強化学習による差動二輪車制御における未知実証環境での安全推論

門垣　幸樹†　　高井　勇志††　　宮口　幹太††　　北野　信吾††　　大島　裕明†

† 兵庫県立大学 情報科学研究科　〒 651–2197 兵庫県神戸市西区学園西町 8-2-1
†† 株式会社竹中工務店 技術研究所　〒 270–1395 千葉県印西市大塚 1-5-1
E-mail: †mo.0709.uoh@gmail.com, ††{takai.takeshi, miyaguchi.mikita, kitano.shingo}@takenaka.co.jp, †ohshima@ai.u-hyogo.ac.jp

あらまし　本研究では，強化学習を用いて，現地学習を必要とせずに，未知実証環境で差動二輪車を安全に制御する手法を提案する．本研究における未知実証環境とは，学習環境とは異なる環境条件や予期せぬ外乱が発生する環境を指す．学習時に経験しなかった外乱が存在する環境では，ロボットの動作が不安定化し，軌道逸脱のような危険が発生する可能性がある．本研究では，タスク遂行における危険を軌道逸脱と定義し，将来の軌道逸脱を予測する危険判別器を構築する．そして，タスクの効率的達成に特化した最適方策と，安全状態への復帰に特化した安全方策を，危険判別器が推論した危険度に基づいて動的に切り替える安全推論制御を提案する．実験では，車輪の不調や路面摩擦の変化等を再現した 6 種類の未知実証環境を用いて，従来手法と提案手法の安全性と効率性を比較評価した．実験の結果，重度の車輪故障環境において，提案手法は経路誤差を抑制し，それに伴い成功率も従来手法の 79.0%から 93.2%へと向上した．本手法により，現地学習を必要とせずに，未知実証環境下でのタスク遂行の安全性を向上させる可能性が示された．

キーワード　強化学習，車体制御，未知実証環境

1　はじめに

近年，月面や災害現場といった，人間が直接調査することが困難または危険な未知実証環境において，自律型探査ロボットの活用が期待されている [1], [2], [3]．これらのロボットの制御方策は，事前に構築された学習環境で獲得されることが多い．しかし，未知実証環境について事前に情報を得ることは困難であるため，全く同じ環境条件の学習環境を設計することはできない．そのため，未知実証環境で予期せぬ外乱や環境条件に遭遇した際に，学習環境に最適化された方策は行動が不安定化し，危険な行動を引き起こす可能性がある．一度の致命的な失敗がタスクの成否を左右するような状況では，制御方策にはタスク達成能力だけでなく，高い安全性が求められる．

これまでの研究では，未知環境に遭遇した後にオンラインで追加学習を行い，方策を環境に適応させる手法が提案されている [4], [5]．この手法は環境変化に対して柔軟に適応できる利点がある一方で，現地での学習には多大な計算コストや時間が必要となるため，エネルギーや計算資源が限られる過酷な環境では適用が難しい．また，安全性を考慮した研究として，安全制約を導入することで危険な行動を抑制する安全強化学習があるが，この手法は既知の安全領域を前提としており，学習時に想定されていない未知の危険領域に対しては十分に対応できないという問題がある．

そこで本研究では，強化学習による未知実証環境での差動二輪車制御において，現地での学習を必要としない安全推論制御手法を提案する．本手法を用いた制御概要図を図 1 に示す．本手法は，タスクの効率的な達成に特化するよう事前学習した最適方策と，危険状態からの復帰に特化するように事前学習した安全方策という，役割の異なる 2 つの方策を用いる．本研究では，危険として軌道逸脱を扱い，将来の軌道逸脱を予測する危険判別器を構築する．そして，未知実証環境での制御時に危険判別器が推論した危険度に応じてこれらの方策を動的に切り替えることで，効率性と安全性を両立させた制御手法の実現に取り組む．

2　関連研究

安全性を考慮した強化学習制御や未知環境での強化学習制御に関する様々な研究が行われている．

2.1　安全強化学習

安全強化学習は，期待報酬の最大化だけでなく，事前に定義された安全制約を満たす方策を学習することを目的とする．代表的なアプローチとして，制約付きマルコフ決定過程に基づく手法が挙げられる．Achiam ら [6] が提案した Constrained Policy Optimization（CPO）は，信頼領域法を拡張し，期待報酬を向上させつつ，安全制約違反の期待値を所定の閾値以下に抑えるように方策を更新する手法である．これにより，学習プロセス全体を通じて安全性を維持することが可能となる．安全強化学習に関する包括的な調査 [7] においても，制約付き最適化や安全層の導入が主要なアプローチとして挙げられている．

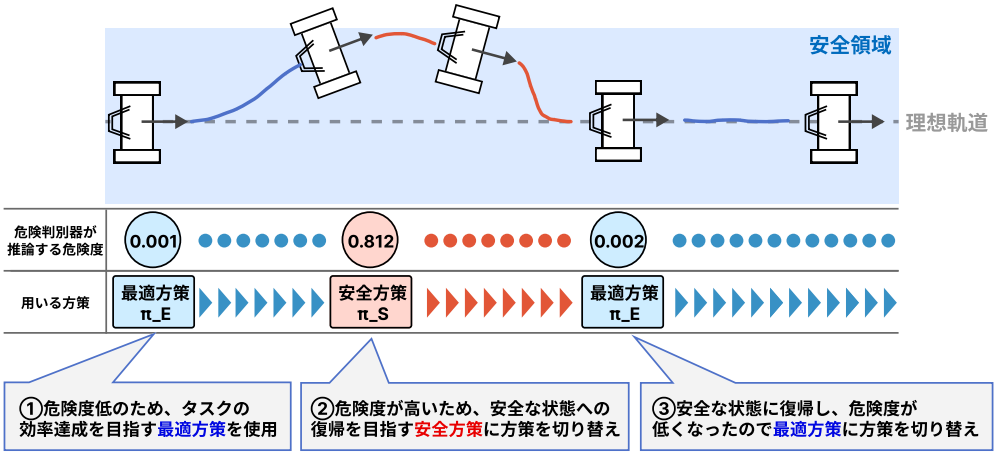

図 1　軌道逸脱を危険とした場合の本手法による制御の概要

また，タスク遂行と安全確保の役割を分離するアプローチも提案されている．Thananjeyan ら [8] による Recovery RL は，タスク方策とは別に，危険な状態からの回復に特化した回復方策を学習する．この手法では，安全判定器を用いて将来の制約違反確率を予測し，危険度が高いと判断された場合に制御権を回復方策に移すことで，探索中の安全性を高めている．

