人工知能学会第二種研究会資料
AIチャレンジ研究会
SIG-Challenge-066-07

世界モデルベースの深層強化学習による音源追跡の検討

Sound source tracking using deep reinforcement learning based on world model

平塚 謙良 [1] *　小島 諒介 [2]
Kaneyoshi Hiratsuka[1]　Ryosuke Kojima[2]

[1] 京都大学工学部物理工学科
[1] Undergraduate School of Engineering Science, Kyoto University
[2] 京都大学大学院医学研究科
[2] Graduate School of Medicine, Kyoto University

Abstract: ロボットが周囲の音環境を認識し，与えられたタスクをこなす問題は，カメラの使用が難しい状況，たとえば遮蔽物が多い場面で特に重要である．この技術は，救助ロボットやサービスロボットが周囲の音環境を理解し，適切に対応する上で欠かせない技術として注目されている．本研究では，環境のダイナミクスをモデル化する「世界モデル」を用いた深層強化学習手法を音環境へと拡張した手法を提案し，音源追跡タスクに適用して評価を行う．具体的には、世界モデルベースの深層強化学習である DreamerV3 を拡張し，マイクロフォンアレイを用いて音源定位を行った結果を音環境情報として統合可能な手法を開発した．シミュレーション実験により，音源追跡タスクにおける提案手法のパラメータや環境を変えた場合の影響を評価した．本研究の評価実験のソースコードは github（`https://github.com/Azuma413/sound_wm_turtlebot`）から利用可能である．

1　はじめに

ロボットに搭載されたマイクロフォンアレイを用いて周囲の音環境を理解する問題は，ロボット聴覚の分野で長く研究されており，特に障害物が多い環境などでカメラなどの他のセンサの利用が難しい場合には，音が重要な手掛かりとなるためロボットの環境認識にとって重要である [1, 2]．例えば，ドローンや移動ロボットに搭載されたマイクロフォンアレイを用いて音源定位および音源追跡を行う技術は様々な場面で必要とされており，救助ロボットが助けを求めている人の位置を特定する手法 [3] や室内での音情報を用いたナビゲーション手法 [4, 5] など多くの研究が報告されている．これらの応用では，周囲の音環境を認識すると同時に，音に近づいて正確な救助者の位置を見つけるタスクや，障害物を避けて目的音源の近くに到達するといったタスクを達成することが求められる．

一方で，環境とのインタラクションによって与えられたタスクを達成するための行動を決定する問題は強化学習として定式化され，特に，近年の深層学習技術の発展に伴い，深層学習を用いた深層強化学習技術が注目されている．中でも，環境を陽にモデルとして学習する「世界モデル」を用いた強化学習は，観測データの背後にあるダイナミクスを学習することで，未知の環境や変化に対するロバスト性の向上を目指している [6]．世界モデルをベースにした手法は，ビデオゲームでの強化学習や様々な環境でのプランニングなど画像タスクにおいて高い性能を示してきた [6, 7]．また，これらの手法は，エージェントが観測データから現在の状況をどのように認識しているかや将来をどう予測しているかを確認できるといった解釈性の利点も備えていることも多い．

本研究では，世界モデルベースの深層強化学習を音環境へと拡張する手法を提案し，音源追跡タスクを通じてその有効性を検証する．具体的には，これまで主に画像に対して適用されていた世界モデルベースの深層強化学習手法である DreamerV3[4] に対して，マイクロフォンアレイから得られる音源の推定位置情報を統合できるように拡張する．

深層強化学習を用いた音源追跡や音環境下のナビゲーションの既存研究として，音響特徴量を利用する手法 [4] や，信号データから end-to-end で学習する手法 [8] が知られている．我々の世界モデルベースの手法は深層強化学習の間にエージェントが認識している状態を確認できるといった解釈性が期待でき，音源定位のモジュールを明確に強化学習部分から分離することで，従

*連絡先： 京都大学工学部
京都府京都市左京区吉田本町
E-mail: hiratsuka.kaneyoshi.72i@st.kyoto-u.ac.jp

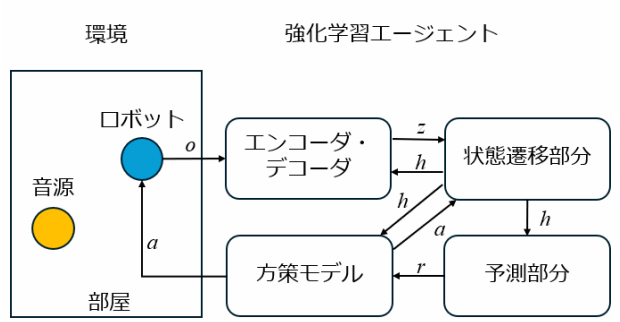

図 1: 強化学習エージェントである DreamerV3 と，環境の関係性．図中の o は観測， a は行動，z は観測の潜在表現，h は状態ベクトル，r は報酬を表す．

