A Minimum Energy Filter for Distributed Multirobot Localisation [⋆]

Jack Henderson [*] **Jochen Trumpf** [*] **Mohammad Zamani** [**]

[*] *Research School of Electrical, Energy and Materials Engineering,*
Australian National University
E-mail: {jack.henderson, jochen.trumpf}@anu.edu.au
[**] *Land Division, Defence Science and Technology Group, Australia*
E-mail: mohammad.zamani@dst.defence.gov.au

Abstract: We present a new approach to the cooperative localisation problem by applying the theory of minimum energy filtering. We consider the problem of estimating the pose of a group of mobile robots in an environment where robots can perceive fixed landmarks and neighbouring robots as well as share information with others over a communication channel. Whereas the vast majority of the existing literature applies some variant of a Kalman Filter, we derive a set of filter equations for the global state estimate based on the principle of minimum energy filtering. We show how the filter equations can be decoupled and the calculations distributed among the robots in the network without requiring a central processing node. Finally, we provide a demonstration of the filter's performance in simulation.

Keywords: Nonlinear observers and filter design, Localization, Cooperative perception, Autonomous Mobile Robots, Multi-vehicle systems

1. INTRODUCTION

In a wide range of robotics applications, an accurate estimation of the current position and orientation (pose) of a robot is essential for its primary mission. However, in environments where sensor performance is degraded, traditional approaches to state estimation begin to fail, for example when GNSS systems are actively jammed or robots have very limited access to external landmarks.

Collaborative localisation (CL) is an approach that can be utilised where multiple robots are operating in the same environment. The core concept is that the information gained by sensors on one robot can be shared with other robots in the environment thereby increasing the accuracy of the pose estimates. This is useful in the aforementioned case of sensor degradation, and can also be used for groups of heterogeneous robots where robots have different sensing capabilities.

When information such as local state estimates are shared between robots in a filtering framework, the pose estimates of each robot are no longer independent. If this state-dependency is not properly accounted for, it can lead to data incest and over-confidence problems (Howard et al., 2003).

One solution to this problem is presented by Roumeliotis and Bekey (2002), who derive a centralised Extended Kalman Filter (EKF) which jointly estimates the pose of all robots in the network and where all of the information dependencies are tracked in a single covariance matrix. They go on to show that the joint filter equations can be decoupled into a set of smaller, communicating filters distributed among the robots in the network.

There are two drawbacks to this type of approach. Firstly, the entire joint covariance matrix must be tracked, either in a centralised or distributed arrangement. Secondly, after each measurement step, a robot must communicate to all other robots in the network to update the estimates and covariance terms that are tracked by other robots. This is often impractical in most scenarios as a robust, fully-connected communications network topology cannot be guaranteed.

Recent approaches to CL have focused on reducing the data that is tracked by each robot and relaxing the communication network topology constraints. For example, Carrillo-Arce et al. (2013) proposes a filter local to each robot which only tracks the local state. Cross-covariance terms are not tracked and are instead estimated locally when two robots meet and share information. While this reduces the communication overhead, it comes at the cost of being too conservative in the dependency estimation and not utilising available data to the maximal possible extent. Their experimental results show the performance of this type of filter is worse than the joint EKF. Further work by Luft et al. (2018) aims to better approximate the cross covariance terms, and demonstrates performance very similar to, although slightly worse than, the joint EKF from Roumeliotis and Bekey (2002).

The common element in these works is that they all utilise the Extended Kalman Filter and are all compared to the

[⋆] This research is supported by the Commonwealth of Australia as represented by the Defence Science and Technology Group of the Department of Defence and by the Australian Research Council Discovery Project DP190103615: "Control of Network Systems with Signed Dynamical Interconnections"

joint EKF presented by Roumeliotis and Bekey (2002) as it provides a baseline estimate given no restrictions on computation or communication. While the EKF and variants such as the Multiplicative EKF (MEKF) (Markley, 2003) are industry standard in terms of pose filtering algorithms, there are potentially better alternatives. All variants of the EKF rely on linearising the system, which can result in instability and convergence issues when estimating highly non-linear systems such as robot pose kinematics. More recently, the approach of minimum energy filtering has been demonstrated by Zamani et al. (2013) as a more accurate and robust algorithm for pose estimation. A collaborative minimum-energy pose estimation algorithm has been proposed by Zamani and Hunjet (2019), but opts for estimating cross-covariance terms rather than tracking the full state of the system.

In this paper, we present the derivation of a centralised geometric approximate minimum-energy (GAME) filter to estimate the poses of a network of robots using both interoceptive and exteroceptive measurements. We show how this filter can be equivalently derived as a set of collaborative filters which run locally on each robot in the network. The decoupled filters provide exactly the same pose estimates as the centralised filter and information-sharing is only necessary during the filter update step, where exteroceptive measurements are processed.