2.2　メタ学習による未知環境でのオンライン適応

未知環境への適応手法として，メタ学習が注目されている．メタ学習は学習する方法を学習する枠組みであり，新たなタスクや環境に対して，少数のデータを用いて迅速に適応することを目的とする．代表的な手法である Model-Agnostic Meta-Learning（MAML）[9] は，わずかな勾配更新で新しいタスクに適応可能な初期パラメータを学習する手法である．

ロボティクスの分野においても，シミュレーションと実環境のギャップを埋めるためにメタ学習が応用されている．Arndt ら [4] は，MAML をシミュレーション等で学習したモデルを現実世界に転移させて制御する Sim-to-Real に適用し，シミュレーションで多様な物理パラメータを経験させることで，実機上の未知の力学特性に対して数回の更新で適応可能な方策を獲得できることを示した．また，Nagabandi ら [5] は，モデルベース強化学習とメタ学習を組み合わせることで，4 脚ロボットが故障や環境変化に対してオンラインで適応する手法を提案している．Wang ら [10] は，メタ学習の過程で軌道最適化手法である iLQR [11] を用いて自己生成した教師データを利用する手法を提案している．人間のデモンストレーションを用いることなく訓練時のサンプル効率を大幅に改善し，結果として未知の操作タスクに対しても高い適応性能が得られることを示している．

しかし，これらの勾配ベースのメタ学習手法は，実環境においてエージェントがデータを収集し，その場で勾配計算とパラメータ更新を行う必要がある．Arndt ら [4] の研究でも示されているように，適応のためには実機データの収集とバックプロパゲーションが不可欠であり，計算リソースやメモリが制限された探査ロボット等においては，計算コストが大きな課題となる．また，オンライン学習中の挙動の安定性を保証することも困難である．対して本研究では，オンラインでの勾配更新を行わず，推論のみで完結する動的な方策切り替えを用いるため，計算コストを抑えつつ未知環境へ適応可能である点で優位性があると考える．

2.3　方策切り替えによる適応制御

単一の方策ではなく，状況に応じて複数の方策や制御器を動的に切り替えることで，未知環境や不確実性に対応するアプローチが提案されている．代表的なものとして，状況の不確実性に応じて行動を調整する手法や，危険な行動を事前に検知して介入する手法がある．

不確実性に応じた行動調整として，Kahn ら [12] は，衝突確率の不確実性をブートストラップ法 [13] により推定し，不確実性が高い場合にはロボットの速度を低下させる手法を提案している．同様に，Tran ら [14] は，複雑な環境における自律航行のために，探索と活用の方策を動的に切り替える協調的深層強化学習フレームワークを提案している．また，Chemingui ら [15] は，オフライン強化学習において，安全性制約に応じて複数の方策を適応的に切り替える手法の有効性を示している．

一方，危険な行動への介入による安全確保として，方策が選択しようとする行動を事前にチェックし，危険と判断された場合に別の安全な行動へ強制的に変更するシールドを用いた手法が提案されている．前述の不確実性に応じた行動調整が速度低下などの緩やかな対応を行うのに対し，シールド手法は危険な行動そのものを遮断するより直接的な介入を行う．Alshiekh ら [16] は，エージェントが将来的に危険な状態に陥る可能性のある行動を選択しようとした際に，即座に介入して安全な行動へと切り替える手法を提案している．この手法により，学習の試行錯誤の最中であっても事故の発生を防ぐことができる．Yang ら [17] は確率モデルに基づくシールド手法を提案し，安全制約を方策学習に組み込むことで安全性を向上させている．

本研究では，軌道逸脱における予測した危険度を用いて，最適方策と安全方策を推論時に切り替えることで，学習されていない未知の外乱に対しても安全な挙動を実現する．上述のシー

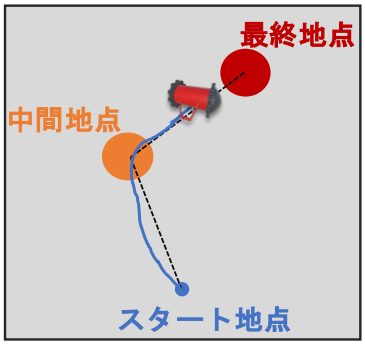

図 2　制御タスクと理想の行動

ルド手法が高度な論理推論や適応機構を用いるのに対し，本研究は計算リソースの限られたロボットにおいて，推論のみで完結する軽量な手法により，制御の効率性と安全性の両立を目指す．

3　問題定義とシミュレーション環境

本節では，本研究が対象とする制御タスクと，実験に用いるシミュレーション環境について述べる．

3.1　問題定義

本研究では，学習環境とは異なる環境条件を持つ未知実証環境での安全性を考慮した制御問題を扱う．本研究で扱う制御タスクを図 2 に示す．制御対象として差動二輪車を用いる．差動二輪車は，左右の車輪を独立に制御することで前後左右に動作可能な車両型ロボットである．制御タスクとしては，指定されたスタート地点から中間地点を経由し，最終地点に到達後，一定時間その場に留まる動作とする．このタスクにおける理想的な制御とは，各地点間を結ぶ直線経路からの逸脱を最小限に抑え，素早く効率的にタスクを達成することである．

本研究で想定する未知実証環境とは，学習時には経験しなかった環境条件や外乱が発生する環境と定義する．具体的には，車輪への異物の巻き込みやシャフトの不具合によって片輪の回転が重くなる等の車輪の出力異常や，路面摩擦の変化などが挙げられる．このような環境下で，学習環境でのみ学習を行なったモデルを用いて制御を行なった場合，直進しようとしても車体が意図せず旋回してしまうといった不安定な挙動を引き起こす．その結果，本来辿るべき軌道から大きく逸脱し，障害物への衝突や車体の転倒など，タスク継続が困難な危険な状態に陥る可能性が高い．