来の音源定位技術がそのまま利用できるという特徴がある．

2 前提

本章では，本研究で用いる強化学習アルゴリズムである DreamerV3 についての説明を行うとともに，本研究が対象とする問題設定について述べる．

2.1 DreamerV3

DreamerV3 は，通常の強化学習と同様に，エージェントが環境と相互作用しながら報酬を最大化することを目的としている．DreamerV3 の特徴として，環境の潜在表現やその遷移を学習する世界モデルを内包するモデルベースの強化学習アルゴリズムである点が挙げられる．世界モデルは，観測されたデータから潜在的な状態空間を学習することによって，状態遷移や報酬の予測を行うモデルであり，これによりエージェント・環境間の相互作用を最小限に抑えながら，仮想環境内での効率的な方策の学習を実現する．ここでは，強化学習の基本的な要素である，環境，状態，行動，報酬，方策，価値関数に着目しつつ，以下の式に示す DreamerV3 の構成要素である，エンコーダ・デコーダ部分，状態遷移部分，予測部分，および Acrtor-critic 部分のそれぞれについて説明する（図 1 右）．

$$\mathbf{z}_t \sim q(\mathbf{z}_t|\mathbf{h}_t, \mathbf{o}_t) \tag{1}$$
$$\hat{\mathbf{o}}_t \sim p(\hat{\mathbf{o}}_t|\mathbf{h}_t, \mathbf{z}_t) \tag{2}$$
$$\mathbf{h}_t = f(\mathbf{h}_{t-1}, \mathbf{z}_{t-1}, a_{t-1}) \tag{3}$$
$$\hat{r}_t \sim p(\hat{r}_t|\mathbf{h}_t, \mathbf{z}_t) \tag{4}$$
$$\hat{c}_t \sim p(\hat{c}_t|\mathbf{h}_t, \mathbf{z}_t) \tag{5}$$

DreamerV3 のエンコーダ・デコーダ部分（式 (1), (2)）は観測 $\mathbf{o}_t$（その推定値 $\hat{\mathbf{o}}_t$）と潜在表現 $\mathbf{z}_t$ を互いに変換する構成になっている．DreamerV3 では，観測として画像を想定しており，入力画像は主に畳み込みニューラルネットワーク（CNN: convolutional neural network）から構成されるエンコーダによって潜在表現に圧縮して取り扱われる．状態遷移部分（式 (3)）は現在時刻 $t-1$ の潜在表現のベクトル $\mathbf{h}_t$，$\mathbf{z}_t$ および行動 a_{t-1} を用いて，次の時刻 t の潜在ベクトル $\mathbf{h}_{t-1}$ をニューラルネットワークによって計算している．予測部分（式 (4), (5)）では潜在表現を用いて，単純な全結合層により構成される予測ヘッドを用いて報酬 $\hat{r}_t$ や，エピソードの終了を表す値 $\hat{c}_t$ の予測を行っている．これらの予測値 $\hat{r}_t$ や $\hat{c}_t$ を用いることで，直接環境を利用せずに方策モデルを学習することができる．DreamerV3 の Acrtor-critic 部分では，Acrtor-critic をベースに，潜在状態を入力にして，行動（連続値もしくは離散値）を出力する方策ネットワークと潜在状態を入力にして状態の価値を出力する状態価値関数を学習している．ここで，状態価値関数を学習する際には，世界モデルを利用して k ステップ先までを予測し，その平均の報酬を用いて学習する．この時の k を Imagination Horizon と呼ぶ．また，本稿では，エンコーダ・デコーダ部分，状態遷移部分，予測部分をまとめて世界モデルと呼ぶこととする．

2.2 問題設定

本研究では，単一のマイクロフォンアレイを備えた自己位置推定が可能な移動ロボット（実際の例は図 8 に示す）一台を想定し，室内にある移動音源の位置を特定し，追跡するための方策を学習するタスクに取り組む．

図 1 は DreamerV3 から構成される強化学習エージェントと，環境の関係性を図示したものである．ロボットが収集した情報は観測として世界モデルに与えられ，世界モデルの学習に使われる．方策モデルは世界モデルに行動を送り，返ってきた潜在状態をもとに学習を行う．また，方策モデルから出力される行動は環境内のロボットにも反映され，世界モデルに与えられる観測情報が更新される．

3 提案手法：DreamerV3 による音源追跡

本章では，DreamerV3 を音環境へと拡張することで，移動ロボットを用いた音源追跡を可能とした提案手法について説明する．具体的には，移動ロボットを制御する強化学習エージェントに与える観測空間（特に，音環境を観測として世界モデルへと入力する方法）や報酬の設計について述べる．

3.1 観測の作成

本提案手法では，音源定位マップ（$\mathbf{S}$），地図情報マップ（$\mathbf{M}$），ロボット位置情報マップ（$\mathbf{G}$）の 3 つの 2 次元マップ情報を用いる．これらは 2 次元情報なので，2 章で述べた 2 次元画像に対する CNN を用いたエンコーダ・デコーダを本提案手法でも同様に利用できる．以下ではこれらの 2 次元マップの作成方法について述べる．

