

単調環境における協調自己位置推定のための Collaborative Stein Particle Filter

Tomoki Arita^{1†} and Toru Namerikawa²

¹School of Integrated Design Engineering, Keio University, Kanagawa, Japan

²Department of System Design Engineering, Keio University, Kanagawa, Japan
(Tel: +81-45-563-1151; E-mail: arita.tomoki@keio.jp, namerikawa@sd.keio.ac.jp)

Abstract: 農地や森林などの単調で広大な屋外環境では、画像マッチングによるグローバル自己位置推定は、累積誤差や外れ値の影響により失敗することが多い。特に複数エージェントによる協調推定では、各エージェントの誤った状態推定が全エージェントに影響し、システム全体の誤った推定確率がエージェントごとの外れ値の割合に応じて指数関数的に増加する。本研究では、Stein Particle Filter と Relaxed ADMM を組み合わせることで、従来の画像特徴量のマッチングに位置情報による尤度を考慮しつつ、複数のエージェント間で推定状態を合意するフレームワークを提案する。提案手法は、特殊ユークリッド空間 $SE(d)$ 上での確率分布の表現と操作、ADMM による分散最適化、パーティクルベースの表現による不確実性の適切な取り扱いにより、複数エージェント間の合意制約と多峰性分布の扱いを同時に可能にし、従来手法では困難であった外れ値混在環境下での安定的な位置合意を実現する。2次元および3次元シミュレーションと実機実験結果から、提案手法が外れ値の割合が0.2程度の環境下でも安定した収束を達成し、分布近似と勾配情報を活用することで、既存手法よりもロバストかつ柔軟な自己位置推定が可能であることが示された。

Keywords: Collaborative Localization, Unmanned Aerial Vehicles, Stein Particle Filter, Monotone Environments, Visual-Inertial Navigation System.

1. INTRODUCTION

多くの自律移動可能なロボットは、動作の基盤となる自己位置推定を必要とする。また、自律移動ロボットの社会への浸透に伴って、過酷環境や山間地域等GPS/GNSSが利用できない環境での自己位置推定技術が数多く研究されている。自己位置推定には、力学制御や局所の動作計画などに利用する局所の自己位置推定の階層と、一貫した環境地図の作成や広域の経路計画に利用する大域の自己位置推定の階層が存在するが、GPS/GNSSが利用できない環境における大域の自己位置推定では、主にLiDARやカメラ等によって得られた点群及び画像データを種々のマッチング手法で構造化することにより自己位置推定を達成する手法が一般的である。実用的には、カメラとIMU(慣性計測ユニット)を組み合わせるVisual Inertial System(VINS)が安価にハードウェアを実装できる点から多く利用されている[1]。

更に、自律移動ロボットの普及によって、複数のエージェントが同時に動作するマルチエージェント環境が近年主要な研究分野となっている。GPS/GNSSが使用できない過酷環境や山間地域においても、システム全体としての対故障性やそれぞれの目的におけるスケーラビリティの観点から、いくつかの研究でマルチエージェントによる協調作業が試みられている[2][3]。

これらの背景から、GPS/GNSSを用いない大域自己位置推定手法が求められているところであるが、農地や森林など単調かつ広大な野外環境下では、誤差の蓄積と外れ値の影響により画像のマッチングによる大域の自己位置推定が十分に機能しない。このような環境では、視覚的特徴の類似性や反復性により、特徴点マッチングに基づく従来のSLAM手法は誤対応を生じやすく、結果として位置推定の精度が著しく低下する。特に複数のエージェントによる協調推定において

は、各エージェントの誤った状態推定が全エージェントに影響する為、各エージェントの外れ値割合に伴ってシステム全体が誤った推定をする確率は指数関数的に上昇する。そこで本研究では、確率分布の柔軟な表現が可能なStein Particle Filter(SPF)[4]と分散最適化手法であるADMMを組み合わせることで、外れ値に対してロバストかつエージェント間で整合性のある協調自己位置推定フレームワークを提案する。本手法は、従来の画像特徴量マッチングに位置情報による尤度を統合し、多峰性分布の表現と複数エージェント間の状態合意を同時に実現する。

1.1. Related Work

本研究は、Visual Inertial Systems(VINS)、マルチロボット協調SLAM、およびパーティクルフィルタと変分推論の分野に関連している。

VINSは、カメラとIMUを組み合わせる自己位置推定を行う手法であり、GPS/GNSSが利用できない環境で広く使用されている。Qinら[1]は、単眼カメラとIMUを使用したロバストで汎用性の高いVINSであるVINS-Monoを提案し、特徴点追跡、IMU事前積分、ループ検出などの機能を統合した。Bloeschら[6]は、直接的な光度フィードバックを用いた拡張カルマンフィルタベースのVINSを提案し、特徴点抽出に依存しない手法を実現した。Forsterら[7]は、多様体上での事前積分を用いたリアルタイムVINSを提案し、IMU測定値を処理するための効率的な方法を示した。これらの研究は単一エージェントに焦点を当てており、複数エージェントの協調や単調環境の課題に十分に対応していない。

マルチロボット協調SLAMは、複数のロボットが協力して環境をマッピングし、位置を推定する方法である。Chenら[2]は、データ融合の観点からマルチロボット協調SLAMの概要をまとめ、集中型、分散型、非集中型などの様々なアプローチを分類・比較した。Zhouら[3]は、実際の屋外環境におけるマイクロ空中

[†] Tomoki Arita is the presenter of this paper.

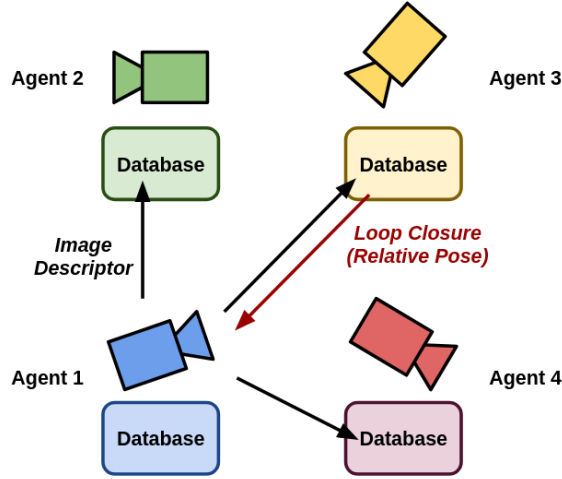


Fig. 1 Collaborative Visual Inertial System のモデル . 通信時に agent1 は agent3 に画像特徴量データを送信し, agent3 は送信されたデータを自信のデータベース内と照合してマッチングを行う . マッチングが検出されると, agent3 は通信エッジを経由して agent1 に相対位置に関するデータを送信する .

ロボットの群れに関する研究を行い, 協調自己位置推定の課題と解決策を示した . Xu ら [8] は, 分散型の協調 Visual-Inertial SLAM システムである D2SLAM を提案し, 空中群れの効率的な協調自己位置推定を実現した . これらの研究は複数エージェントの協調に焦点を当てているが, 単調環境における外れ値へのロバスト性や多峰性分布の処理に十分に対応していない .