本研究における危険な状態とは，タスクの失敗に直結，あるいはその可能性を高める状態と定義し，タスク遂行における危険として軌道逸脱を用いる．

本研究の目的は，上述した未知実証環境において，エージェントが危険な状態に陥ることを防ぎ，あるいは陥った際に速やかに安全な状態へ復帰させ，制御タスクを効率よく安全に達成する制御方法を実現することである．

3.2　MDP の定式化

本研究では，上述の制御問題を観測空間 $\mathcal{O}$，行動空間 $\mathcal{A}$，遷移確率 $\mathcal{P}$，報酬関数 $\mathcal{R}$ からなるマルコフ決定過程（MDP）として定式化する．エージェントは各ステップ t において，環境からの観測 $o_t \in \mathcal{O}$ に基づき行動 $a_t \in \mathcal{A}$ を決定し，その結果として次の観測 o_{t+1} と報酬 r_t を受け取る．

本研究で対象とする差動二輪車の行動空間 $\mathcal{A}$ は，左右の車輪への出力値からなる 2 次元の連続空間として定義される．

$$\mathcal{A} = \{(a^{(left)}, a^{(right)}) \mid -1.0 \leqq a^{(left)}, a^{(right)} \leqq 1.0\} \tag{1}$$

ここで，1.0 は最大速度での前進，-1.0 は最大速度での後退を意味する．

観測空間 $\mathcal{O}$ および報酬関数 $\mathcal{R}$ の具体的な設計については，最適方策と安全方策の必要な情報が異なるため，詳細な定義は第 4 節にてそれぞれ述べる．

3.3　シミュレーション環境

本研究では，方策の学習と評価を行うためのシミュレーション環境として Unity ML-Agents [1]を使用する．制御対象は，本体と左右の車輪から構成される差動二輪車である．車体には，外部環境および自身の状態を認識するためのセンサとして，3 つの機能が搭載されている．1 つ目は IMU（慣性計測装置）である．IMU は車体の 3 軸加速度および角速度を計測し，自身の姿勢の不安定化の予兆を検知することができる．2 つ目は，IR センサである．これは，車体前方の左右に配置され，障害物や壁までの距離を計測することができる．3 つ目は，自己位置推定機能である．これは理想軌道に対する相対位置や，目標地点までの距離を正確に取得できる機能である．これらのセンサ情報に基づき，エージェントは環境の状態を観測する．

本研究における学習環境と評価環境の設計を図 3 に示す．学習環境では，Domain Randomization（DR）を用いて路面の凹凸，車輪の不調，および地面摩擦をエピソードごとにランダムに変動させる．これらのパラメータはエピソード開始時に決定され，エピソード中は一定である．路面の凹凸は，地形の高さをフラクタルノイズで生成し，振幅とスケールを変動させることで多様な起伏を表現する．車輪の不調は，エージェントが出力しようとしたトルクに対して実際に出力されるトルクが減少する状態を表す．学習環境では出力しようとしたトルクに 0.7 ～ 1.0 の範囲で変動する出力係数を乗じることで，車輪の不調を模擬する．地面摩擦係数は 0.3 ～ 1.0 の範囲で変動させる．これにより，エージェントは多様な環境条件を経験し，未知環境に対するロバスト性を獲得する．

評価環境である未知実証環境には，学習環境とは地形設計そのものが異なる富士麓の洞窟環境を使用する．この環境上で，以下の 2 種類の条件を設定し評価を行う．

OOD 条件　学習時に経験した外乱や環境条件の変動範囲から逸脱した条件である．車輪の不調では出力係数を 0.3 ～ 0.7

1：https://github.com/Unity-Technologies/ml-agents

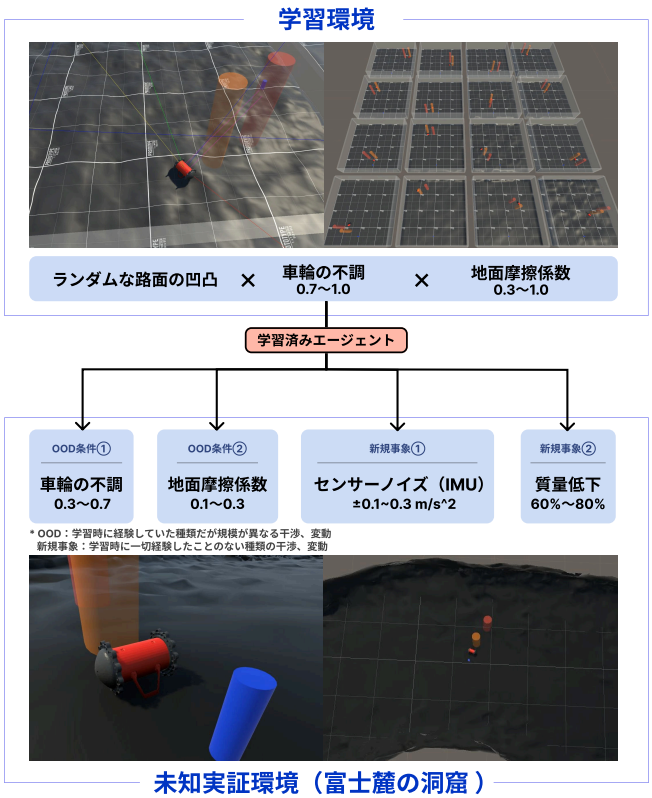

図 3　学習環境と未知実証環境の設計概要.