The filter we derive provides a baseline which can be used as a benchmark for future implementations of other minimum-energy filters, much in the same way that Roumeliotis and Bekey (2002) has been used as the baseline for further developments of EKF-based filters.

The remainder of this paper is organised as follows. In Section 2 we briefly introduce a number of concepts and the notation used in the paper. Section 3 formally states the problem we aim to solve and we present our solution in Section 4. We then demonstrate an implementation of the filter through a simulation in Section 5 and conclude the paper in Section 6.

2. PRELIMINARIES

In this section we introduce the notation and conventions used throughout the paper.

2.1 Notation

$(.)^\top$ denotes the matrix transpose, $\boldsymbol{I}_n$ denotes an $n \times n$ identity matrix. The operators exp and log denote the matrix exponential and matrix logarithm respectively.

We use the matrix Lie groups SO(3) to represent rotations and SE(3) to represent poses in homogeneous coordinates. The corresponding Lie algebras are $\mathfrak{so}(3)$ and $\mathfrak{se}(3)$, respectively.

$$\mathrm{SO}(3) = \left\{R \in \mathbb{R}^{3\times 3} \mid R^\top R = \boldsymbol{I}_3,\ \det R = 1\right\} \tag{1}$$

$$\mathrm{SE}(3) = \left\{X = \begin{bmatrix} R & p \\ 0_{1\times 3} & 1\end{bmatrix} \mid R \in \mathrm{SO}(3),\ p \in \mathbb{R}^3\right\} \tag{2}$$

$$\mathfrak{so}(3) = \left\{\Psi \in \mathbb{R}^{3\times 3} \mid \Psi^\top = -\Psi\right\} \tag{3}$$

$$\mathfrak{se}(3) = \left\{\Gamma = \begin{bmatrix} \Psi & v \\ 0_{1\times 3} & 0\end{bmatrix} \mid \Psi \in \mathfrak{so}(3),\ v \in \mathbb{R}^3\right\} \tag{4}$$

We define the following maps which allow us to switch between matrix and vector representations.

$$(.)_\times : \mathbb{R}^3 \to \mathfrak{so}(3) \qquad \omega_\times := \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0\end{bmatrix} \tag{5}$$

$$\mathrm{vex} : \mathfrak{so}(3) \to \mathbb{R}^3 \qquad \mathrm{vex}(\Omega) := \mathrm{vex}(\omega_\times) = \omega \tag{6}$$

$$(.)^\wedge : \mathbb{R}^6 \to \mathfrak{se}(3) \qquad \gamma^\wedge := \begin{bmatrix} \omega_\times & v \\ 0 & 0\end{bmatrix} = \Gamma \tag{7}$$

$$(.)^\vee : \mathfrak{se}(3) \to \mathbb{R}^6 \qquad \Gamma^\vee := (\gamma^\wedge)^\vee = \gamma \tag{8}$$

$$(.)^\curlyvee : \mathfrak{se}(3)^n \to \mathbb{R}^{6n} \qquad \boldsymbol{\Gamma}^\curlyvee = \left[\Gamma_1^{\vee\top}, \dots, \Gamma_n^{\vee\top}\right]^\top \tag{9}$$

$$(.)^\curlywedge : \mathbb{R}^{6n} \to \mathfrak{se}(3)^n \qquad \boldsymbol{\gamma}^\curlywedge := \left(\boldsymbol{\Gamma}^\curlyvee\right)^\curlywedge = \boldsymbol{\Gamma} \tag{10}$$

where

$$\gamma = \begin{bmatrix} \omega \\ v\end{bmatrix},\ \omega, v \in \mathbb{R}^3 \qquad \Gamma \in \mathfrak{se}(3) \tag{11}$$

$$\boldsymbol{\gamma} \in \mathbb{R}^{6n} \qquad \boldsymbol{\Gamma} = (\Gamma_1, \dots, \Gamma_n) \in \mathfrak{se}(3)^n \tag{12}$$

The following maps are useful when working in homogeneous coordinates.

$$\bar{(.)} : \mathbb{R}^3 \to \mathbb{R}^4 \qquad \bar{v} := \begin{bmatrix} v^\top & 1\end{bmatrix}^\top \tag{13}$$

$$\mathring{(.)} : \mathbb{R}^3 \to \mathbb{R}^4 \qquad \mathring{v} := \begin{bmatrix} v^\top & 0\end{bmatrix}^\top \tag{14}$$

$$\tilde{(.)} : \mathbb{R}^{3\times 3} \to \mathbb{R}^{4\times 4} \qquad \tilde{M} := \begin{bmatrix} M & 0 \\ 0 & 1\end{bmatrix} \tag{15}$$

Observe the following identities for $\Gamma \in \mathfrak{se}(3), v \in \mathbb{R}^3$.