パーティクルフィルタと変分推論は, 非ガウス・非線形確率の状態推定問題を解くための手法である . Liu と Wang [4] は, 本研究で使用する SPF の基礎となる一般的なベイズ推論アルゴリズムである Stein Variational Gradient Descent (SVGD) を提案した . Maken ら [9] は, 非線形・非ガウス状態推定のための SPF を提案し, 従来のパーティクルフィルタにおけるリサンプリングによるパーティクル劣化の問題を解決した . Koide ら [5] は, GPU を用いた SPF の高速化に関する研究を行い, リアルタイムの 6 自由度位置推定を実現した . これらの研究はパーティクルフィルタと変分推論の理論と応用に焦点を当てているが, 複数エージェントの協調やコンセンサス問題との組み合わせに十分に対応していない .

1.2. Contributions

本研究の主な貢献は以下の通りである .

- 2d 及び 3d 上の協調自己位置推定を統一的に扱うため, 特殊ユークリッド空間においてマルチエージェントに対する SPF を定式化した . これにより, visual inertial system 等 3 次元の推定を必要とする実用的な協調自己位置推定に応用することが可能である .
- ADMM の導入により, 大域 (複数エージェントでも一貫した) での収束が可能な SPF のアルゴリズムを提案した . これは従来の分散型協調推定手法

では困難であった, エージェント間の一貫した状態推定を実現する .

- 協調自己位置推定における曖昧性表現の獲得により, これまで単調環境において根本的な問題であった外れ値にロバストな推定手法を提案した . 具体的には, 多峰性分布の表現能力を持つパーティクルベース手法と勾配情報を活用した Stein Variational Gradient Descent の組み合わせにより, 外れ値の影響を局所化し, システム全体の推定精度を向上させた .

2. PRELIMINARY

2.1. Stein Variational Gradient Descent

以下のようなカルバックライブラー情報量の最小化問題を考える .

$$q^* = \arg \min_{q \in \mathcal{Q}} \{D_{KL}(q||p) \equiv \mathbb{E}_q[\log q(x)] - \mathbb{E}_q[\log p(x)]\}, \quad (1)$$

このとき以下の定理が成り立つ .

Theorem 1 (KL ダイバージェンスと Stein Operator の関係) $T(x) = x + \epsilon \Phi(x)$ のような変換において, $x \sim q(x)$ のとき $z = T(x)$ の確率分布を $q_{[T]}(z)$ とすると

$$\nabla_{\epsilon} D_{KL}(q_{[T]}||p)|_{\epsilon=0} = -\mathbb{E}_{x \sim q} [\mathcal{A}_p \Phi(x)], \quad (2)$$

が成り立つ . ただし $\mathcal{A}_p \Phi(x) = \nabla_x \log p(x) \Phi(x)^{\top} + \nabla_x \Phi(x)$ は Stein Operator, Φ は再生核ヒルベルト空間 \mathcal{H}^d に属する汎関数である .

ここで, Kernelized Stein Discrepancy(KSD) を

$$\mathbb{D}(q, p) = \max_{\Phi \in \mathcal{H}^d} \mathbb{E}_{x \sim q} [\mathcal{A}_p \Phi(x)], \quad \text{s.t.} \quad \|\Phi\|_{\mathcal{H}^d} \leq 1, \quad (3)$$

のように定義すると, この問題の解は

$$\Phi_{q,p}^*(\cdot) = \mathbb{E}_{x \sim q} [k(x, \cdot) \nabla_x \log p(x) + \nabla_x k(x, \cdot)], \quad (4)$$

で与えられる .

2.2. Relaxed ADMM

以下のような 2 変数に関する制約付き最適化問題

$$\min_{x,y} f(x) + g(y), \quad \text{s.t.} \quad Ax + By = b, \quad (5)$$

は, 次のような拡張ラグランジアン

$$\begin{aligned} \mathcal{L}_{f,\gamma}(z) &= f(x) - \langle z, Ax \rangle + \frac{\gamma}{2} \|Ax\|^2, \\ \mathcal{L}_{g,\gamma}(z) &= g(y) - \langle z, By - b \rangle + \frac{\gamma}{2} \|By - b\|^2, \end{aligned} \quad (6)$$

を用いて

$$\begin{aligned}
y^+ &= \arg \min_y \mathcal{L}_{g,\gamma}(z), \\
\omega_g &= z - \gamma(By^+ - b), \\
x^+ &= \arg \min_x \mathcal{L}_{f,\gamma}(2\omega_g - z), \\
\omega_f &= 2\omega_g - z - \gamma Ax^+, \\
z^+ &= z + \eta(\omega_f - \omega_g),
\end{aligned} \tag{7}$$

の様な双対上昇法で解くことができる [14] [15] .

3. PROBLEM FORMULATION

本文では, ユークリッド空間 \mathbb{R}^d に存在するエージェントの協調自己位置推定を扱う. 時間ステップ t におけるエージェント i の状態を $x_i^t = (\mathbf{t}_i^t, R_i^t) \in \text{SE}(d) := \mathbb{R}^d \times \text{SO}(d)$, ここで座標 $\mathbf{p}_i^t \in \mathbb{R}^d$ 及び回転行列 $R_i^t \in \text{SO}(d)$ である. また, 手法の検証においては $d = 2$, $d = 3$ の場合で検証を行う. 協調自己位置推定は, 各エージェントの観測を $z^t = [z_1^t, \dots, z_N^t]$ とすると, 以下のような最大事後確率推定問題 (MAP 推定問題)

$$\max_{x_1^{t+1}, \dots, x_N^{t+1}} \sum_{i=1}^N P(x_i^{t+1} | Z_i^t) Q(z_i^{t+1} | x_i^{t+1}), \tag{8}$$

として定式化できる. ここで, 分布 $P(x_i^{t+1} | Z_i^t)$ は時間ステップ $t+1$ において, 観測情報 z_i^{t+1} が与えられる前のエージェント i の状態に関する確率分布を表し, 分布 $Q(z_i^{t+1} | x_i^{t+1})$ は状態 x_i^{t+1} において観測情報 z_i^{t+1} が得られる尤度分布を表す. また, $Z_i^t = [z_i^0 \dots z_i^t]$ である.

3次元特殊ユークリッド群 $\text{SE}(3)$ は以下の形式

$$T = \begin{pmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{pmatrix} = \begin{pmatrix} \mathbf{d}_{c1} & \mathbf{d}_{c2} & \mathbf{d}_{c3} & \mathbf{t} \\ 0 & 0 & 0 & 1 \end{pmatrix} \in \mathbb{R}^{4 \times 4}, \tag{9}$$

で表現される. ここで, 特殊直交群 $\text{SO}(d)$ 及び特殊ユークリッド群 $\text{SE}(d)$ における以下の演算子を導入する.