とし，学習時より深刻な故障を模擬する．地面摩擦係数では 0.1 ～ 0.3 とし，学習時より滑りやすい路面を設定する．

新規事象　学習時には一切経験していない種類の干渉である．センサノイズでは，IMU の加速度に $\pm 0.1 \sim 0.3$ m/s^2 の定常バイアスを付加し，センサ異常を模擬する．車体の欠損では，部品欠損等による質量低下を想定して，車体質量を 60% ～ 80% に低下させる．

4　方策切り替えを用いた安全推論制御

本節では，未知実証環境において現地学習を必要とせずに安全性とタスク遂行効率を両立する安全推論制御手法について述べる．本手法は，第 4.1 節と第 4.2 節で述べる役割の異なる 2 つの事前学習済み方策を，第 4.3 節で述べる危険判別器が予測する危険度に応じて動的に切り替える．これにより，安全性の高い状況では効率的な制御を，危険性の高い状況では安全性を優先した制御を実現する．

4.1　最適方策

最適方策は，タスクを素早く効率的に達成することを目指して，制御を行うように学習される方策である．

4.1.1　学習環境とカリキュラム学習

最適方策の学習アルゴリズムには，Proximal Policy Optimization（PPO）[18] を採用した．未知実証環境に対するロバスト性を獲得するため，Domain Randomization（DR）を用いて，物理パラメータがエピソードごとにランダムに変動する環境で学習を行なった．DR はシミュレーション環境において摩擦係数や車輪の不調，路面の凹凸などの環境パラメータに意図的なばらつきを導入することで，多様な環境条件に対するロバストな制御方策の獲得を目指す手法である．近年の研究では，DR が Sim-to-Real 問題の解決に有効であることが示されている [19], [20], [21]．また，著者らの先行研究 [22] において，本研究と同様の差動二輪車制御タスクにおいて DR が未知実証環境への適応に有効であることを確認している．本研究では，この結果を踏まえ，DR で学習した方策をベースラインとして採用し，その上で危険判別器による動的な方策切り替えを導入することで，さらなる安全性の向上を目指す．なお，多様な環境条件を一度に学習することは困難であるため，3 段階のカリキュラム学習を導入し，段階的に環境の変動範囲を拡大することで安定した学習と高い汎化性能の獲得を図った．

表 1　最適方策と安全方策の観測空間一覧

方策	観測項目	次元	内容概要
最適	中間目標情報	4	相対位置，ヨー角誤差，距離
	最終目標情報	4	相対位置，ヨー角誤差，距離
	到達フラグ	1	中間地点への到達状態
	姿勢情報	2	車体のピッチ角およびロール角
	IMU 情報	3	車体座標系における 3 軸線形加速度
	IR センサ	2	左右のセンサによる障害物までの距離
	角速度	1	Y 軸周りの角速度
安全	直近目標情報	4	直近目標への相対位置，ヨー角誤差，距離
	復帰基準情報	2	復帰軌道始点からの変位
	制御誤差情報	3	クロストラックエラー，方位角誤差など
	姿勢情報	2	車体のピッチ角およびロール角
	IMU 情報	3	車体座標系における 3 軸線形加速度
	IR センサ	2	左右のセンサによる障害物までの距離

4.1.2　観測空間

最適方策の観測空間は，タスクの進行状況と自己の状態を把握するための計 17 次元のベクトルで構成される．観測空間の詳細を表 1 に示す．タスク達成に直接関わる中間地点や最終地点に関する情報のほか，姿勢安定性の監視のための IMU 情報や，障害物回避のための IR センサ情報が含まれている．

強化学習モデルへ入力される情報としては，現在の観測情報 o_t と 1 ステップ前の観測情報 o_{t-1}，および 1 ステップ前の行動情報 a_{t-1} をスタックした，計 36 次元のベクトル o_{t-1}, o_t, a_{t-1} を用いる．

4.1.3　報酬設計

最適方策の報酬関数は，効率的なゴール到達を促しつつ，無駄な時間経過や危険な挙動を抑制するように設計した．各ステップの報酬 r_t は，以下の式で計算される．

$$r_t = r_{progress} + r_{time} + r_{posture} + r_{event} \tag{2}$$

進捗報酬 $r_{progress}$ は，車体が目標方向を向いている場合に，目標への接近距離 Δd に係数 200 を掛けた値を付与する．これは「目標に 1m 近づくと報酬 200 を得る」ことを意味し，後退による接近は評価しない．時間ペナルティ $r_{time} = -0.1$ は

表 2　最適方策と安全方策の報酬設計一覧

方策	報酬項	値/係数	説明
最適	進捗報酬	200.0	目標接近距離に乗算
	中間地点到達	+50	中間地点到達時
	最終地点到達	+100	停止条件達成時
	時間ペナルティ	−0.1	毎ステップ
	姿勢ペナルティ	−0.1	車体の傾きに応じて
	逸脱ペナルティ	−1.0	到達時に蓄積値を減算
安全	復帰進捗	1000.0	軌道接近距離に乗算
	復帰成功	+5.0	安全領域復帰時
	方向ペナルティ	−0.1	軌道付近での角度誤差
	姿勢ペナルティ	−0.1	車体の傾きに応じて
	時間ペナルティ	−0.01	毎ステップ

毎ステップ付与され，無駄な停滞を抑制する．姿勢ペナルティ $r_{posture}$ は，車体の傾きに応じて最大 -0.1 を付与し，安定走行を促す．

イベント報酬 r_{event} として，中間地点到達時に+50，最終地点での停止成功時に+100 を付与する．また，理想軌道からの逸脱については，走行中の逸脱距離を蓄積しておき，目標地点に到達した時点で係数 -1.0 を掛けてペナルティとして減算する．これにより，学習の安定性を保ちつつ直線的な走行を促進する．各報酬項の具体的な値を表 2 に示す．

4.2 安全方策

安全方策は，理想軌道から逸脱した危険な状態から，軌道上へ復帰することに特化した方策である．本研究では，タスク遂行における危険として軌道逸脱を対象とし，安全方策は軌道復帰に特化した設計とする．

4.2.1 学習環境と目的

安全方策の学習には，カリキュラム学習的なアプローチを用いる．エピソード開始時に，エージェントを意図的に軌道から大きく外れた位置や不安定な姿勢に配置し，そこから短時間で安全領域へ復帰することをタスクとする．また，最適方策と同様に DR を適用し，摩擦係数や車輪出力などの物理パラメータをランダムに変動させた環境で学習を行う．これにより，未知実証環境においても安定した復帰制御が可能な方策の獲得を目指す．

4.2.2 観測空間

安全方策の観測空間は，復帰制御に必要な情報に特化した 16 次元のベクトルと，過去 1 フレーム分の履歴で構成される．観測情報の詳細を表 1 に示す．最適方策と比較して，クロストラックエラーや復帰方向誤差など，現在の軌道からの逸脱状況を示す情報が含まれている．