$$\Gamma\bar{v} = F(v)\Gamma^\vee \qquad F(v) := \begin{bmatrix} -v_\times & \boldsymbol{I}_3 \\ 0 & 0\end{bmatrix} \tag{16}$$

$$\Gamma^\top\bar{v} = G(v)\Gamma^\vee \qquad G(v) := \begin{bmatrix} v_\times & 0 \\ 0 & v^\top\end{bmatrix} \tag{17}$$

We define the symmetric and skew-symmetric projections of $\mathbb{R}^{n\times n}$, $\mathbb{P}_s$ and $\mathbb{P}_a$ respectively, and the unique orthogonal projection, $\mathbb{P}$, of $\mathbb{R}^{4\times 4}$ onto $\mathfrak{se}(3)$ with respect to the Frobenius inner product.

$$\mathbb{P}_s : \mathbb{R}^{n\times n} \to \mathrm{sym}(n) \qquad \mathbb{P}_s(M) := \frac{1}{2}(M + M^\top) \tag{18}$$

$$\mathbb{P}_a : \mathbb{R}^{n\times n} \to \mathfrak{so}(n) \qquad \mathbb{P}_a(M) := \frac{1}{2}(M - M^\top) \tag{19}$$

$$\mathbb{P} : \mathbb{R}^{4\times 4} \to \mathfrak{se}(3) \quad \mathbb{P}\left(\begin{bmatrix} A_{3\times 3} & B_{3\times 1} \\ C_{1\times 3} & D_{1\times 1}\end{bmatrix}\right) := \begin{bmatrix} \mathbb{P}_a(A) & B \\ 0 & 0\end{bmatrix} \tag{20}$$

Lastly, we define the element-wise multiplication operator, $\odot$, for a general group, G.

$$\odot : G^n \times G^n \to G^n \tag{21}$$

$$(x_1, \dots, x_n) \odot (y_1, \dots, y_n) := (x_1y_1, \dots, x_ny_n) \tag{22}$$

We will omit the $\odot$ symbol when the meaning is clear from context.

2.2 Metrics

Let $T_X\mathrm{SE}(3)$ denote the tangent space to the manifold SE(3) at the point X. Note that the Lie algebra $\mathfrak{se}(3)$ coincides with $T_{\boldsymbol{I}}\mathrm{SE}(3)$ and that for all $\Gamma \in \mathfrak{se}(3)$, the tangent vector $X\Gamma \in T_X\mathrm{SE}(3)$.

Let the metric $\langle .,. \rangle_X : T_X\mathrm{SE}(3) \times T_X\mathrm{SE}(3) \to \mathbb{R}$ denote the standard left-invariant Riemannian metric on SE(3), that is

$$\langle X\Gamma, X\Psi\rangle_X = \langle\Gamma, \Psi\rangle_I = \langle\Gamma, \Psi\rangle \tag{23}$$

$$= \operatorname{tr}\left(\begin{bmatrix}0.5\boldsymbol{I}_3 & 0\\ 0 & 1\end{bmatrix}\Gamma^\top\Psi\right) \tag{24}$$

$$= \langle\Gamma^\vee, \Psi^\vee\rangle = \left(\Gamma^\vee\right)^\top \Psi^\vee \tag{25}$$

for $\Gamma, \Psi \in \mathfrak{se}(3)$.

We define the distance function, d_P, applied to two elements $\boldsymbol{X}_1, \boldsymbol{X}_2 \in \mathrm{SE}(3)^n$, weighted by a positive definite matrix $P \in \mathbb{R}^{6n\times 6n} \succ 0$ as

$$d_P(\boldsymbol{X}_1, \boldsymbol{X}_2) := \sqrt{\langle P\log(\boldsymbol{X}_2^{-1}\odot\boldsymbol{X}_1)^{\curlyvee}, \log(\boldsymbol{X}_2^{-1}\odot\boldsymbol{X}_1)^{\curlyvee}\rangle}. \tag{26}$$

2.3 Differential Geometric Notation

Let $f : \mathrm{SE}(3) \to \mathbb{R}$ denote a differentiable map. Then $\mathcal{D}_X f(X) : T_X\mathrm{SE}(3) \to \mathbb{R}$ denotes the Fréchet derivative and we have

$$\mathcal{D}_X f(X) \circ (X\Gamma) = \langle\nabla_X f(X), X\Gamma\rangle_X \tag{27}$$

where $X\Gamma \in T_X\mathrm{SE}(3)$ denotes the tangent direction in which the derivative is evaluated and $\nabla_X f(X)$ denotes the gradient at the point X with respect to the metric $\langle .,.\rangle_X$.