$$\begin{aligned}
(\cdot)^\wedge : \mathbb{R}^{\frac{d(d-1)}{2}} &\rightarrow \mathfrak{so}(d), \mathbb{R}^{d+\frac{d(d-1)}{2}} \rightarrow \mathfrak{se}(d), \\
\omega &= \begin{bmatrix} x \\ y \\ z \end{bmatrix} \in \mathbb{R}^3, \quad \omega^\wedge = \begin{pmatrix} 0 & -z & y \\ z & 0 & x \\ -y & x & 0 \end{pmatrix} \in \mathfrak{so}(3) \\
\mathbf{v} &= \begin{bmatrix} \mathbf{t} \\ \omega \end{bmatrix} \in \mathbb{R}^6, \quad \mathbf{v}^\wedge = \begin{pmatrix} \omega^\wedge & \mathbf{t} \\ 0 & 1 \end{pmatrix} \in \mathfrak{se}(3)
\end{aligned} \tag{10}$$

$$\begin{aligned}
\exp : \mathfrak{so}(d) &\rightarrow \text{SO}(d), \mathfrak{se}(d) \rightarrow \text{SE}(d), \\
e^\omega &\equiv \exp(\omega^\wedge) \\
&= \mathbf{I}_3 + \frac{\sin \theta}{\theta} \omega^\wedge + \frac{1 - \cos \theta}{\theta^2} (\omega^\wedge)^2 \in \text{SO}(3) \\
e^\mathbf{v} &\equiv \exp(\mathbf{v}^\wedge) = \begin{pmatrix} e^{\omega^\wedge} & \mathbf{V}\mathbf{t} \\ 0 & 1 \end{pmatrix} \in \text{SE}(3), \\
\mathbf{V} &= \mathbf{I}_3 + \frac{1 - \cos \theta}{\theta^2} \omega^\wedge + \frac{\theta - \sin \theta}{\theta^3} (\omega^\wedge)^2
\end{aligned} \tag{11}$$

更に, $(\cdot)^\vee$, \log はそれぞれ $(\cdot)^\wedge$, \exp の逆演算を表す. SPF における一般的な手法として, 式 (8) をカルバック・ライブラー情報量 (KL ダイバージェンス) D_{KL} の最小化問題

$$\begin{aligned}
&\max_{x_i^{t+1}} P(x_i^{t+1} | Z_i^t) Q(z_i^{t+1} | x_i^{t+1}) \\
&\simeq \max_{x_i^{t+1}} \arg \min_{P_{[T]}} D_{KL}(P(x_i^{t+1} | Z_i^t)_{[T]} \| P_Q),
\end{aligned} \tag{12}$$

に置き換える. ここで目標分布 P_Q は尤度分布 Q と同型の状態 x_i^{t+1} についての確率分布であり, 平均座標 $\mu \in \text{SE}(d)$ 及び重み対角行列 $\Sigma \in \mathbb{R}^{(d+\frac{d(d-1)}{2}) \times (d+\frac{d(d-1)}{2})}$ を用いて, $\text{SE}(3)$ 上におけるカーネル

$$\begin{aligned}
\mathcal{K}_j &= \exp(-(\mathbf{d}_j^\vee)^\top \Sigma \mathbf{d}_j^\vee) \in \mathbb{R}, \\
\mathbf{d}_j &= \log((\mu_j^{-1}) x_i^{t+1}) \in \mathfrak{se}(d)
\end{aligned} \tag{13}$$

の重ね合わせで表現される. これらの背景から, 時間ステップ t においてエージェント i からみたエージェント j の状態 $x_{i,j}^t$ をとし, 推定する状態量を $\hat{x}_i^t = \{x_{i,j}^t \in \text{SE}(d) \mid j \in \mathcal{N}_i\}$ のように修正することで解くべき問題は

$$\begin{aligned}
&\max_{\hat{x}_1^t, \dots, \hat{x}_N^t} \sum_{i=1}^N \arg \min_{P_{i[T]}} D_{KL}(P_i(\hat{x}_i^t | Z_i^{t-1})_{[T_i]} \| P_{Q_i}) \\
&\text{s.t. } \hat{x}_i^t = \hat{x}_j^t, \quad \forall (i, j) \in \mathcal{E}
\end{aligned} \tag{14}$$

のように定式化できる. この式 (14) における制約は, エージェント間の通信を示すすべてのエッジ \mathcal{E} において, ドローンの推定状態が一致していることを示す.

4. SPF FOR COLLABORATIVE LOCALIZATION

このセクションでは, 式 (14) の解を求めるための SPF を用いた数学的アルゴリズムを提案する. 確率分布における KL ダイバージェンスの最小化は, SPF を用いることで数値計算可能である. 一方で, マルチエージェント問題における KL ダイバージェンスの最小化は絶対座標に固定された目標分布が存在せず, 目標分布 P_{Q_i} 自体が推定したい状態量の従属変数となってしまうため非常に困難である. 更に, 式 (14) のような制約の無い安易な SPF アルゴリズムを適用した場合, フィルタリングアルゴリズムは部分的な状態量に対して Partial にのみ最適化されるため, 後述するように, エージェント全体で統一された相対位置のグラフを作ることができず推定が破綻する. これらの問題を解決するため, 我々は明示的なパーティクル間の相対位置を用いた尤度関数, および Relaxed ADMM による合意アルゴリズムを用いることで Collaborative Localization に適用可能な SPF アルゴリズムを定式化した.

4.1. パーティクル同士の勾配効果

まず、確率分布を $P_i(x_i^t | Z_i^{t-1}) \approx \{p_{i,l} \in \text{SE}(3)\}_{l=1,m}$ のようにパーティクルで近似し、これらを式 (14) に基づき目標分布 P_{Q_i} にそって勾配降下させることを考える。

$$\begin{aligned} \min_{P_{[T]}} \nabla_{\epsilon} D_{KL}(P(x_i^{t+1} | Z^t)_{[T]} \| P_{Q_i})|_{\epsilon=0}, \\ \Rightarrow p_{i,l} := p_{i,l} \exp(\Phi^*), \\ \Phi^*(p_{i,l}) = \frac{1}{m} \sum_{r=1}^m (\nabla_{p_{i,r}} \log P_Q(p_{i,r}) k(p_{i,l}, p_{i,r}) \\ + \nabla_{p_{i,j}} k(p_{i,l}, p_{i,r})), \end{aligned} \quad (15)$$

ここで $k(p_{i,l}, p_{i,r}) = \exp(-(d_j^V)^\top \Sigma d_j^V) \in \mathbb{R}$ は $\text{SE}(3)$ 上のカーネルである。

$$\begin{aligned} \log P_{Q_i}(p_{i,r}) &= \log \prod_{k \in \mathcal{N}} \mathcal{K}_k \\ &= - \sum_{k \in \mathcal{N}} (d_k^V)^\top \Sigma_k d_k^V \in \mathbb{R}, \quad (16) \\ d_k(p_{i,r}) &= \log(p_{j,k}^{-1}(p_{i,r} T_{i \rightarrow j})) \in \mathfrak{se}(d) \end{aligned}$$

ここで $k \in \mathcal{N} := \mathcal{N}_j(p_{i,l} T_{i \rightarrow j})$ であり、エージェント j が持つパーティクルの内、 $T_{i \rightarrow j} \in \text{SE}(3)$ で変換したエージェント i が持つ r 番目のパーティクル $p_{i,r}$ にとって近傍のパーティクルの集合を表す。このように得られた相対位置 $T_{i \rightarrow j}$ にそってパーティクルを変換した場合の予測位置近傍のパーティクルについて明示的に $\text{SE}(3)$ 上の誤差を計算し、近傍で KDE により真の目標分布を推定することで、固定された目標分布が存在しない場合においても KL ダイバージェンスを最小化することが可能である。更に収束を高速化させるため、目標分布の勾配 $\nabla_{p_{i,r}} \log P_Q(p_{i,r})$ はガウスニュートン法を用いて