3.1.1 音源定位マップの作成

音源定位マップは音源方向推定（DOA: Direction of Arrival）の結果を利用して作成する．ここで，音源定位のアルゴリズムにはノイズに対して比較的ロバストに動作することから MUSIC（MUltiple SIgnal Classification）法を採用した．具体的な音源定位を行うための手順としては，まず，多チャンネル音響信号をバッファに貯め，逐次，短時間フーリエ変換（STFT: Short-Time Fourier Transform）によって 8ch のスペクトログラムに変換する．この時のステップ幅の単位をここでは 1 frame と呼ぶ．STFT 計算後のスペクトログラムから，MUSIC 法では相関行列を計算しており，最終的に，マイクロフォンアレイを中心に 1 度刻みでの音源方向に対応するパワー $s(\theta)$（MUSIC スペクトル）を得る．本実験設定では，16kHz 8ch のマイクロフォンアレイから得られる信号を利用しており，STFT の窓幅は 256 [sample]，ステップ幅は 128 [sample] とした．MUSIC 法の相関行列の計算は 139 [frame] ごとに行い，伝達関数はマイク配置から決まる幾何計算によって求めた．

上述の MUSIC スペクトル $s(\theta)$ に基づき，各タイムステップごとに音源定位マップの各ピクセルの値を更新することで音源定位マップを作成する．具体的には以下の順に計算を行う．

まず，各ピクセルの位置に対応する角度 θ_i [rad] を $\theta_i = \arctan\left(\frac{y-p_y}{x-p_{r,\,x}}\right)$ により計算する．ここで，(x, y) は音源定位マップ上のピクセル座標であり，$\mathbf{p} = (p_x, p_y)$ はマイクロフォンアレイの座標である．

次に，マップのピクセル値 $\mathbf{S}_{x,\,y}$ を次のように更新する．

$$\mathbf{S}_{x,\,y} \leftarrow \max(D, \mathbf{S}_{x,\,y} \cdot ((\max(s(\theta_i), A) - A) \cdot B + C))$$

ここで，A は MUSIC スペクトル $s(\theta)$ の値が閾値 A より小さい場合に，その影響を無視するためのパラメータである．B，C は閾値による処理後の MUSIC スペクトルのスケールとオフセットを調整するパラメータである．D は，更新後のマップのピクセル値が D より小さい場合に D でクリッピングするパラメータであり，これは乗算により値を更新する都合上，値が小さくなりすぎると更新がほとんど行われなくなってしまう問題への対策である．4 章の実験では $(A, B, C, D) = (4.5, 0.05, 0.85, 3)$ とした．

3.1.2 地図情報マップの作成

地図情報マップにはロボットが観測した部屋の形状情報を表しており，移動可能かそうでないかを表す 2 値の 2 次元マップであり，以下のように定義される．

$$\mathbf{M}_{x,y} = \begin{cases} 0 & (x, y) \text{ が通行可能,} \\ 1 & (x, y) \text{ が通行不可能.} \end{cases}$$

地図情報マップの指定方法は二つあり，それは部屋の地図情報をあらかじめ指定する方法と SLAM（Simultaneous Localization and Mapping）の結果を利用する方法である．SLAM は，移動ロボットや自動運転車などが未知の環境を探索する際に，自己位置を推定しながら周囲の環境地図を同時に構築する技術である．SLAM の実行には，カメラや LiDAR などのセンサーから得られるデータと，加速度計やエンコーダのデータから算出されたロボットの運動モデルが利用できる．

3.1.3 ロボットの位置情報マップの作成

位置座標マップには，ロボットの位置を 2 次元マップ情報として格納しており，その方法として $\mathbf{G}^{(1)}$，$\mathbf{G}^{(2)}$ の 2 種類を考える．

$\mathbf{G}^{(1)}$ の方法では，マップの値を以下の式に示すようにロボットの座標 $\mathbf{p}$ を，0 から 1 に正規化した値が設定される．

$$\mathbf{G}^{(1)}_{x,\,y} = \begin{cases} \frac{p_x}{W} & \text{if } x \leq \frac{W}{2}, \\ \frac{p_y}{H} & \text{if } x > \frac{W}{2}. \end{cases}$$

ここで，W は画像の横幅，H は画像の縦幅である．

一方で $\mathbf{G}^{(2)}$ の方法では，ロボットの座標 $\mathbf{p}$ に対応するマップ上の座標の，周囲の n ピクセルに 1 を割り当て，それ以外のピクセルに 0 を割り当てる．

$$\mathbf{G}^{(2)}_{x,\,y} = \begin{cases} 1 & \text{if } |p_x - x| < n \wedge |p_y - y| < n, \\ 0 & \text{otherwise.} \end{cases}$$

3.2 報酬の作成

強化学習で最大化する対象である報酬 R は重み w を用いて，以下の 2 つの要素の重み付き和で表現する（4 章の実験時には $w = 0.4$ とした）．

$$R = wR_{\text{dist}} + (1 - w)R_{\text{est}}$$

この成分の 1 つは，音源定位マップ $\mathbf{S}$ で表現される音源のばらつく範囲の小さいほど高い報酬を与える「分布報酬（R_{dist}）」．もう 1 つは，音源定位マップに基づいて推定された音源の位置と，実際の音源の位置の誤差が小さいほど高い報酬を与える「推定報酬（R_{est}）」である．最終的な報酬は-1 から 1 の範囲内として，例外として，ロボットが壁にぶつかった場合は報酬として -1 を与える．分布報酬と推定報酬に関する詳細な定義を以下の節で説明する．