強化学習モデルへ入力される情報としては，最適方策と同様に，現在の観測情報 o_t と 1 ステップ前の観測情報 o_{t-1}，および 1 ステップ前の行動情報 a_{t-1} をスタックした，計 36 次元のベクトルを用いる．

4.2.3 報酬設計

安全方策の目的は，速やかに軌道上かつ目標方向を向いた安全な状態へ復帰することである．そのため，復帰行動の進捗を評価する報酬設計を行なった．各ステップの報酬 r_t は，エージェントの状態に応じて以下のように計算される．

$$r_t = \begin{cases} w_{rec} \cdot \Delta d_{err} & \text{(復帰中)} \\ -w_{align} \cdot |\theta_{err}| & \text{(軌道付近)} \end{cases} + r_{common} \tag{3}$$

復帰中は，軌道への復帰方向を向いている場合に限り，クロストラックエラーの減少量 Δd_{err} に比例した正の報酬を与える．これにより最短経路での復帰を促す．軌道付近に到達した後は，方位角誤差 θ_{err} に対するペナルティを与え，軌道方向への整列を促す．また，共通項 r_{common} として，姿勢安定化や時間経過に対するペナルティ，および復帰成功時の報酬を含む．具体的なパラメータを表 2 に示す．

4.3 方策切り替えのための危険判別器

本節では，最適方策から安全方策への切り替え判断を担う危険判別器について述べる．未知実証環境における予期せぬ悪路や外乱に対して，効率性を重視する最適方策は行動が不安定化する可能性がある．そこで，観測情報から車体の状態を監視し，危険な状態に陥る予兆を検知するモデルを構築する．本研究では，タスク遂行における危険として軌道逸脱を対象とし，将来の逸脱発生度を予測する．

危険判別器の設計において決定すべきパラメータは，逸脱判定閾値 D_{th}，予測時間 T_{pred}，入力系列長 T_{ctx}，および分類閾値 θ_{cls} の 4 つである．これらのパラメータは，以下に述べる学習データセットを用いて決定した．

4.3.1 学習データセットの構築

判別器の学習には，最適方策を用いて多様な DR 環境を走行させることで収集した約 1,070 万ステップのログデータを使用した．学習の安定化のため，チャタリング除去を行なった．危険フラグは一度発生すると短期間に 0 と 1 を繰り返す傾向があるため，各エピソードにおいて最初にフラグが立った時点のみを正例とし，それ以降のデータは学習から除外した．データを学習用とテスト用に 8 : 2 で分割し，約 420 万ステップの学習データは正例と負例の割合をアンダーサンプリングで 1 : 1 に揃えた．約 105 万ステップのテストデータは実運用時の分布での性能評価を行うため，サンプリングを行わずに元の分布のまま使用した．テストデータにおける正例の割合は約 1.7% である．以下では，このデータセットを用いた各パラメータの導出方法について述べる．

4.3.2 逸脱判定閾値の決定

逸脱判定閾値 D_{th} は，最適方策が軌道からのずれを自力で修正できる限界の距離として定義する．この閾値を超えると，最適方策だけでは軌道に戻ることが難しくなり，安全方策への切り替えが必要となる．

閾値を決定するため，収集したログデータを用いて，軌道からの距離が時間とともにどのように変化するかを分析した．具体的には，ある時点で軌道から距離 D だけ離れている場合に，次の時点でその距離が縮まる傾向にあるか，それとも広がる傾向にあるかを調べた．

表 3　分類閾値と各評価指標

分類閾値 θ_{cls}	再現率	誤警報率	適合率	F1 値
0.5	0.937	0.115	0.122	0.215
0.6	0.917	0.096	0.141	0.244
0.7	0.883	0.078	0.162	0.273
0.8	0.818	0.056	0.199	0.320

分析の結果，軌道からの距離が 0.15m 以下の場合は，最適方策が誤差を減少させる方向に制御できていた．しかし，距離が 0.16m を超えると，誤差が増大する傾向に転じることが確認された．この結果から，最適方策が自力で回復できる限界は約 0.16m であると判断し，逸脱判定閾値を $D_{th} = 0.16$m と決定した．

4.3.3　予測時間と入力系列長の決定

逸脱判定閾値 $D_{th} = 0.16$m を固定した上で，危険予測に最適な予測時間 T_{pred} と入力系列長 T_{ctx} をグリッドサーチにより決定した．

危険判別器には，時系列データからの特徴抽出に優れた Long Short-Term Memory（LSTM）[23] を採用した．1 ステップ分の入力特徴量は，第 4.1.2 節で述べた最適方策の 17 次元の観測空間にクロストラックエラーと角度誤差，左右の車輪出力を加えた，21 次元のベクトルである．モデルは隠れ層サイズ 64 の LSTM 層と，それに続く全結合層によって構成され，シグモイド関数を通して危険度 $\hat{y} \in [0, 1]$ を出力する．

評価指標には，正例が少ない不均衡データにおいて正例の検出性能を重視する PR-AUC（Precision-Recall AUC）を採用した．入力系列長 T_{ctx} を 0.0 秒から 1.8 秒，予測時間 T_{pred} を-0.4 秒から 10.0 秒の範囲で変化させ，グリッドサーチを実施した結果を図 4 に示す．

結果より，$T_{pred} = 1.2$ 秒において PR-AUC $= 0.30$ を達成し，これはランダム分類器の約 0.017 と比較して約 18 倍の性能である．$T_{pred} > 2.0$ 秒では性能が急落し，長期予測の限界が確認された．入力系列長に関しては，増加させても PR-AUC の向上は見られなかった．以上の結果から，必要な情報量を最小限に抑えつつ，将来の危険を取りこぼしなく予測できる値として $T_{pred} = 1.2$ 秒，$T_{ctx} = 0.6$ 秒を採用した．

4.3.4　分類閾値の決定

モデル出力を「危険」と「安全」に二値化する分類閾値 θ_{cls} は，再現率で評価される安全性と，誤警報率の抑制で評価される効率性のバランスを考慮して決定した．閾値を変化させたときの各指標を表 3 に示す．

$\theta_{cls} = 0.7$ において，再現率 88.3%，誤警報率 7.8%を達成した．これは危険状況の約 88%を事前検知可能であり，かつ誤警報による不要な方策切り替えを約 8%に抑制できることを意味する．$\theta_{cls} > 0.7$ では再現率が低下し，閾値 0.8 では 82%となるため，安全性を優先する観点から $\theta_{cls} = 0.7$ を採用した．