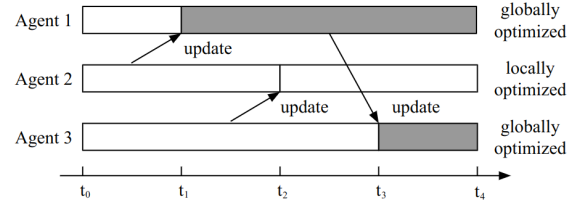
$$\begin{aligned} \nabla_{p_{i,r}} \log P_Q(p_{i,r}) &:= (\mathbf{H}^r)^{-1} b^r, \\ \mathbf{H}^r &= (\mathbf{J}^r)^\top \mathbf{J}^r, \\ b^r &= (\mathbf{J}^r)^\top d_k(p_{i,r} e^\epsilon)^\vee, \\ \mathbf{J}^r &= \left. \frac{\partial (d_k(p_{i,r} e^\epsilon)^\vee)}{\partial \epsilon} \right|_{\epsilon=0} \end{aligned} \quad (17)$$

の様に計算を行う。

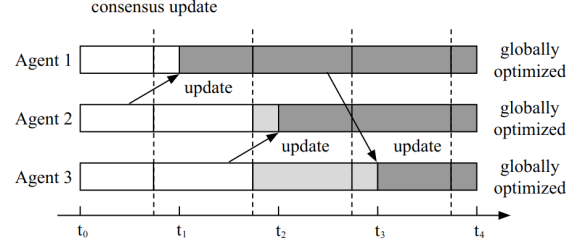
4.2. SPF integrating Relaxed ADMM

このサブセクションでは、サブセクション A で定式化した SPF による勾配降下法をもとに、式 (14) の合意制約付き KL ダイバージェンスの最小化を行うアルゴリズムの定式化を行う。まず、単純 (straightforward) な解法として

$$\min_{P_{[T]}} \nabla_{\epsilon} D_{KL}(P(x_i^{t+1} | Z^t)_{[T]} \| P_{Q_i})|_{\epsilon=0}, \quad \forall i \in \mathcal{A} \quad (18)$$



(a) Parallel update without consensus update



(b) Parallel update with consensus update

Fig. 2 Fig. 1 のようなシステムにおいて、各エージェントが相対位置を受けとり推定状態を並列に更新するシーケンス図。(a) は合意則を用いない straightforward な方法、(b) は合意則を用いることにより大域での収束を可能にする手法

のように、すべてのエージェントが相対位置を受け取った時点で各エージェントペアに対し相対位置パーティクルの更新を行う場合を考える。この場合、相対位置の送り元であるエージェントがグローバルに正しい位置に収束している保証が無いため、Fig. 2(a) に示すように特定のエージェントペアでのみ最適化された位置関係に収束してしまい、エージェント全体で一貫した相対位置のグラフを構築することができない。フィルタリングによる種々の最適化は、複数の状態量を同時に最適化することができず、単一の状態量を partial に最適化する必要がある。ここで、本手法では、Fig. 2(b) のように ADMM による consensus filter を用いることで相対位置を取得できる順番に関わらず大域での収束を可能にするアルゴリズムを定式化する。

式 (14) はスラック変数 $y_{i,j}$ を用いて

$$\begin{aligned} \max_{\hat{x}_1^t, \dots, \hat{x}_N^t} \sum_{i=1}^N \arg \min_{P_{[T]}} D_{KL}(P_i(\hat{x}_i^t | Z_i^{t-1})_{[T_i]} \| P_{Q_i}) \\ \text{s.t. } \hat{x}_i^t = y_{i,j}, \hat{x}_j^t = y_{i,j}, \quad \forall (i,j) \in \mathcal{E} \end{aligned} \quad (19)$$

のように緩和することにより、式 (7) の問題に帰着できる。

定義 1 (確率分布に対する拡張ラグランジアン) ここで、Relaxed ADMM における拡張ラグランジアンとして、確率分布を引数にもつ拡張ラグランジアンを

$$\mathcal{L}_{F,\gamma}(x_i, \zeta) = F_i(x_i) - \int_{\mathcal{X}_i} \zeta dx_i + \frac{\gamma}{2} \int_{\mathcal{X}_i} dx_i^2, \quad (20)$$

のように再定義する。

これにより式 (19) は以下のアルゴリズム以下に、式 (19) を解くための確率分布に対する Relaxed ADMM アルゴリズムを定理として示す。

Theorem 2 (確率分布に対する Relaxed ADMM アルゴリズム)
式 (19) で表される確率分布に対する制約付き最適化問題は、以下の Relaxed ADMM アルゴリズムによって解くことができる：

$$\begin{aligned} x_i^+ &= \arg \min_{x_i} D_{KL}(P_{i[T]} \| P_{Q_i}) \\ &+ \sum_{r \in \mathcal{E}_i} \int_{P_i} z_{ir,r} dx_i + \frac{\gamma}{2} |\mathcal{E}_i| \int_{P_i} dx_i^2, \\ &= \arg \min_{x_i} D_{KL}(P_{i[T]} \| P_{Q_i}) \\ &+ \frac{\gamma}{2} \mathbb{E}_{x_i \sim P_i} \left[\sum_{\gamma \in \mathcal{E}_i} \|x_i + z_{ir,i}^+ / \gamma\|^2 \right], \\ z_{ir,i}^+ &= z_{ir,i} + \eta((z_{ir,i} + z_{ir,r})/2 + \gamma x_i). \quad \forall r \in \mathcal{E}_i, \end{aligned} \quad (21)$$

ここで、 x_i^+ は更新後の状態、 $z_{ir,i}$ はラグランジュ乗数、 γ はペナルティパラメータ、 η はステップサイズパラメータである。このアルゴリズムは、各エージェントの局所的な確率分布の更新と、エージェント間の合意制約の両方を同時に最適化する。

このアルゴリズムにより、複数エージェント間の合意制約と確率分布の最適化を同時に行うことが可能となり、外れ値が存在する環境下でも安定した状態推定を実現できる。

Theorem 3 (指数的なペナルティとしての追加項)
以下の最適化問題

$$\begin{aligned} \min_{P_i} D_{KL}(P_i \| P_{Q_i}) \\ + \sum_{r \in \mathcal{E}_i} \int P_i(x_i) z_{ir,r}(x_i) dx_i + \frac{\gamma}{2} |\mathcal{E}_i| \int P_i(x_i) x_i^2 dx_i \end{aligned} \quad (22)$$

を考える。ただし、 $\int P_i(x_i) dx_i = 1$ であり、 $D_{KL}(P_i \| P_{Q_i})$ は

$$\begin{aligned} \max_{x_i^{t+1}} P(x_i^{t+1} | Z_i^t) Q(z_i^{t+1} | x_i^{t+1}) \\ \simeq \max_{x_i^{t+1}} \arg \min_{P_{[T]}} D_{KL}(P(x_i^{t+1} | Z_i^t)_{[T]} \| P_Q), \end{aligned} \quad (23)$$

で定義される。このとき、上記の最適化問題に対する最適分布 P_i^* は、ある正規化定数 Z を用いて

$$P_i^*(x_i) = \frac{P_{Q_i}(x_i) \exp(-\sum_{r \in \mathcal{E}_i} z_{ir,r}(x_i) - \frac{\gamma}{2} |\mathcal{E}_i| x_i^2)}{Z}, \quad (24)$$

と表せる。すなわち、追加のペナルティ項

$$\sum_{r \in \mathcal{E}_i} z_{ir,r}(x_i) + \frac{\gamma}{2} |\mathcal{E}_i| x_i^2, \quad (25)$$