The second order differential map $\mathcal{D}^2_X f(x) : T_X\mathrm{SE}(3) \times T_X\mathrm{SE}(3) \to \mathbb{R}$ is defined as

$$\mathcal{D}^2_X f(x) \circ (X\Gamma, X\Psi) = \langle\mathrm{Hess}_X f(X) \circ (X\Psi), X\Gamma\rangle_X \tag{28}$$

$$= \langle\mathrm{Hess}_X f(X) \circ (X\Gamma), X\Psi\rangle_X \tag{29}$$

where $\mathrm{Hess}_X f(X)$ denotes the Hessian operator. The map can also be written in terms of first-order derivatives:

$$\mathcal{D}^2_X f(x) \circ (X\Gamma, X\Psi) = \mathcal{D}_X(\mathcal{D}_X f(X) \circ (X\Gamma)) \circ (X\Psi) \\ -\langle\nabla_X f(X), X\boldsymbol{\Lambda}_\Psi(\Gamma)\rangle_X \tag{30}$$

where $\boldsymbol{\Lambda}_\Psi : \mathfrak{se}(3) \to \mathfrak{se}(3)$ is the connection function. In this paper, we use the symmetric Cartan connection:

$$\boldsymbol{\Lambda}_\Psi(\Gamma) := \frac{1}{2}(\Psi\Gamma - \Gamma\Psi) \tag{31}$$

3. PROBLEM FORMULATION

We consider n mobile robots in a fully-connected network with node set $N := \{1, \ldots, n\}$. A set of r landmarks $L := \{l_i \in \mathbb{R}^3 \mid i = 1, \ldots, r\}$ are placed in the environment at fixed locations. Each robot is equipped with a suite of interoceptive and exteroceptive sensors as well as a method to communicate directly to other robots in the network. We aim to derive a deterministic second-order approximate minimum energy filter to estimate the pose of each robot in the network. Initially, this will be formulated as a set of centralised equations but we will show how the filter can be decoupled and distributed among the robots in the network.

3.1 Kinematics

The rotation, R_i, and translation, p_i, of each robot $i \in N$ with respect to a fixed reference frame is represented as a 4×4 homogeneous matrix, X_i. The pose has the following left-invariant kinematics.

$$X_i = \begin{bmatrix}R_i & p_i\\ 0 & 1\end{bmatrix} \in \mathrm{SE}(3) \qquad \Omega_i = \begin{bmatrix}\omega_i\\ v_i\end{bmatrix}^\wedge \in \mathfrak{se}(3) \tag{32}$$

$$\dot{X}_i = X_i\Omega_i \qquad X_i(0) = X_{i,0} \tag{33}$$

where ω_i and v_i are the angular and linear velocities of the robot with respect to the reference frame.

3.2 Measurements

A robot, i, can independently measure its own velocity. The measurement, u_i, is corrupted by zero-mean sensor noise, $\epsilon_i \in \mathbb{R}^6$.

$$u_i = \begin{bmatrix}\omega_i\\ v_i\end{bmatrix} + B_i\epsilon_i \tag{34}$$

where $B_i \in \mathbb{R}^{6\times 6}$ is determined by the sensor properties.

Each robot is equipped with a sensor that measures the relative translation between the robot and landmarks in the environment. A measurement, $y \in \mathbb{R}^3$, of the landmark $l \in L$ taken by robot i is corrupted by zero-mean sensor noise, $\delta \in \mathbb{R}^3$.

$$\bar{y} = X_i^{-1}\bar{l} + \tilde{C}\mathring{\delta} \tag{35}$$

where $C \in \mathbb{R}^{3\times 3}$ is determined by the sensor properties.

A similar sensor on each robot also measures the relative translation to other robots in the network. A robot, i, senses and identifies a known marker point, m_j, affixed to another robot, j. The measurement, $z_{ij} \in \mathbb{R}^3$, is corrupted by zero-mean sensor noise, $\eta \in \mathbb{R}^3$.

$$\bar{z}_{ij} = X_i^{-1}X_j\bar{m}_j + \tilde{D}\mathring{\eta} \tag{36}$$

where $D \in \mathbb{R}^{3\times 3}$ is determined by the sensor properties. The marker point, m_j, is known and is defined with respect to the body-fixed frame of robot j.

Landmark and robot measurements are not necessarily available at all times or to all robots. Measurements may be intermittent and a robot may only be able to observe a subset of L and N at any given time.

3.3 Global State Formulation

We introduce the global state variable, $\boldsymbol{X}$, which comprises of the states of all robots in the network.

$$\boldsymbol{X} := (X_1, \ldots, X_n) \in \mathrm{SE}(3)^n \tag{37}$$

We then have

$$\dot{\boldsymbol{X}} := \boldsymbol{X} \odot (\Omega_1, \ldots, \Omega_n), \qquad \boldsymbol{X}(0) = \boldsymbol{X}_0, \tag{38}$$

and denote

$$\boldsymbol{u} := (u_1, \ldots, u_n), \qquad \boldsymbol{\epsilon} := (\epsilon_1, \ldots, \epsilon_n). \tag{39}$$

3.4 Cost Functional

As discussed in Section 3.2, each robot can receive information from three different sensors to provide measurements of velocity, positions of landmarks and positions of other robots. We follow the approach taken by Zamani and Trumpf (2019) to define the problem in terms of a continuous-time propagation step that uses the velocity measurements, and a discrete time update step, using either the landmark or robot measurements.