3.2.1 分布報酬

分布報酬は，音源の位置の推定のばらつきが小さくなることに対する報酬として設計している．具体的には，分布報酬 R_{dist} は以下のように計算する．

$$R_{\text{dist}} = \left(2 \cdot e^{-n \cdot E} - 1\right)$$

$$n = \sum_{x,y} I[\mathbf{S}_{x,\,y} > c]$$

ここで，式中の $I[\cdot]$ は引数が真ならば 1 を返す指示関数である．ただし，E は分布報酬用のパラメータ，c は音源の存在重みに対するしきい値パラメータとする．この分布報酬では，しきい値 c より大きい値を持つ音源定位マップ $\mathbf{S}$ のピクセル数 n を用いて，ピクセル数 n が少ないほど報酬 R_{dist} が高くなるように，指数関数を利用して調整している．この指数関数 $e^{-n \cdot E}$ は，n が小さい場合に急激に減少するため，ピクセル数が少ない場合の変化を強調する役割を果たす．その後，$2 \cdot e^{-n \cdot E} - 1$ という変換を行うことで，報酬値を -1 から 1 の範囲に正規化している．4 章の実験では $E = 0.001$，$c = 0.7$ とした．

3.2.2 推定報酬

推定報酬 R_{est} は音源の位置の推定精度が高くなることに対する報酬として以下のように計算される．

$$R_{\text{est}} = \left(2 \cdot \exp\left(-F \cdot ||\mathbf{m}_{\text{est}} - \mathbf{m}_{\text{real}}||\right) - 1\right)$$

$$\mathbf{m}_{\text{est}} = \frac{1}{n} \sum_{x,\,y} I[\mathbf{S}_{x,\,y} > c] \begin{bmatrix} x \\ y \end{bmatrix}$$

ただし，$\mathbf{m}_{\text{est}}$ は音源の推定位置の中心座標を表しており，F はスケーリングパラメータである．ここで，$\mathbf{m}_{\text{real}}$ は実際の音源位置の座標を表している．推定報酬は，推定した音源位置 $\mathbf{m}_{\text{est}}$ が実際の音源位置 $\mathbf{m}_{\text{real}}$ に近いほど高い報酬を与えることで，推定精度が向上するように方策を学習することを目的として設計している．推定位置 $\mathbf{m}_{\text{est}}$ は，しきい値 c を超える画素の重心座標として計算される．音源の推定誤差は $||\mathbf{m}_{\text{est}} - \mathbf{m}_{\text{real}}||$ で計算され，指数関数を利用してその値が小さいほど報酬が高くなるようにしている．その後，報酬が -1 から 1 の範囲を取るようにスケールとオフセットを調整している．この際に，しきい値を超える画素が存在しない場合，推定位置 $\mathbf{m}_{\text{est}}$ が定義できないため，この場合は報酬として -1 を与える．この処理は，音源位置の推定に失敗した場合にペナルティを与える目的がある．4 章の実験では $F = 0.5$ とした．

4 実験

2 章で導入した音源追跡タスクに対して，DreamerV3 を用いた提案手法の評価実験を行った．最初に，前実験と DreamerV 3 のパラメータに関する 2 実験，ロボットが動作する部屋環境の違いによる影響を評価する実験を行った．その後，他の強化学習モデルとの比較実験と，音源数および移動音源に対する実験を行った．

4.1 実験設定

本実験では，図 1 に示した状況における音源追跡タスクをシミュレーションを用いて検証する．シミュレーションには音響シミュレーションライブラリである Pyroomacoustics を用いた．Pyroomacoustics は音波の反射や減衰をシミュレートすることが可能であり，音源の位置や部屋の形状を変化させてシミュレーションを行うことができる．

DreamerV3 に与える観測データは 128×128 ピクセルの 3 つの 2 次元マップを使用し，3 チャンネルの情報として DreamerV3 に入力する．各エピソードは最大 100 ステップで構成され，各ステップでの報酬は-1 から 1 の範囲内に設定した．したがって，エピソード全体の報酬は-100 から 100 の範囲を取る．学習は合計 100,000 ステップ行い，各設定で 5 回ずつ実験を実施した．基準条件として，モデルサイズを Medium，Imagination Horizon を 15，部屋の形状を長方形，音源数を 1 とした設定をもとに他の条件を変化させ，タスクの性能への影響を検証した．

4.2 前実験: 位置情報マップの比較

ロボットの位置情報マップ $\mathbf{G}^{(1)}$，$\mathbf{G}^{(2)}$ の 2 つの表現方法を評価するために前実験を行った．

図 2 より，学習速度において $\mathbf{G}^{(1)}$ の手法は $\mathbf{G}^{(2)}$ の手法よりも早い段階で高い報酬が得られていることが分かる．