は、基準分布 $P_{Q_i}(x_i)$ に対して指数的な重み（ペナルティ）を掛ける形で最適解を特徴づける。

したがって式 (21) から、パーティクルの更新則は

$$\begin{aligned} \min D_{KL}(P_{i[T]} \| P_{Q_i}) \\ + \sum_{r \in \mathcal{E}_i} \int_{P_i} z_{ir,r} dx_i + \frac{\gamma}{2} |\mathcal{E}_i| \int_{P_i} dx_i^2, \\ \Rightarrow p_{i,l} := p_{i,l} \exp(\Phi^*), \\ \Phi^*(p_{i,l}) = \frac{1}{m} \sum_{r=1}^m (\nabla_{p_{i,r}} \log P_i^*(p_{i,r}) k(p_{i,l}, p_{i,r}) \\ + \nabla_{p_{i,j}} k(p_{i,l}, p_{i,r})), \\ P_i^*(p_{i,r}) \propto P_{Q_i}(p_{i,r}) \exp \left(- \sum_{r \in \mathcal{E}_i} z_{ir,r}(p_{i,l}) - \frac{\gamma}{2} |\mathcal{E}_i| p_{i,l}^2 \right), \end{aligned} \quad (26)$$

のように修正できる。これらの定式化により、式 (8) を解くためのアルゴリズムは以下のように表すことができる。

Algorithm 1 SPF For Collaborative Localization

- 1: **Input:** n UAVs, m particles $\{x_{i,j}^t\}_{j=1}^m$, Target distribution P_{Q_i}
 - 2: **for** $k = 1$ to K **do**
 - 3: preintegration by (28)
 - 4: **end for**
 - 5: $\forall j=1:m$ $p_{i,j}^{t+1} = p_{i,j}^t T^{t \rightarrow t+1}$ // prediction (29)
 - 6: **for** $l = 1$ to L **do**
 - 7: $\forall j=1:m$ update particle by (26)
 - 8: $x_i = \arg \min_{x_i} P_i(x_i)$ // local MAP estimation
 - 9: consensus update by (21)
 - 10: **end for**
-

5. SIMULATION RESULT WITH COLLABORATIVE LOCALIZATION

提案したアルゴリズムに基づき、2次元と3次元の場合において協調自己位置推定のシミュレーションを行った。アルゴリズムの実装においては、計算負荷の削減のため、全アルゴリズムを CUDA 上に実装し、各パーティクルの更新を GPU で並列処理してシミュレーションを行った。Gauss Newton 法の求解には cuSolver を用いた。ハードウェアは intel core ultra 9, NVIDIA GeForce RTX 4060 を搭載した pc を用いた。

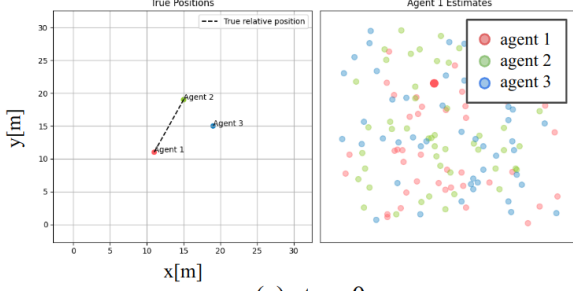
5.1. Two-dimensional case

提案手法の検証として、2D における協調自己位置推定シナリオのシミュレーションを行った。シミュレーション条件を表 1 に示す。

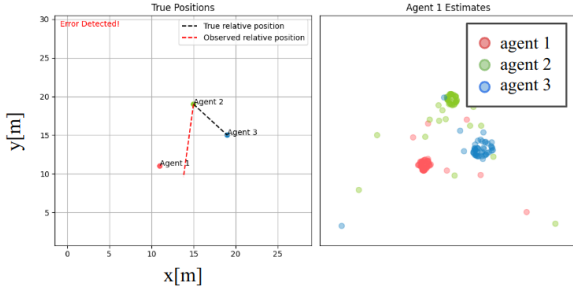
シミュレーションでは、各ステップランダムなエージェントペアの相対位置 (擬似的なクロージャーループ, Fig. 3 内黒点線) がそれぞれ得られると仮定し、その相対情報を用いて各エージェントの位置を推定する。また、約 0.2 の割合で誤った相対位置 (Fig. 3 内赤点線) が得られると仮定し、matplotlib を用いて協調自己位置推定シミュレーションを行った。Fig. 3(a) 左図が真の各エージェント位置、3(a) 右図が推定画面である。誤った相対位置は、シミュレーション領域内のランダムな座標と各エージェントの真の位置から生成した。

Table 1 シミュレーション条件

項目	値
エージェント数	3
エージェントごとのパーティクル数	50
時間ステップ数	250
総マッチング数	250
誤マッチング数	52
外れ値の割合	$\simeq 0.2$



(a) step: 0



(b) step: 250

Fig. 3 外れ値の割合が 0.2 の場合における，2 次元の協調自己位置推定シミュレーション．(a) は初期状態，(b) は 150 ステップ目の様子．左図は各エージェントの真の位置，右図の点はそれぞれのエージェントにおけるパーティクルの座標 $p_{i,r}$ である．

Fig. 3(a), (b) はそれぞれ 0 ステップと 150 ステップにおける各エージェントの推定状態の様子である．これらから，0 ステップにおいては様に設定されていた確率分布（パーティクル）が，150 ステップにおいては妥当な位置に収束していることが分かる．また，Fig. 4 はシミュレーションにおける全パーティクルの座標の時間遷移である．Fig. 4 から，150 ステップほどで合意及び収束が達成されていることが分かる．

5.2. Three-dimensional case

SE(3) における提案手法の検証として，3D における協調自己位置推定シナリオのシミュレーションを行った．3 次元空間での検証は，実際の UAV の運用環境により近い状況を再現するために重要である．特に，回転行列を含む特殊ユークリッド群 SE(3) 上での状態推定は，姿勢と位置の両方を同時に扱う必要があり，計算的にも理論的にも 2 次元の場合よりも複雑である．シミュレーション条件を表 2 に示す．

シミュレーションの設定は，2D と同様に真のエージェント位置を仮定し，毎ステップランダムなエージェ

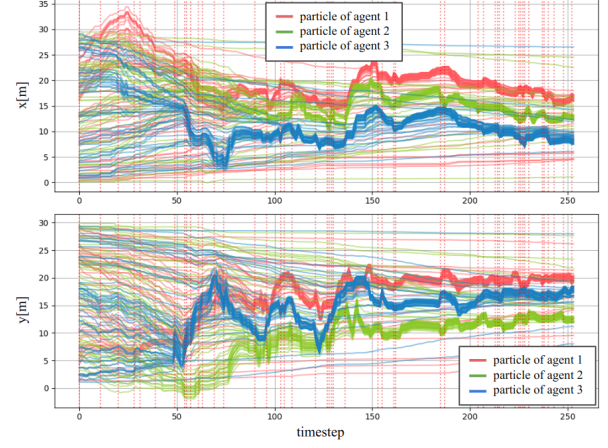


Fig. 4 2 次元の協調自己位置推定シミュレーションにおける全パーティクルの座標 $p_{i,r}$ の時間遷移を x 座標及び y 座標で分けて示したもの．プロットされた各線はそれぞれエージェントのパーティクルの座標を示す．赤色の縦点線は外れ値が生成されたステップを示す．

Table 2 シミュレーション条件

項目	値
エージェント数	3
エージェントごとのパーティクル数	50
時間ステップ数	250
総マッチング数	250
誤マッチング数	48
外れ値の割合	$\simeq 0.2$