Following Mortensen's formulation of the deterministic minimum energy problem (Mortensen, 1968), we introduce the following continuous-time cost functional, J_t.

$$J_t(\boldsymbol{X}, \boldsymbol{\epsilon}) := \frac{1}{2}d^2_{P_0}\left(\boldsymbol{X}(0), \hat{\boldsymbol{X}}_0\right) + \frac{1}{2}\sum_{i\in N}\int_0^t \|\epsilon_i\|^2\, d\tau \tag{40}$$

where P_0 is a positive definite matrix which weights the initial estimate. We assume that, relative to the exteroceptive measurements, the velocity measurements are available at a high enough frequency that they can be regarded as a continuous signal.

We can now formally define the minimum energy filtering problem: Given a sequence of velocity measurements, $\boldsymbol{u}[0,t]$, find an estimate, $\hat{\boldsymbol{X}}(t) \in \mathrm{SE}(3)^n$, of the state of the system, $\boldsymbol{X}(t)$, that minimises the cost functional $J_t(\hat{\boldsymbol{X}}, \boldsymbol{\epsilon})$ and is consistent with the kinematics described in (38). The estimate must also be formulated as a recursive equation, dependent only on the measurements and the state estimate at the current time.

Minimising J_t is performed in two steps – firstly by minimising over $\boldsymbol{\epsilon}$, and then minimising over a point $\boldsymbol{X}$ on the trajectory. We introduce the value function, V, to represent the first step in this process.

$$V(\boldsymbol{X}, t) := \min_{\boldsymbol{\epsilon}[0,t]} J_t(\boldsymbol{X}, \boldsymbol{\epsilon}) \tag{41}$$

$$V(\boldsymbol{X}(0), 0) = \frac{1}{2} d^2_{P_0}\left(\boldsymbol{X}(0) - \hat{\boldsymbol{X}}_0\right) \tag{42}$$

The optimal state estimate is then given by

$$\hat{\boldsymbol{X}}(t) := \arg\min_{\boldsymbol{X}} V(\boldsymbol{X}, t). \tag{43}$$

We now consider the exteroceptive landmark and robot measurements. As in Zamani and Trumpf (2019), we introduce a discrete-update value function, V^+, for landmark measurements. Additionally, we introduce a second value function, $V^{+\!\!+}$, for measurements of other robots.

$$V^+(\boldsymbol{X}, t) := V(\boldsymbol{X}, t) + \frac{1}{2}\left\|X_i \bar{y}_i - \bar{l}\right\|^2_{P_{\bar{y}}^{\text{-}1}} \tag{44}$$

$$V^{+\!\!+}(\boldsymbol{X}, t) := V(\boldsymbol{X}, t) + \frac{1}{2}\left\|X_i \bar{z}_{ij} - X_j \bar{m}_j\right\|^2_{P_{\bar{z}}^{\text{-}1}} \tag{45}$$

$$P_{\bar{y}} := \tilde{C}\tilde{C}^\top, \quad P_{\bar{z}} := \tilde{D}\tilde{D}^\top \tag{46}$$

The optimal minimum-energy state estimate is given by $\hat{\boldsymbol{X}}^+(t)$ or $\hat{\boldsymbol{X}}^{+\!\!+}(t)$, respectively

$$\hat{\boldsymbol{X}}^+(t) := \arg\min_{\boldsymbol{X}} V^+(\boldsymbol{X}, t) \tag{47}$$

$$\hat{\boldsymbol{X}}^{+\!\!+}(t) := \arg\min_{\boldsymbol{X}} V^{+\!\!+}(\boldsymbol{X}, t) \tag{48}$$

While these equations have been formulated for a single measurement, they are applied to each landmark or robot measurement at the time they are received.

4. RESULTS

In this section, we derive the filter equations for the centralised state estimation problem and then show how they can be decoupled and distributed among the robots in the network.

4.1 Central GAME Filter Formulation

The following lemma is a simple consequence of the relevant definitions.