また，$\mathbf{G}^{(1)}$，$\mathbf{G}^{(2)}$ それぞれの場合についてモデルへの入力と，それに対応する DreamerV3 内部のデコーダによる再構成画像を比較し，図 3，4 に示す．図は 3 種

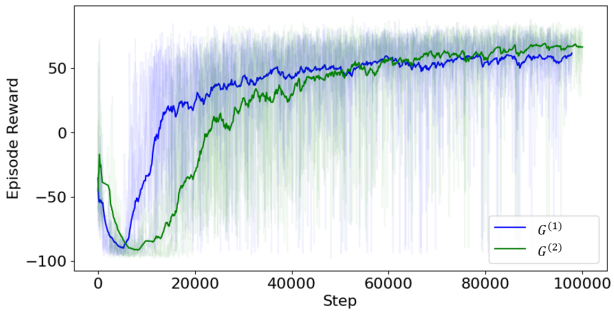

図 2: ロボットの位置情報の与え方を変化させた際の報酬の変化．横軸は強化学習の学習ステップ数，縦軸はエピソード全体の報酬和を表す．青線が $\mathbf{G}^{(1)}$ の方法で位置情報を表現した場合，緑線が $\mathbf{G}^{(2)}$ の方法で位置情報を表現した場合．

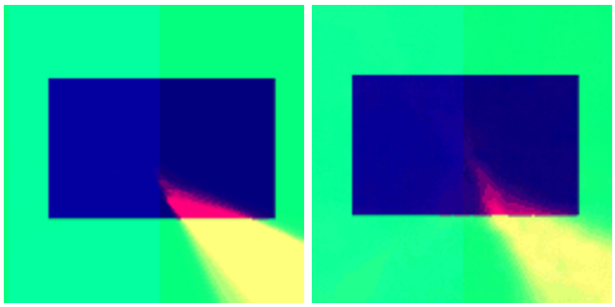

図 3: $\mathbf{G}^{(1)}$ の手法を用いた際の，入力画像（左）とデコーダの出力画像（右）の比較．

類のマップ $\mathbf{S}$，$\mathbf{M}$，$\mathbf{G}$ を，画像の Red，Green，Blue の 3 つのチャンネルにそれぞれ割り当てたものである．このとき，図 3 より，$\mathbf{G}^{(1)}$ の手法では Blue チャンネルを正しく再構成できている一方で，$\mathbf{G}^{(2)}$ の手法では，Blue チャンネルの特徴をうまく再構成できていないことが分かる．

以上の前実験の結果を踏まえ，以降ではロボットの位置情報をマップ全体で表現する $\mathbf{G}^{(1)}$ の手法を採用する．

4.3 モデルサイズの影響

ここでは DreamerV3 の世界モデル部分のモデルサイズがタスクの性能に与える影響を評価する．実験結果を図 5 に示す．それぞれのモデルサイズにおける，パラメータ数の詳細を表 1 に示す．ここで，表中の CNN Depth はエンコーダとデコーダのそれぞれ CNN の深さ，Reward Layers と Reward Head Units は報酬予測ヘッドにおける隠れ層の数とその次元数，Cont Layers と Cont Head Units はエピソード終了を予測するヘッドの隠れ層の数とその次元数を表す．

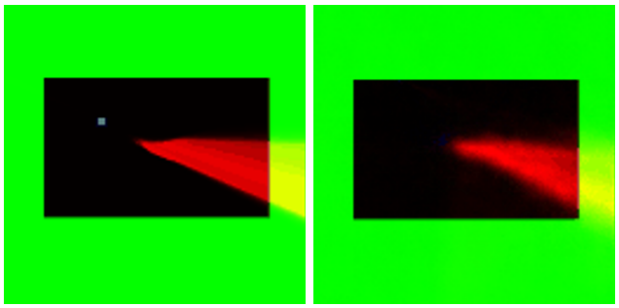

図 4: $\mathbf{G}^{(2)}$ の手法を用いた際の，入力画像（左）とデコーダの出力画像（右）の比較．

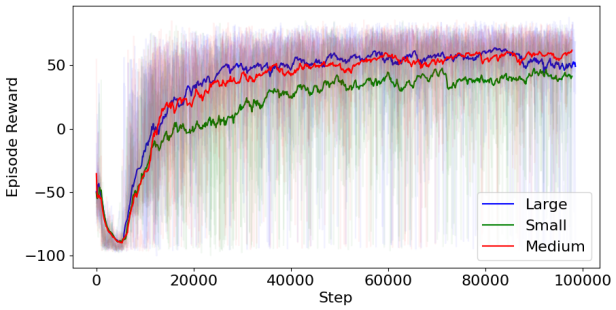

図 5: DreamerV3 のモデルサイズの変更によるエピソード報酬の変化．緑線はモデルサイズを Small，赤線は Medium，青線は Large としたときのエピソード報酬の推移を表す．5 回分のそれぞれの試行の報酬の値を細線，その指数移動平均を太線として示している．