4.3.5　方策切り替え方法

推論制御時には，学習済みの危険判別器を用いて各ステップごとに危険度を算出する．判別器が出力する軌道逸脱危険度

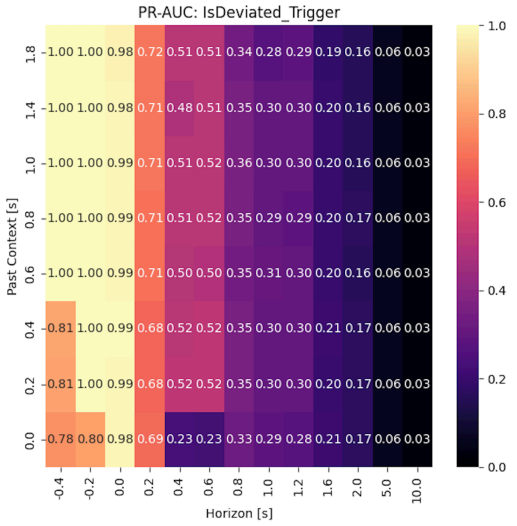

図 4　グリッドサーチによる PR-AUC ヒートマップ

P_{dev} が，事前に設定した閾値 $\theta_{danger} = 0.7$ を超えた場合，制御方策を最適方策から安全方策へと切り替える．安全方策によって車体が安定し，かつ軌道付近への復帰が完了して危険度が十分に低下した $\theta_{danger} < 0.7$ と判断された時点で，再び最適方策へと制御を戻す．これにより，危険度が低い状態では最適方策による高速な移動を行い，危険度が高い状態では，即座に安全方策による安全な状態への復帰を行う効率性と安全性を両立した制御を実現する．

5　実　　験

本節では，提案手法である安全方策と最適方策の動的切り替え制御の有効性を検証する評価実験について述べる．実験の目的は，未知の環境変動や外乱に対して，提案手法が既存の古典的制御手法や単一の強化学習方策と比較して，より高い安全性とタスク遂行能力を発揮できることを定量的に示すことである．

5.1　実 験 設 定

5.1.1　比 較 手 法

提案手法の性能を相対的に評価するため，以下の 3 つの手法を比較対象とした．

PID 制御　軌道追従のための直列 PID 制御器である．学習環境と同様の凹凸路面において，遺伝的アルゴリズム [24] を用いてゲインパラメータを事前にチューニングしたものを使用する．

最適方策　第 4.1 節で述べた DR 環境で学習された，タスクの効率達成に特化した強化学習方策である．

提案手法　本研究で提案する，将来の危険度に基づき最適方策と安全方策を動的に切り替える手法である．

5.1.2　評 価 環 境

各手法の汎化性能とロバスト性を評価するため，表 4 に示す複数の環境で実験を行なった．評価環境の基本地形には，実際の富士麓の洞窟のスキャンデータに基づいた凹凸地形を使用し，未知実証環境を再現した．Normal 環境は学習時と同様

表 4　評価環境の設定

環境名	設定内容
Normal	学習時と同様のパラメータ範囲
WheelFault	車輪出力係数 0.3 ～ 0.7
WheelFaultHard	左車輪出力係数 0.3，右車輪 1.0
LowFriction	地面摩擦係数 0.1 ～ 0.3
IMUNoise	加速度センサに ±0.1 ～ 0.3 m/s^2 の定常バイアス
MassLoss	車体の質量を 60% ～ 80% に低下

のパラメータ範囲を持つ基準環境である．WheelFault 環境と WheelFaultHard 環境は車輪の出力異常を模擬しており，学習時の出力係数範囲 0.7 ～ 1.0 より深刻な故障状態を再現している．特に WheelFaultHard 環境は左右の車輪で大きく異なる出力係数を設定し，重度の非対称故障を表現している．LowFriction 環境は，学習時の摩擦係数範囲 0.3 ～ 1.0 を下回る 0.1 ～ 0.3 を設定することで，学習時より滑りやすい路面を再現した．IMUNoise 環境と MassLoss 環境は，学習時には一切経験していない新規の外乱である．IMUNoise 環境では加速度センサに定常バイアスを付加することでセンサ異常を模擬し，MassLoss 環境では部品欠損等による質量低下を想定した設定とした．

各環境において，1 エピソードあたり最大 5,000 ステップを制限時間とし，500 エピソードの試行を行い，各手法の安全性と効率性を比較評価した．エピソードの終了条件は，タスク成功時，すなわち最終目標地点に到達し停止条件を満たした時，または制限時間到達時，すなわち 5,000 ステップ経過時のいずれかとした．

5.2　評 価 指 標

制御の安全性と効率性を多角的に評価するため，以下の 7 つの指標を用いる．

成功率　全エピソードのうち，制限時間内に最終目標地点に到達し，停止条件を満たした割合．

平均ステップ数　1 エピソードの終了までにかかった平均ステップ数．

成功時平均ステップ数　タスクに成功したエピソードのみを対象とした平均ステップ数．タスク遂行の効率性を評価する指標である．

平均総経路誤差　各エピソードにおける，理想経路からの逸脱距離の累積値の平均．この値が小さいほど，外乱に対してふらつくことなく，安全な軌道を維持できていることを示す．

成功時平均総経路誤差　タスクに成功したエピソードのみを対象とした累積経路誤差の平均．成功したエピソードにおける経路追従性を評価する．

平均最大経路誤差　各エピソードにおける，理想経路からの逸脱距離の最大値の平均．一時的に大きく逸脱したかどうかを評価する指標である．

成功時平均最大経路誤差　タスクに成功したエピソードのみを対象とした最大経路誤差の平均．成功したエピソードにおける一時的な逸脱の程度を評価する．

5.3　結果と考察

各環境における評価結果を表 5 および表 6 に示す．表 5 は全エピソードを対象とした評価結果であり，表 6 は成功エピソードのみを対象とした評価結果である．

成功率について，提案手法は 6 環境中 5 環境で最も高いスコアとなった．特に，左右の車輪出力が大きく異なる重度の非対称故障の状況を模した WheelFaultHard 環境において，提案手法の優位性が確認された．最適方策の成功率が 79.0%まで低下したのに対し，提案手法は 93.2%を達成し，14.2 ポイントの大幅な改善が確認された．これは，危険判別器が車輪故障から発生する軌道逸脱の予兆を検知し，安全方策へ切り替えることで実現されたと考えられる．