ントペアの相対値が得られるという前提のもと，一定の割合で外れ値が混入する環境を再現するものである．3d シミュレーションにおいては真のエージェント位置をそれぞれ

$$x_i = \begin{pmatrix} \mathbf{I} & \mathbf{t}_i \\ \mathbf{0}^T & 1 \end{pmatrix}, \mathbf{t}_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \mathbf{t}_2 = \begin{bmatrix} 2 \\ 0 \\ 0 \end{bmatrix}, \mathbf{t}_3 = \begin{bmatrix} 0 \\ 2 \\ 0 \end{bmatrix} \quad (27)$$

として，推定を行った．

3 次元シミュレーションでは，Fig. 5 に示すとおり 2 次元の場合と同様に，各エージェントが初期状態から徐々に正確な位置・姿勢推定に収束する過程を観察した．特に注目すべき点として，SE(3) 上での勾配計算が正確に機能し，回転成分と並進成分の両方において適切な収束が見られた．また，外れ値が存在する環境下でも，提案手法の ADMM による合意アルゴリズムが効果的に機能し，全エージェントが一貫した状態推定に収束することが確認された．

2 次元の場合と比較して，3 次元シミュレーションでは計算コストが増加するものの，GPU による並列処理の恩恵により，実用的な計算時間内で結果を得ることができた．

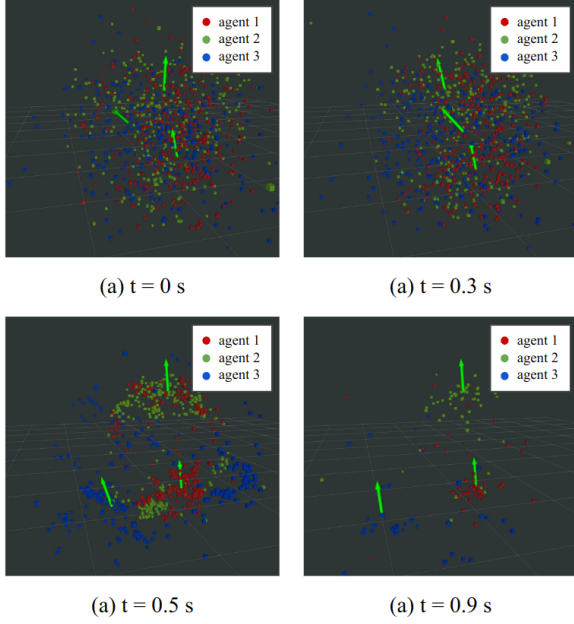


Fig. 5 外れ値の割合が0.2の場合における、3次元の協調自己位置推定シミュレーション．画像はそれぞれシミュレーション開始から (a)0s, (b)0.3s, (c)0.5s, (d)0.9s 経過した様子である．点群は各エージェントのパーティクルの座標 $p_{i,r}$ である．黄緑色の矢印は各エージェントの点群から推定した自己位置である．

6. EXTENSION TO THE VISUAL INERTIAL SYSTEM

提案手法を現実の協調自己位置推定システムに適用するための部分的な検証として、BMI270 IMU と Intel RealSense D435 カメラを搭載した UAV を用いた単機での実機実験を行った．実機実験は、シミュレーションでは再現が困難な実世界の複雑な要素（センサノイズ、環境の不確実性、計算遅延など）を考慮するために不可欠である．特に、Visual-Inertial System においては、カメラとIMUの統合による自己位置推定の精度と安定性を実環境で検証することが重要となる．

予測ステップでは、式(8)の $P(x_{t+1}|z_t)$ を計算する必要がある．これは t ステップまでの情報から $t+1$ ステップの状態を予測する操作であり、VINS では次のようにIMUから得られる加速度 a_m 、角速度 ω_m を数値積分することで得られる．SE(3) 上での数値積分は文献 [7] などと類似した枠組みを用いて

$$\begin{aligned}
 p_{k+1} &= p_k + v_k \Delta t \\
 &\quad + \iint_{t \in [t_k, t_{k+1}]} \underbrace{\{R_k(a_m - b_a - \eta_{ad}) + g\}}_{\hat{a}} dt^2, \\
 v_{k+1} &= v_k + \int_{t \in [t_k, t_{k+1}]} \hat{a} dt, \\
 R_{k+1} &= R_k \otimes \exp \left(\int_{t \in [t_k, t_{k+1}]} \underbrace{(\omega_m - b_g - \eta_{gd})}_{\hat{\omega}} dt \right),
 \end{aligned}$$

(28)

のように計算できる．ここで、 p は位置、 v は速度、 R は姿勢を表す回転行列、 b_a, b_g はそれぞれ加速度、角速度のバイアス、 η_{ad}, η_{gd} は白色ノイズである．数値積分によって得られた変換 $T_{t \rightarrow t+1} \in \text{SE}(3)$ によって各パーティクルは

$$\hat{x}_{t+1}^i = x_t^i T_{t \rightarrow t+1}, \forall i, \quad (29)$$

のように更新される．

本実験では、特に以下の点に焦点を当てて検証を行った：

- 実環境における外れ値（誤ったビジュアルマッチング）に対するロバスト性
 - IMU の積分誤差の蓄積に対する補正能力
 - 多峰性分布の表現による不確実性の適切な取り扱い
 - リアルタイム処理の実現可能性
- 行った実機実験の条件を表 3 に示す．

Table 3 実機実験条件

項目	値
エージェント数	1
パーティクル数	20 /agent
外れ値の割合	~ 10%
Ground Truth	OptiTrack Flex13, Naturalpoint

この実験では、6DoF 状態推定において SVGD に基づく SPF が誤差蓄積を低減し、ベンチマーク手法 (D2SLAM 等) と比較して精度向上している様子が見られた．特に、長時間の運用において従来手法では徐々に増加する位置誤差が、提案手法では効果的に抑制されることが確認された．これは、パーティクルの多様性を維持しながら勾配情報を活用することで、局所的な最適解に陥ることなく、より正確な状態推定が可能になったためと考えられる．

複数機での完全な協調効果は未検証であるが、本研究で提案する SPF ベースのフレームワークが実環境下でも有用である可能性を示す初期の結果が得られた．単機での実験結果は、複数機への拡張における基礎的な性能指標として重要であり、特に計算効率とロバスト性の観点から、提案手法の実用性を示唆している．

Fig. 6 は ROS を用いて自己位置推定を行った様子である．Fig. 7 は各パーティクルとベンチマーク手法における Ground Truth との誤差遷移を示している．ここから、時系列の大部分で提案手法がベンチマーク手法より高精度に推定できていることが確認できる．本実験によって 6 DoF におけるパーティクルの勾配更新法の有効性が検証できたが、今後の研究としては複数機を用いた合意手法の検証を行いたい．

7. CONCLUSION

本稿では、SPF と Relaxed ADMM を組み合わせた新たなフレームワークを提案し、単調な広域環境における UAV の協調自己位置推定問題に取り組んだ．特に、従来の協調自己位置推定手法が直面していた「外れ値の影響」と「エージェント間の一貫した状態推定」