図 5 から分かるように，モデルサイズが小さすぎると環境に対して行動を十分最適化することができず，性能が低下することが分かる．一方で，モデルサイズを Medium とした場合と，Large とした場合を比較すると，性能の改善に大きな影響は見られないことが分かる．この結果から，タスクの難易度に応じた適切なモデルサイズが重要であり，過剰に大きなモデルは性能向上に寄与しない事が分かる．

4.4 Imagination Horizon の設定

Imagination Horizon は 2 章で説明したように，状態価値関数の学習時に世界モデルを用いて将来の観測や報酬を予測する際のステップ数を指定するパラメータである．Imagination Horizon の長さがタスクに与える影響を評価した結果を図 6 に示す．

この結果から，Imagination Horizon が短すぎると学習速度が低下し，逆に長すぎると最終的な性能が低下することが示された．これは，Imagination Horizon が短いと短期的な環境変化しか考慮できないため，適切な価値を学習するのに時間がかかるのだと考えられる．

表 1: モデルサイズとパラメータの関係

Parameter	Small	Medium	Large
h の次元数	512	1024	2048
z の次元数	512	640	768
CNN Depth	32	48	64
Reward Head Units	512	640	768
Cont Head Units	512	640	768
Reward Layers	2	3	4
Cont Layers	2	3	4

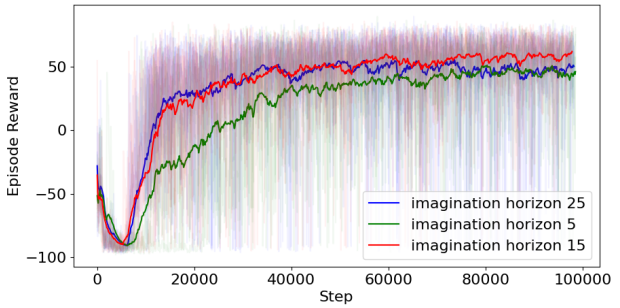

図 6: Imagination Horizon を変更したときの報酬の変化．緑は 5 ステップ，赤は 15 ステップ，青は 25 ステップの設定における結果．

また，一般に Imagination Horizon を長くするほど環境予測の分散は大きくなるため，それが最終的なモデル性能に悪影響を及ぼしている可能性がある．

4.5 部屋の形状による影響

部屋の形状を変更して，図 7 に示す形状に変化させた場合の評価を行った．

ここで，Room 0-2 は座標を指定して作成した地図情報であり，Room 3 は LiDAR を用いた SLAM によって作成した実際の部屋の地図情報である．ここで SLAM を実行するに当たり，図 8 に示す移動ロボットを用いた．また，全ての部屋において部屋の高さは 3m とした．

それぞれの部屋でエージェントを学習させた結果（図 9）から，長方形型の Room 0 と L 字型の Room 1 に関してはモデルの性能にはほとんど変化が生じていないことがわかる．一方で，部屋に仕切りがある Room 2 やさらに複雑な形状を持つ Room 3 の場合に関しては，同一ステップ数においての性能が低下していることが分かる．これらの性能低下の主な原因は，エージェントが仕切りへの衝突を回避する方策を十分に学習するためにはステップ数が多く必要になるためと考えられる．また，一度衝突した際に，壁から十分に離れる行動が選択されない場合には，罰則を受け続けるという問題もあることが分かった..

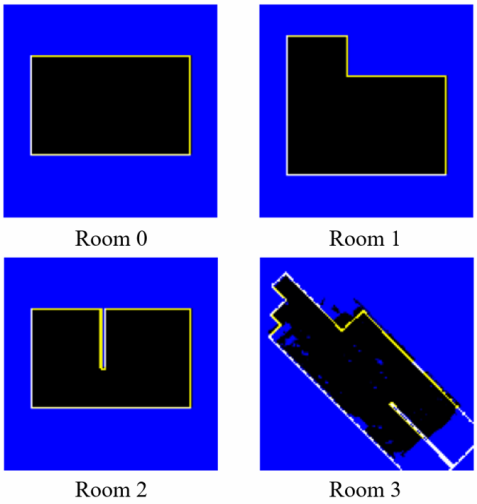

図 7: 検証した部屋の形状．青はアクセスできない領域，黄色は音の反響をシミュレートする際の壁の位置を示す．

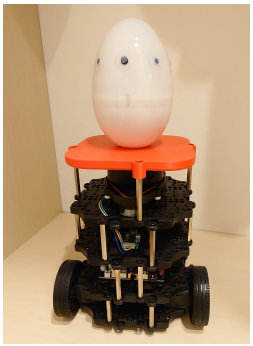

図 8: ROBOTIS 社 TurtleBot3 Burger に 8ch マイクロフォンアレイ TAMAGO (SYSTEM IN FRONTIER Inc.) を搭載したロボット．