タスク達成の効率性について，成功時平均ステップ数では最適方策が多くの環境で最短を示した．一方，提案手法は WheelFaultHard 環境において成功時平均ステップ数が 2,050 と，PID の 1,559 と比較して約 31%増加した．これは安全方策への切り替えにより慎重な制御が行われるためであり，安全性と効率性の間にトレードオフが存在することを示していると考える．ただし，効率性を優先してタスク自体が失敗しては本末転倒であり，PID が 91.8%に対し提案手法は 93.2%と成功率の向上を実現している点で，この効率性の低下は許容できると考える．Normal 環境や LowFriction 環境など外乱が比較的軽微な環境では，提案手法の成功時平均ステップ数は最適方策と同等である．これは困難な環境においては安全性を考慮し，比較的容易な環境においては効率性を重視する提案手法の特性を表していると考えられる．

経路追従の安全性について，平均総経路誤差と平均最大経路誤差から考察する．WheelFault 環境において，PID 制御は平均総経路誤差 29.70 と最適方策の 6.39，提案手法の 5.475 と比較して著しく大きな値を示した．定性的な観察により，PID 制御は車輪故障という想定外の状況に適応できず，車体が転倒して大きな軌道逸脱を引き起こすケースが確認された．PID 制御では軌道追従と転倒抑制を同時に考慮した制御設計が困難であるのに対し，強化学習では報酬関数に転倒ペナルティを組み込むことで，これらを統合的に学習できる．提案手法は最適方策よりも低い経路誤差を達成しており，危険予測に基づく事前の方策切り替えが転倒を含む重大な軌道逸脱の抑制に効果的であることが確認された．平均最大経路誤差においても，WheelFaultHard 環境で提案手法が 0.116 と最適方策の 0.117，PID の 0.136 を下回り，一時的な大きな逸脱を抑制できていることが示された．成功エピソードのみの結果では，PID 制御が多くの環境で低い経路誤差を示しているが，これはタスクに失敗した困難なエピソードが除外されているためである．全エピソードを対象とした評価において，提案手法が PID 制御より優れた経路追従性を示したことは，困難な状況下でも安全な走行を維持できることを表していると考える．

表 5　各手法の評価結果（全エピソード）

環境	手法	成功率 [%] ↑	平均ステップ数↓	平均総経路誤差↓	平均最大経路誤差↓
Normal	最適方策	99.0	867.3	2.31	0.072
	PID	98.0	959.9	3.79	**0.064**
	提案手法	**99.6**	**867.0**	**2.19**	0.073
WheelFault	最適方策	96.4	**1557.3**	6.39	0.084
	PID	89.2	1836.6	29.70	0.145
	提案手法	**96.8**	1580.8	**5.47**	**0.076**
WheelFaultHard	最適方策	79.0	2645.8	**11.24**	0.117
	PID	91.8	**1841.4**	18.30	0.136
	提案手法	**93.2**	2250.5	11.58	**0.116**
LowFriction	最適方策	**99.0**	**873.6**	2.77	0.075
	PID	97.6	968.3	4.08	**0.065**
	提案手法	**99.0**	874.5	**2.37**	0.074
IMUNoise	最適方策	**99.6**	**843.1**	**2.07**	0.072
	PID	97.6	962.0	3.88	**0.068**
	提案手法	99.4	868.8	2.13	0.072
MassLoss	最適方策	98.4	901.0	2.59	0.075
	PID	96.8	1002.3	4.36	**0.056**
	提案手法	**99.2**	**888.2**	**2.26**	0.071

表 6　各手法の評価結果（成功エピソードのみ）

環境	手法	成功率 [%] ↑	成功時平均ステップ数↓	成功時平均総経路誤差↓	成功時平均最大経路誤差↓
Normal	最適方策	99.0	**825.6**	1.92	0.070
	PID	98.0	877.5	**1.88**	**0.053**
	提案手法	**99.6**	850.4	2.07	0.073
WheelFault	最適方策	96.4	**1428.7**	4.52	0.078
	PID	89.2	1453.6	**2.94**	**0.055**
	提案手法	**96.8**	1467.8	3.89	0.072
WheelFaultHard	最適方策	79.0	2019.9	8.84	0.114
	PID	91.8	**1559.2**	**3.52**	**0.081**
	提案手法	**93.2**	2049.9	9.30	0.110
LowFriction	最適方策	**99.0**	**832.0**	2.24	0.074
	PID	97.6	869.2	2.00	**0.055**
	提案手法	**99.0**	832.8	**1.96**	0.073
IMUNoise	最適方策	**99.6**	**826.4**	1.99	0.072
	PID	97.6	862.7	2.14	**0.057**
	提案手法	99.4	843.8	**1.93**	0.072
MassLoss	最適方策	98.4	**834.3**	2.03	0.070
	PID	96.8	870.1	2.43	**0.050**
	提案手法	**99.2**	855.1	**2.02**	0.071

6 ま と め

本研究では，学習環境とは異なる外乱や環境条件を持つ未知実証環境において，現地学習を必要とせずに差動二輪車のタスク遂行と安全性を両立させる安全推論制御手法を提案した．提案手法では，タスクの効率達成に特化した最適方策と，危険状態からの復帰に特化した安全方策という役割の異なる 2 つの方策を事前に学習し，LSTM を用いた危険判別器によって推定された危険度に基づき，方策を動的に切り替える．この切り替えにより，タスクの効率性と安全性を両立させることが可能になると考えた．

シミュレーション実験により，6 種類の未知実証環境において提案手法の有効性を検証した．車輪の深刻な非対称故障を模擬した WheelFaultHard 環境では，最適方策のみの成功率 79.0%に対し，提案手法は 93.2%を達成し，14.2 ポイントの改