Lemma 1. Given any two tangent directions $\boldsymbol{X\Gamma}$, $\boldsymbol{X\Psi} \in T_{\boldsymbol{X}}\mathrm{SE}(3)^n$, the Hessian of the value function, acting as a symmetric mapping with respect to the inner product is equivalently represented with a positive definite matrix, $P \in \mathbb{R}^{6n \times 6n}$, operating on vectors $\boldsymbol{\Gamma}^\curlyvee, \boldsymbol{\Psi}^\curlyvee \in \mathbb{R}^{6n}$.

$$\left\langle P\boldsymbol{\Psi}^\curlyvee, \boldsymbol{\Gamma}^\curlyvee \right\rangle := \langle \mathrm{Hess}_{\boldsymbol{X}} V(\boldsymbol{X}, t) \circ \boldsymbol{X\Psi}, \boldsymbol{X\Gamma} \rangle_{|\boldsymbol{X} = \hat{\boldsymbol{X}}(t)} \tag{49}$$

Propagation of Velocity Measurements. Following the methodology in Theorem 1 and Theorem 2 of Zamani and Trumpf (2019) results in the following filter state propagation equations for the centralised system.

$$\dot{\hat{\boldsymbol{X}}}(t) = \hat{\boldsymbol{X}}(t) \odot \boldsymbol{u}(t) \tag{50}$$

$$\dot{P}(t) = -P\boldsymbol{B}\boldsymbol{B}^\top P + \mathbb{P}_s(P\boldsymbol{U}), \qquad P(0) = P_0 \tag{51}$$

where

$$U := \begin{bmatrix} (u_\omega)_\times & 0 \\ (u_v)_\times & (u_\omega)_\times \end{bmatrix}, \tag{52}$$

$$\boldsymbol{U} := \mathrm{blkdiag}(U_1, U_2, \ldots, U_n), \tag{53}$$

$$\boldsymbol{B} := \mathrm{blkdiag}(B_1, B_2, \ldots, B_n). \tag{54}$$

We note here that $\boldsymbol{B}$ is block diagonal, indicating our assumption that velocity measurements on-board one robot are independent of all other robots.

Landmark Measurement Update

Theorem 2. Consider a single relative position measurement of a landmark, y, as defined in (35). The approximate minimum-energy recursive solution to the estimate of the state $\boldsymbol{X}$, as defined in (47) is

$$\hat{\boldsymbol{X}}^+ = \hat{\boldsymbol{X}} \odot \boldsymbol{\Theta} \tag{55}$$

where

$$\boldsymbol{\Theta} = \exp\left(\left(-(P^+)^{\text{-}1}\left(\hat{\boldsymbol{X}}^{\text{-}1}\nabla_{\boldsymbol{X}} V^+(\hat{\boldsymbol{X}}(t), t)\right)^\curlyvee\right)^\curlywedge\right) \tag{56}$$

$$\nabla_{X_i} V^+(\hat{\boldsymbol{X}}(t), t) = \hat{X}_i \mathbb{P}\left(\hat{X}_i^\top P_{\bar{y}}^{\text{-}1}(\hat{X}_i \bar{y} - \bar{l})\bar{y}^\top(2\tilde{\boldsymbol{I}}_3)\right) \tag{57}$$

$$\nabla_{X_j} V^+(\hat{\boldsymbol{X}}(t), t) = 0 \quad \forall j \neq i. \tag{58}$$

P^+ is the matrix equivalent to $\mathrm{Hess}_{\boldsymbol{X}} V^+(\hat{\boldsymbol{X}}(t), t)$, as defined in Lemma 1, and can be calculated as

$$P^+ = P + Q \tag{59}$$

$$\begin{aligned} Q_{ii} = \mathbb{P}_s &\left(F(\bar{y})^\top G(\hat{X}_i^\top P_{\bar{y}}^{\text{-}1}(\hat{X}_i \bar{y} - \bar{l}))\right) \\ &+ F(\bar{y})^\top \hat{X}_i^\top P_{\bar{y}}^{\text{-}1} \hat{X}_i F(\bar{y}) \end{aligned} \tag{60}$$

$$Q_k = \boldsymbol{0} \quad \forall k \neq (i, i). \tag{61}$$

Here, $Q \in \mathbb{R}^{6n \times 6n}$ is indexed in blocks of 6×6 elements, so that Q_{ii} refers to the ith 6×6 block matrix on the diagonal of Q. Recall the definitions of F and G from (16) and (17), respectively.

Proof: We first perform a Taylor expansion of $V^+(\boldsymbol{X}, t)$ to second order around the point $\boldsymbol{X} = \hat{\boldsymbol{X}}(t)$ along the geodesic $\boldsymbol{\Psi} = \log(\hat{\boldsymbol{X}}^{\text{-}1} \odot \boldsymbol{X})$. This yields an approximate solution as the value function is not guaranteed to be of second order. Ignoring the higher order terms,

$$\begin{aligned} V^+(\boldsymbol{X}, t) = V^+(\hat{\boldsymbol{X}}, t) &+ \langle \nabla_{\boldsymbol{X}} V^+(\hat{\boldsymbol{X}}, t), \hat{\boldsymbol{X}}\boldsymbol{\Psi} \rangle \\ &+ \frac{1}{2}\langle \mathrm{Hess}_{\boldsymbol{X}} V^+(\hat{\boldsymbol{X}}, t) \circ \hat{\boldsymbol{X}}\boldsymbol{\Psi}, \hat{\boldsymbol{X}}\boldsymbol{\Psi} \rangle \end{aligned} \tag{62}$$