4.6 手法間比較

DreamerV3 を用いた提案手法と，DrQ-v2（Data-regularized Q-v2）[9]，SAC（Soft Actor-Critic）[10] を用いた場合とで比較を行った．DrQ-v2 は画像入力に対して高いパフォーマンスを発揮するため，モデルフリー手法の代表例として採用し，SAC（Soft Actor-Critic）は連続制御タスクにおいて広く用いられる手法であるため採用した．これらの比較手法における観測データは，提案手法と同様の 128×128 ピクセルの 3 チャンネルの情報として利用する．

エージェントの学習結果（図 10）より，SAC に対しては学習速度においても最終性能においても DreamerV3 が上回っていることが分かる．一方で，DreamerV3 と DrQ-v2 を比較すると，学習速度においても最終性能においても DrQ-v2 の方が高い結果であった．既報論文

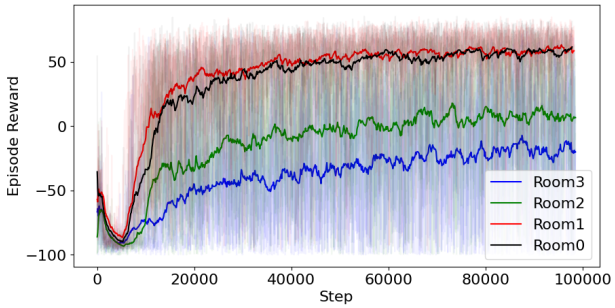

図 9: 部屋の形状変更による報酬の変化．黒線は Room 0，赤線は Room 1，緑線は Room 2，青線は Room 3.

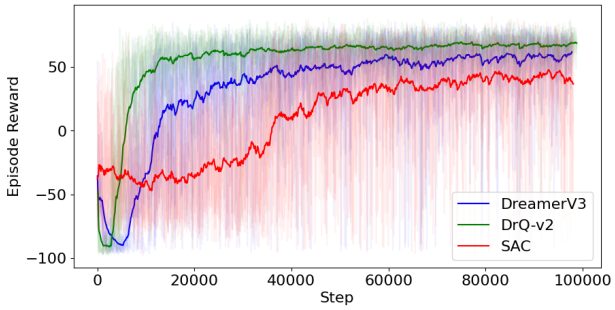

図 10: 異なるモデルで学習を行った場合のエピソード報酬の変化．緑線は DrQ-v2，青線は DreamerV3，赤線は SAC.

では VisualControl タスク [11, 6] において DreamerV3 は平均的に DrQ-v2 を上回っているというデータが示されているが，今回の音環境の音源追跡タスクでは異なることが分かった．

4.7　音源の数と移動音源の影響

最後に，音源の数を増やした場合と音源が移動する場合にエージェントの性能が固定音源一つの場合と比べてどれくらいの性能かを評価した．検証した状況は，複数音源の状況は，固定音源が室内のランダムな位置に 2 つ存在する状況を想定した．また，移動音源の状況は，一つの音源が室内のランダムな位置で円運動をしている場合を想定した．ただし，移動音源は，部屋の中心から $1.4m$ 四方の範囲内でランダムに中心を取り，半径 $0.7 + U(0, 0.3)$ の円上を一周 $70 + U(0, 30)$ ステップで動くものとした．$U(a, b)$ は a〜b の範囲で一様な乱数とする．

それぞれの状況でのエージェントの学習結果（図 11）より音源の数が増えると，エピソード報酬は大きく下がることが分かった．その要因として，今回の環境に採用している報酬関数の設計上，複数の音源が存在している環境において十分機能していないことが挙げられる．また複数の音源が存在する場合は，DOA の精度も低下していると考えられ，これもエピソード報酬の低下の要因として考えられる．一方で，単一の音源が移動する場合においては，エージェントは比較的に性能を維持できていることが確認できた．単一の固定音源の環境で学習結果と比較すると，やや性能が低下しているが，この原因は，３章で説明した音源定位マップの更新パラメータを移動音源に合わせて設定する必要があり，移動音源に対して音源定位マップの更新が適切に行えなかった可能性がある．

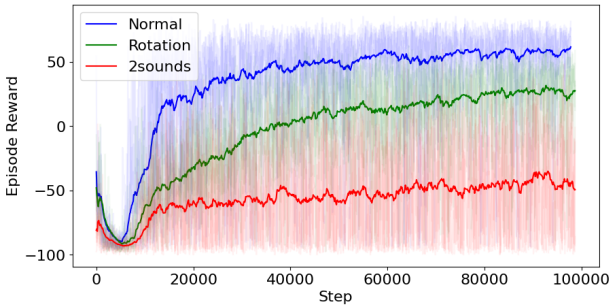

図 11: 音源数 2 の状況と移動音源の状況に対するエピソード報酬の変化．赤線は固定音源が 2 つの場合，緑線は移動音源が 1 つの場合，青線は固定音源が 1 つの場合．

実際に，シミュレーション環境内で移動音源をロボットが追跡している様子を図 12 に示す．この結果から移動音源に対しても音源位置はうまく推定できており，位置推定性能を上げるために音源の周囲をまわる動作が学習できていることが分かる．