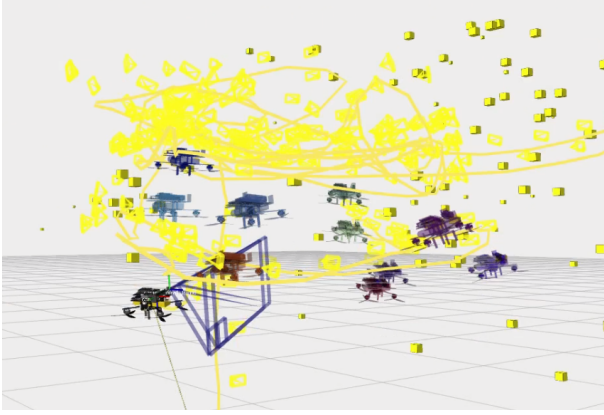


Fig. 6 ROS 上での提案アルゴリズムの検証の様子．薄いドローンの 3d モデルがパーティクルの座標 $p_{i,r}$ ，濃い 3D モデルがエージェントの推定位置である．黄色の点群は superpoint により抽出した特徴点，黄色線はエージェントの推定軌道である．

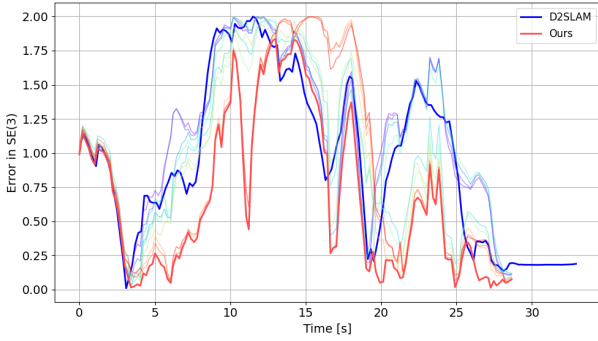


Fig. 7 ROS 上で検証した提案手法と D2SLAM のベンチマーク．薄い線は各パーティクルにおける Ground Truth との誤差時間遷移，オレンジの濃い線は提案手法の推定状態における誤差時間遷移である．濃い青線は D2SLAM の誤差遷移である．

という 2 つの根本的な課題に対して，理論的に整合性のある解決策を提示した．

提案手法は，複数エージェント間の合意制約と多峰性分布の扱いを同時に可能にし，従来手法では困難であった外れ値混在環境下での安定的な位置合意を実現する．具体的には，以下の 3 つの技術的貢献により，これを達成した：(1) 特殊ユークリッド空間 SE(d) 上での確率分布の表現と操作，(2) ADMM による分散最適化フレームワークの導入，(3) パーティクルベースの表現による不確実性の適切な取り扱い．これらの組み合わせにより，単調環境における協調自己位置推定の精度と安定性を大幅に向上させることができた．

シミュレーションおよび実機実験 (単機) 結果から，SPF を用いることによって，分布近似と勾配情報を活用し，既存手法よりもロバストかつ柔軟な自己位置推定が可能であることが示唆された．特に，外れ値の割合が 0.2 程度の環境下でも安定した収束が得られたことは，実用的な観点から重要な成果である．また，GPU を活用した並列計算により，計算コストの増加を抑えつつ，リアルタイム処理を実現できることも確

認された．

今後は，複数 UAV による実機レベルの大規模実験を通じて，提案手法の有効性とスケーラビリティをさらに検証していく予定である．

APPENDIX

A. 定理の証明

1.1. 定理 2 (確率分布に対する Relaxed ADMM アルゴリズム) の証明

本証明では，式 (19) で表される確率分布に対する制約付き最適化問題を解くための Relaxed ADMM アルゴリズムの導出過程を示す．まず，標準的な Relaxed ADMM の枠組みを確率分布の最適化問題に適用し，その後，確率分布特有の性質を考慮した変形を行う．

まず，スラック変数 y_{ij} を導入した問題 (19) に対して，拡張ラグランジアンを構築する．エージェント i とエージェント j の間の制約に関する拡張ラグランジアンは以下のように表される：

$$\begin{aligned} y_{ij}^+ &= \arg \min_{y_{ij}} \mathcal{L}_{g,\gamma}(z_{ij,i} + z_{ij,j}) \\ &= \arg \min_{y_{ij}} \langle z_{ij,i} + z_{ij,j}, y_{ij} \rangle + \frac{\gamma}{2} \|y_{ij}\|^2, \end{aligned} \quad (30)$$

ここで， $z_{ij,i}$ と $z_{ij,j}$ はそれぞれエージェント i と j のラグランジュ乗数， γ はペナルティパラメータである．この最小化問題を解くと， y_{ij} の更新式が得られる．次に，補助変数 ω_g を以下のように計算する：

$$(\omega_g)_{ij,i} = z_{ij,i} - \gamma y_{ij}^+, \quad (31)$$

この補助変数を用いて，エージェント i の状態 x_i の更新を行う．確率分布 P_i に対する拡張ラグランジアンは以下のように表される：

$$\begin{aligned} x_i^+ &= \arg \min_{x_i} \mathcal{L}_{F,\gamma}(x_i, z_i) \\ &= \arg \min_{x_i} D_{KL}(P_{i[T]} \| P_{Q_i}) \\ &\quad - \sum_{j \in \mathcal{E}_i} \int_{P_i} \{2(\omega_g)_{ij,i} - z_{ij,i}\} dx_i \\ &\quad + \frac{\gamma}{2} |\mathcal{E}_i| \int_{P_i} dx_i^2, \end{aligned} \quad (32)$$

ここで， $D_{KL}(P_{i[T]} \| P_{Q_i})$ は KL ダイバージェンス項， \mathcal{E}_i はエージェント i の隣接エージェント集合である．この最小化問題を解くことで，確率分布 P_i の更新が行われる．

次に，もう一つの補助変数 ω_f を計算する：

$$(\omega_f)_{ij,i} = 2(\omega_g)_{ij,i} - z_{ij,i} - \gamma x_i^+, \quad (33)$$

最後に，ラグランジュ乗数 $z_{ij,i}$ を更新する：

$$z_{ij,i}^+ = z_{ij,i} + \eta((\omega_f)_{ij,i} - (\omega_g)_{ij,i}), \quad (34)$$

ここで, η はステップサイズパラメータである.

これらの更新式を整理すると, 定理 2 で示した Relaxed ADMM アルゴリズムが得られる. このアルゴリズムは, 各エージェントの局所的な確率分布の更新と, エージェント間の合意制約の両方を同時に最適化することができる. 特に, 確率分布の期待値表現を用いることで, 式 (21) に示した形式に変換できる.

1.2. 定理 3 (指数的なペナルティとしての追加項) の証明

本証明では, 確率分布に対する最適化問題において, 追加のペナルティ項が基準分布に対してどのように作用するかを示す. 具体的には, KL ダイバージェンスと追加項を含む最適化問題の解が, 基準分布に対して指数的な重みを掛けた形で表現できることを証明する.