As a consequence of (47), we have

$$\{\mathcal{D}_{\boldsymbol{X}} V^+(\boldsymbol{X}, t) \circ \boldsymbol{X\Gamma}\}_{\boldsymbol{X} = \hat{\boldsymbol{X}}^+(t)} = 0 \tag{63}$$

which we can substitute in (62). Combined with the consequence from (43) that $\nabla_{\boldsymbol{X}} V(\hat{\boldsymbol{X}}, t) = 0$, it follows that

$$\begin{aligned} 0 = \Big\{ &\Big\langle \nabla_{\boldsymbol{X}} V^{+}(\hat{\boldsymbol{X}}(t), t), \mathcal{D}_{\boldsymbol{X}}(\hat{\boldsymbol{X}}\boldsymbol{\Psi}) \circ (\hat{\boldsymbol{X}}\boldsymbol{\Gamma}) \Big\rangle \\ &+ \Big\langle \mathrm{Hess}_{\boldsymbol{X}} V^{+}(\hat{\boldsymbol{X}}(t), t) \circ \hat{\boldsymbol{X}}\boldsymbol{\Psi}, \\ &\qquad \mathcal{D}_{\boldsymbol{X}}(\hat{\boldsymbol{X}}\boldsymbol{\Psi}) \circ (\hat{\boldsymbol{X}}\boldsymbol{\Gamma}) \Big\rangle \Big\}_{\boldsymbol{X} = \hat{\boldsymbol{X}}^{+}(t)} \end{aligned} \tag{64}$$

Rearranging to solve for $\hat{\boldsymbol{X}}^{+}(t)$, together with (49), results in (55). Equation (57) is then derived by evaluating

$$\langle \nabla_{X_i} V^{+}(\hat{\boldsymbol{X}}(t), t), \hat{X}_i \Gamma \rangle \tag{65}$$

using (27), (44), and (63). We then evaluate the derivative and reformulate in terms of (24) to solve for $\nabla_{X_i} V^{+}(\hat{\boldsymbol{X}}(t), t)$. We calculate P^{+} by observing that

$$\begin{aligned} \mathrm{Hess}_{\boldsymbol{X}} V^{+}(\hat{\boldsymbol{X}}(t), t) &= \mathrm{Hess}_{\boldsymbol{X}} V(\hat{\boldsymbol{X}}(t), t) \\ &+ \mathrm{Hess}_{\boldsymbol{X}} \left(\frac{1}{2} \left\| X_i \bar{y}_i - \bar{l} \right\|^2_{P_{\bar{y}}^{-1}} \right). \end{aligned} \tag{66}$$

□

Robot Measurement Update

Theorem 3. In the same manner as Theorem 2, the approximate minimum-energy recursive estimate for the state $\boldsymbol{X}$ after a robot measurement, z_{ij}, can be calculated as

$$\hat{\boldsymbol{X}}^{+\!\!+} = \hat{\boldsymbol{X}} \odot \boldsymbol{\Xi} \tag{67}$$

where

$$\boldsymbol{\Xi} = \exp\left(\left(-(P^{+\!\!+})^{-1} \left(\hat{\boldsymbol{X}}^{-1} \odot \nabla_{\boldsymbol{X}} V^{+\!\!+}(\hat{\boldsymbol{X}}(t), t) \right)^{\curlyvee} \right)^{\curlywedge} \right) \tag{68}$$

$$\nabla_{X_i} V^{+\!\!+}(\hat{\boldsymbol{X}}(t), t) = \hat{X}_i \mathbb{P}\left(\hat{X}_i^{\top} P_{\bar{z}}^{-1} \left(\hat{X}_i \bar{z} - \hat{X}_j \bar{m} \right) \bar{z}^{\top} (2\tilde{\boldsymbol{I}}_3) \right) \tag{69}$$

$$\nabla_{X_j} V^{+\!\!+}(\hat{\boldsymbol{X}}(t), t) = \hat{X}_j \mathbb{P}\left(\hat{X}_j^{\top} P_{\bar{z}}^{-1} \left(\hat{X}_i \bar{z} - \hat{X}_j \bar{m} \right) \bar{m}^{\top} (2\tilde{\boldsymbol{I}}_3) \right) \tag{70}$$

$$\nabla_{X_k} V^{+\!\!+}(\hat{\boldsymbol{X}}(t), t) = 0 \quad \forall k \notin \{i, j\} \tag{71}$$