善が確認された．この結果は，危険判別器が学習時に経験していない故障パターンに対しても，軌道逸脱の予兆を捉えて有効に機能することを示している．また，Normal，WheelFault，LowFriction，MassLoss 環境においても，提案手法は成功率と経路追従性の両面で優れた性能を示した．効率性の面では，困難な環境において成功時平均ステップ数が増加したが，これは安全方策による慎重な制御の結果であり，成功率の向上という形で補われていることが示された．

提案手法は状況に応じて効率性と安全性を適応的に切り替える制御を実現しており，未知実証環境における車体制御の安全性向上に寄与できると考える．

謝　　辞

本研究は，JSPS 科研費 JP24K03228 の助成，ならびに，JST【ムーンショット型研究開発事業】【JPMJMS2238】の支援を受けたものです．ここに記して謝意を表します．

文　　献

[1] Amr Eldemiry, Yajing Zou, Yaxin Li, Chih-Yung Wen, and Wu Chen. Autonomous exploration of unknown indoor environments for high-quality mapping using feature-based RGB-D SLAM. *Sensors*, Vol. 22, No. 14, 2022.

[2] Daniel Lawson and Ahmed Hussain Qureshi. Control transformer: Robot navigation in unknown environments through PRM-guided return-conditioned sequence modeling. In *Proceedings of the 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2023)*, pp. 9324–9331, 2023.

[3] Ravi Raj and Andrzej Kos. Intelligent mobile robot navigation in unknown and complex environment using reinforcement learning technique. *Scientific Reports*, Vol. 14, , 2024.

[4] Karol Arndt, Murtaza Hazara, Ali Ghadirzadeh, and Kyrre Glette. Meta reinforcement learning for sim-to-real domain adaptation. In *Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA 2020)*, pp. 2725–2731, 2020.

[5] Anusha Nagabandi, Ignasi Clavera, Simin Liu, Ronald S. Fearing, Pieter Abbeel, Sergey Levine, and Chelsea Finn. Learning to adapt in dynamic, real-world environments through meta-reinforcement learning. In *Proceedings of the 7th International Conference on Learning Representations (ICLR 2019)*, pp. 1–17, 2019.

[6] Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained policy optimization. In *Proceedings of the 34th International Conference on Machine Learning (ICML 2017)*, pp. 22–31, 2017.

[7] Shangding Gu, Long Yang, Yali Du, Guang Chen, Florian Walter, Jun Wang, Yaodong Yang, and Alois Knoll. A review of safe reinforcement learning: Methods, theory and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 46, No. 12, pp. 11216–11235, 2024.

[8] Brijen Thananjeyan, Ashwin Balakrishna, Suraj Nair, Michael Luo, Krishnan Srinivasan, Minho Hwang, Joseph E. Gonzalez, Julian Ibarz, Chelsea Finn, and Ken Goldberg. Recovery RL: Safe reinforcement learning with learned recovery zones. In *Proceedings of the 17th Robotics: Science and Systems (RSS 2021)*, pp. 1–10, 2021.

[9] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML 2017)*, pp. 1126–1135, 2017.

[10] Lin Wang, Yu Zhang, Daqi Zhu, and Sarah Coleman. Supervised meta-reinforcement learning with trajectory optimization for manipulation tasks. *IEEE Transactions on Cognitive and Developmental Systems*, Vol. 16, No. 2, pp. 681–691, 2024.

[11] Weiwei Li and Emanuel Todorov. Iterative linear quadratic regulator design for nonlinear biological movement systems. In *Proceedings of the 1st International Conference on Informatics in Control, Automation and Robotics (ICINCO 2004)*, pp. 222–229, 2004.

[12] Gregory Kahn, Adam Villaflor, Vitchyr Pong, Pieter Abbeel, and Sergey Levine. Uncertainty-aware reinforcement learning for collision avoidance. *arXiv preprint*, Vol. abs/1702.01182, , 2017.

[13] Bradley Efron. Bootstrap methods: Another look at the jackknife. *The Annals of Statistics*, Vol. 7, No. 1, pp. 1–26, 1979.

[14] Van Manh Tran and Gon-Woo Kim. Cooperative deep reinforcement learning policies for autonomous navigation in complex environments. *IEEE Access*, Vol. 12, pp. 101053–101065, 2024.

[15] Yassine Chemingui, Aryan Deshwal, Honghao Wei, Alan Fern, and Janardhan Rao Doppa. Constraint-adaptive policy switching for offline safe reinforcement learning. In *Proceedings of the 39th AAAI Conference on Artificial Intelligence (AAAI 2025)*, pp. 15722–15730, 2025.

[16] Mohammed Alshiekh, Roderick Bloem, Rüdiger Ehlers, Bettina Könighofer, Scott Niekum, and Ufuk Topcu. Safe reinforcement learning via shielding. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI 2018)*, pp. 2669–2678, 2018.

[17] Wen-Chi Yang, Giuseppe Marra, Gavin Rens, and Luc De Raedt. Safe reinforcement learning via probabilistic logic shields. In *Proceedings of the 32nd International Joint Conference on Artificial Intelligence (IJCAI 2023)*, pp. 5739–5749, 2023.

[18] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal Policy Optimization Algorithms. *arXiv preprint*, Vol. abs/1707.06347, , 2017.

[19] Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2017)*, pp. 23–30, 2017.

[20] Eugene Valassakis, Zihan Ding, and Edward Johns. Crossing the gap: A deep dive into zero-shot sim-to-real transfer for dynamics. In *Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2020)*, pp. 5372–5379, 2020.

[21] Yevgen Chebotar, Ankur Handa, Viktor Makoviychuk, Miles Macklin, Jan Issac, Nathan Ratliff, and Dieter Fox. Closing the sim-to-real loop: Adapting simulation randomization with real world experience. In *Proceedings of the 2019 IEEE International Conference on Robotics and Automation (ICRA 2019)*, pp. 8973–8979, 2019.

[22] 門垣幸樹, 大島裕明. 強化学習による差動二輪車制御における未知実証環境への適応. 第 17 回データ工学と情報マネジメントに関するフォーラム (DEIM Forum 2025), 2025.

[23] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, Vol. 9, No. 8, pp. 1735–1780, 1997.

[24] John H. Holland. Genetic algorithms. *Scientific American*, Vol. 267, No. 1, pp. 66–73, 1992.