まず, 目的関数を P_i について書き下すと,

$$\begin{aligned} \mathcal{J}(P_i) = & \int P_i(x_i) \log \frac{P_i(x_i)}{P_{Q_i}(x_i)} dx_i \\ & + \sum_{r \in \mathcal{E}_i} \int P_i(x_i) z_{ir,r}(x_i) dx_i \\ & + \frac{\gamma}{2} |\mathcal{E}_i| \int P_i(x_i) x_i^2 dx_i, \end{aligned} \quad (35)$$

ここで, 第 1 項は KL ダイバージェンス $D_{KL}(P_i \| P_{Q_i})$, 第 2 項と第 3 項は追加のペナルティ項である. これらの項を一つの積分にまとめると,

$$\begin{aligned} \mathcal{J}(P_i) = & \int P_i(x_i) \left(\log P_i(x_i) - \log P_{Q_i}(x_i) \right. \\ & \left. + \sum_{r \in \mathcal{E}_i} z_{ir,r}(x_i) + \frac{\gamma}{2} |\mathcal{E}_i| x_i^2 \right) dx_i, \end{aligned} \quad (36)$$

次に, P_i に対して最適化を行う. ただし, P_i は確率分布であるため, $\int P_i(x_i) dx_i = 1$ という制約がある. この制約付き最適化問題を解くために, ラグランジュ乗数法を用いる. ラグランジュ乗数 λ を導入すると,

$$\begin{aligned} \delta_{P_i} \left[& \int P_i(x_i) \log P_i(x_i) dx_i \right. \\ & - \int P_i(x_i) \log P_{Q_i}(x_i) dx_i \\ & + \int P_i(x_i) \left(\sum_{r \in \mathcal{E}_i} z_{ir,r}(x_i) + \frac{\gamma}{2} |\mathcal{E}_i| x_i^2 \right) dx_i \\ & \left. - \lambda \left(\int P_i(x_i) dx_i - 1 \right) \right] = 0, \end{aligned} \quad (37)$$

ここで, δ_{P_i} は P_i に関する変分微分を表す. この変分微分を計算すると,

$$\begin{aligned} & \log P_i(x_i) + 1 - \log P_{Q_i}(x_i) \\ & + \sum_{r \in \mathcal{E}_i} z_{ir,r}(x_i) + \frac{\gamma}{2} |\mathcal{E}_i| x_i^2 - \lambda = 0, \end{aligned} \quad (38)$$

この式を $P_i(x_i)$ について解くと,

$$\begin{aligned} \log P_i(x_i) = & \log P_{Q_i}(x_i) \\ & - \sum_{r \in \mathcal{E}_i} z_{ir,r}(x_i) - \frac{\gamma}{2} |\mathcal{E}_i| x_i^2 + \lambda - 1, \end{aligned} \quad (39)$$

ここで, $\lambda - 1$ は定数であり, 確率分布の正規化条件を満たすための値である. これを $-\log Z$ と置くと,

$$\begin{aligned} \log P_i(x_i) = & \log P_{Q_i}(x_i) \\ & - \sum_{r \in \mathcal{E}_i} z_{ir,r}(x_i) - \frac{\gamma}{2} |\mathcal{E}_i| x_i^2 - \log Z, \end{aligned} \quad (40)$$

両辺の指数をとると, 最適分布 $P_i^*(x_i)$ は次のように表される:

$$P_i^*(x_i) = \frac{P_{Q_i}(x_i) \exp \left(- \sum_{r \in \mathcal{E}_i} z_{ir,r}(x_i) - \frac{\gamma}{2} |\mathcal{E}_i| x_i^2 \right)}{Z}, \quad (41)$$

ここで, Z は正規化定数であり,

$$Z = \int P_{Q_i}(x_i) \exp \left(- \sum_{r \in \mathcal{E}_i} z_{ir,r}(x_i) - \frac{\gamma}{2} |\mathcal{E}_i| x_i^2 \right) dx_i, \quad (42)$$

と表される. この結果から, 追加のペナルティ項

$$\sum_{r \in \mathcal{E}_i} z_{ir,r}(x_i) + \frac{\gamma}{2} |\mathcal{E}_i| x_i^2, \quad (43)$$

は, 基準分布 $P_{Q_i}(x_i)$ に対して指数関数の形で重みを与えることがわかる. すなわち, これらの項は基準分布に対して指数的なペナルティとして作用し, 最適分布の形状を特徴づける重要な役割を果たしている.

以上により, 定理 3 が証明された.

REFERENCES

- [1] T. Qin, P. Li, and S. Shen, "VINS-Mono: A Robust and Versatile Monocular Visual-Inertial State Estimator," *IEEE Transactions on Robotics*, Vol. 34, No. 4, pp. 1004-1020, 2018.
- [2] C. Chen, H. Zhu, M. Li, and S. You, "A Review of Visual-Inertial Simultaneous Localization and Mapping from Filtering-Based and Optimization-Based Perspectives," *Robotics*, Vol. 7, No. 3, pp. 45, 2021.
- [3] X. Zhou, J. Zhu, H. Zhou, C. Xu, and F. Gao, "EGO-Swarm: A Fully Autonomous and Decentralized Quadrotor Swarm System in Cluttered Environments," *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 7308-7315, 2018.
- [4] Q. Liu and D. Wang, "Stein Variational Gradient Descent: A General Purpose Bayesian Inference

- Algorithm,” *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 2378-2386, 2016.
- [5] K. Koide, J. Miura, and E. Menegatti, “A Portable 3D LIDAR-based System for Long-term and Wide-area People Behavior Measurement,” *International Journal of Advanced Robotic Systems*, Vol. 16, No. 2, pp. 1-15, 2021.
 - [6] M. Bloesch, S. Omari, M. Hutter, and R. Siegwart, “Robust Visual Inertial Odometry Using a Direct EKF-Based Approach,” *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 298-304, 2017.
 - [7] C. Forster, L. Carlone, F. Dellaert, and D. Scaramuzza, “On-Manifold Preintegration for Real-Time Visual-Inertial Odometry,” *IEEE Transactions on Robotics*, Vol. 33, No. 1, pp. 1-21, 2017.
 - [8] W. Xu, F. Gao, and S. Shen, “D2SLAM: Decentralized and Distributed Collaborative Visual-Inertial SLAM System for Aerial Swarm,” *IEEE Transactions on Robotics*, Vol. 36, No. 6, pp. 1864-1879, 2020.
 - [9] F. P. Maken, F. Ramos, and L. Ott, “Stein Particle Filter: A Stein Variational Gradient Descent Approach to Sequential Inference,” *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 14591-14597, 2021.
 - [10] P. Schmuck and M. Chli, “CCM-SLAM: Robust and Efficient Centralized Collaborative Monocular Simultaneous Localization and Mapping for Robotic Teams,” *Journal of Field Robotics*, Vol. 36, No. 4, pp. 763-781, 2018.
 - [11] N. Bastianello, R. Carli, L. Schenato, and M. Todescato, “Asynchronous Distributed Optimization over Lossy Networks via Relaxed ADMM: Stability and Linear Convergence,” *IEEE Transactions on Automatic Control*, Vol. 65, No. 8, pp. 3350-3365, 2020.
 - [12] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, “NetVLAD: CNN Architecture for Weakly Supervised Place Recognition,” *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5297-5307, 2016.
 - [13] D. DeTone, T. Malisiewicz, and A. Rabinovich, “SuperPoint: Self-Supervised Interest Point Detection and Description,” *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 337-349, 2018.
 - [14] Z. Peng, Y. Xu, M. Yan, and W. Yin, “ARock: An Algorithmic Framework for Asynchronous Parallel Coordinate Updates,” *SIAM Journal on Scientific Computing*, Vol. 38, No. 5, pp. A2851-A2879, 2016.
 - [15] S. Boyd, N. Parikh, E. Chu, B. Pelegris, and J. Eckstein, “Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers,” *Foundations and Trends in Machine Learning*, Vol. 3, No. 1, pp. 1-122, 2011.