5　おわりに

マイクロフォンアレイからの音源定位結果を情報統合できるよう，世界モデルを利用したモデルベース強化学習アルゴリズムである DreamerV3 を拡張する手法を提案し，音源追跡タスクへの適用可能性を評価した．シミュレーションを用いて DreamerV 3 のパラメータに関する実験を行い，音源追跡タスクにおける各パラメータの影響を確認した．また，異なる環境条件をシミュレーションすることで，提案手法の学習結果が部屋の形状や音源数の影響を受けること，音源の移動に関しては一定のロバスト性を示すことを明らかにした．今後の課題として，シミュレーション環境で学習した方策を実環境に適用すること，end-to-end の学習との比較や融合，さらには視覚情報の統合などが挙げられる．

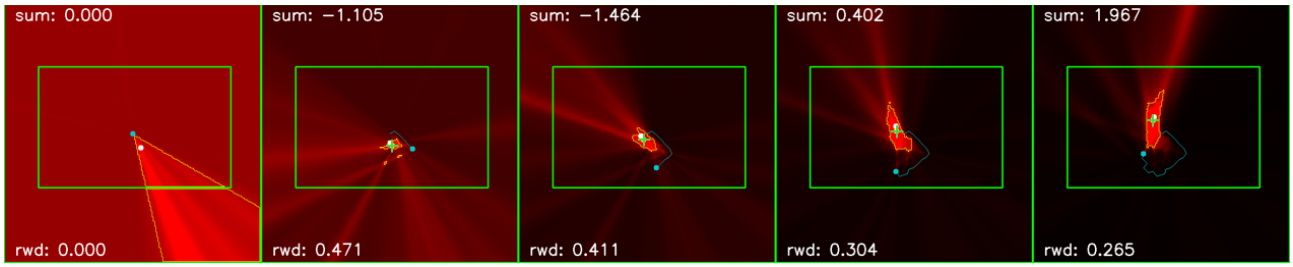

図 12: シミュレーション環境内でのを 5 ステップ毎に可視化した画像緑の線で囲まれた範囲が室内であり，ロボットが移動可能な領域を示す．また，音源定位マップを赤色で表現する．ロボットは水色の円で表現され，その軌跡は水色の細線で表現されている．実際の音源の位置を白い丸，推定された音源の位置を緑色の十字で表現している．また，黄色の細線で囲われた領域はしきい値 c を超える値を持つ音源定位マップのピクセルを示している．左下の数字はそのステップにおける報酬の値であり，左上の数字はそのステップにおける累積報酬の値である．

謝辞

本研究は JSPS 科研費 No. 21H04905 および CREST JPMJCR22D3 の助成を受けた．

参考文献

[1] Kazuhiro Nakadai and Hiroshi G Okuno. Robot audition and computational auditory scene analysis. *Advanced Intelligent Systems*, Vol. 2, No. 9, p. 2000050, 2020.

[2] Kazuhiuro Nakadai, Hiroshi G Okuno, Hirofumi Nakajima, Yuji Hasegawa, and Hiroshi Tsujino. An open source software system for robot audition hark and its evaluation. In *Humanoid Robots, 2008*, pp. 561–566, 2008.

[3] Taiki Yamada, Katsutoshi Itoyama, Kenji Nishida, and Kazuhiro Nakadai. Assessment of sound source tracking using multiple drones equipped with multiple microphone arrays. *International journal of environmental research and public health*, Vol. 18, No. 17, p. 9039, 2021.

[4] Chuang Gan, Yiwei Zhang, Jiajun Wu, Boqing Gong, and Joshua B Tenenbaum. Look, listen, and act: Towards audio-visual embodied navigation. In *ICRA*, pp. 9701–9707. IEEE, 2020.

[5] Changan Chen, Carl Schissler, Sanchit Garg, Philip Kobernik, Alexander Clegg, Paul Calamia, Dhruv Batra, Philip W Robinson, and Kristen Grauman. Soundspaces 2.0: A simulation platform for visual-acoustic learning. In *NeurIPS 2022 Datasets and Benchmarks Track*, 2022.

[6] Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models, 2024.

[7] Baris Kayalibay, Atanas Mirchev, Patrick van der Smagt, and Justin Bayer. Tracking and planning with spatial world models. In *Proceedings of The 4th Annual Learning for Dynamics and Control Conference*, Vol. 168, pp. 124–137. PMLR, 2022.

[8] Changan Chen, Unnat Jain, Carl Schissler, Sebastia Vicenc Amengual Gari, Ziad Al-Halah, Vamsi Krishna Ithapu, Philip Robinson, and Kristen Grauman. Soundspaces: Audio-visual navigation in 3d environments. In *ECCV*, pp. 17–36. Springer, 2020.

[9] Denis Yarats, Rob Fergus, Alessandro Lazaric, and Lerrel Pinto. Mastering visual continuous control: Improved data-augmented reinforcement learning, 2021.

[10] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor, 2018.

[11] Yuval Tassa, Yotam Doron, Alistair Muldal, Tom Erez, Yazhe Li, Diego de Las Casas, David Budden, Abbas Abdolmaleki, Josh Merel, Andrew Lefrancq, Timothy Lillicrap, and Martin Riedmiller. Deepmind control suite, 2018.