボットナビゲーション手法，研究報告マルチメディア通信と分散処理（DPS）2022-DPS-190(27)，pp. 1–7 (2022).

[5] Chen, C., Liu, Y., Kreiss, S. and Alahi, A.: Crowd-robot interaction: Crowd-aware robot navigation with attention-based deep reinforcement learning, *2019 International Conference on Robotics and Automation (ICRA)*, IEEE, pp. 6015–6022 (2019).

[6] Long, P., Fan, T., Liao, X., Liu, W., Zhang, H. and Pan, J.: Towards optimally decentralized multi-robot collision avoidance via deep reinforcement learning, *2018 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, pp. 6252–6259 (2018).

[7] Cui, Y., Zhang, H., Wang, Y. and Xiong, R.: Learning world transition model for socially aware robot navigation, *2021 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, pp. 9262–9268 (2021).

[8] Rusu, A. A., Večerík, M., Rothörl, T., Heess, N., Pascanu, R. and Hadsell, R.: Sim-to-Real Robot Learning from Pixels with Progressive Nets, *Proceedings of the 1st Annual Conference on Robot Learning* (Levine, S., Vanhoucke, V. and Goldberg, K., eds.), Proceedings of Machine Learning Research, Vol. 78, PMLR, pp. 262–270 (2017).

[9] Sadeghi, F. and Levine, S.: Cad2rl: Real single-image flight without a single real image, *arXiv preprint arXiv:1611.04201* (2016).

[10] Schulman, J., Wolski, F., Dhariwal, P., Radford, A. and Klimov, O.: Proximal policy optimization algorithms, *arXiv preprint arXiv:1707.06347* (2017).