$P^{+\!\!+}$ is the matrix equivalent to $\mathrm{Hess}_{\boldsymbol{X}} V^{+\!\!+}(\hat{\boldsymbol{X}}(t), t)$, as defined in Lemma 1, and can be calculated as

$$P^{+\!\!+} = P + W \tag{72}$$

$$\begin{aligned} W_{ii} = \mathbb{P}_s &\left(F(\bar{z})^{\top} G(\hat{X}_i^{\top} P_{\bar{z}}^{-1} (\hat{X}_i \bar{z} - \hat{X}_j \bar{m})) \right) \\ &+ F(\bar{z})^{\top} \hat{X}_i^{\top} P_{\bar{z}}^{-1} \hat{X}_i F(\bar{z}) \end{aligned} \tag{73}$$

$$W_{ij} = F(\bar{z})^{\top} \hat{X}_i^{\top} P_{\bar{z}}^{-1} \hat{X}_j F(\bar{m}), \qquad W_{ji} = W_{ij}^{\top} \tag{74}$$

$$\begin{aligned} W_{jj} = \mathbb{P}_s &\left(F(\bar{m})^{\top} G(\hat{X}_j^{\top} P_{\bar{z}}^{-1} (\hat{X}_j \bar{m} - \hat{X}_i \bar{z})) \right) \\ &+ F(\bar{z})^{\top} \hat{X}_j^{\top} P_{\bar{z}}^{-1} \hat{X}_j F(\bar{m}) \end{aligned} \tag{75}$$

$$W_k = \boldsymbol{0} \quad \forall k \notin \{(i,i), (i,j), (j,i), (j,j)\}. \tag{76}$$

Here, $W \in \mathbb{R}^{6n \times 6n}$ is indexed in the same way as Q from the previous section.

Proof: The proof follows along the same lines as the proof of Theorem 2.

4.2 Decoupled Central GAME Filter Formulation

Given the set of equations that define the propagation and update steps for the centralised GAME filter, we now attempt to decouple the equations so that the calculations can be distributed among the robots in the network. We will find that the decoupling of the filter equations is easier when working with the inverse of the Hessian, $\Sigma := P^{-1}$.

In the following formulation each robot, i, tracks its own state estimate, $\hat{X}_i$, and an $n \times 6$ sub-matrix of Σ, Σ_{ki}.

Propagation Step We note that the state propagation equation (50) is trivial to decouple.

$$\dot{\hat{X}}_i(t) = \hat{X}_i(t) u_i(t) \quad \forall i \in N \tag{77}$$

To decouple the calculation for P, we reformulate (51) in terms of Σ, which allows us to separate $\dot{\Sigma}$ into components.

$$\dot{\Sigma} = \boldsymbol{B}\boldsymbol{B}^{\top} - \mathbb{P}_s(\boldsymbol{U}\Sigma) \tag{78}$$

$$\dot{\Sigma}_{ii} = B_i B_i^{\top} - \mathbb{P}_s(U_i \Sigma_{ii}) \tag{79}$$

$$\dot{\Sigma}_{ij} = -\frac{1}{2}\left(U_i \Sigma_{ij} + \Sigma_{ij} U_j^{\top} \right) \tag{80}$$

Here, we use the same 6×6 block indexing as in previous sections. We observe that the diagonal sub-matrices, $\dot{\Sigma}_{ii}$, only depend on data local to robot i, while the off-diagonal sub-matrices, $\dot{\Sigma}_{ij}$, depend on data local to both robot i and j. Given that (80) is a homogeneous linear ODE, we can find an explicit solution if we assume that U_i and U_j are constant.

$$\Sigma_{ij}(t) = \exp\left(-\frac{t}{2} U_i \right) \Sigma_{ij}(0) \exp\left(-\frac{t}{2} U_j^{\top} \right) \tag{81}$$

In reality, the velocity measurements are received from a sensor which updates at a fixed time interval, which means that U_i and U_j do remain constant for a time Δt, which represents the time between two successive measurements. Thus, we can recursively evaluate Σ_{ij} at a time t_n, after the n-th measurement is recorded by

$$\Sigma_{ij}(t_n) = \exp\left(-\tfrac{\Delta t}{2} U_i(t_n) \right) \Sigma_{ij}(t_{n-1}) \exp\left(-\tfrac{\Delta t}{2} U_j(t_n)^{\top} \right) \tag{82}$$

$$= K_i(t_n) \Sigma_{ij}(0) K_j(t_n)^{\top} \tag{83}$$

$$K_i(t_n) = \prod_{k=n}^{1} \exp\left(-\frac{\Delta t}{2} U_i(t_k) \right) \tag{84}$$

Based on this formulation, we observe that K_i can be computed independently by robot i and similarly K_j can be computed by robot j. Robot i can calculate $\Sigma_{ji}(t_n)$ by receiving a message from robot j that contains $K_j(t_n)$.

In this way, we show a parallel result to Roumeliotis and Bekey (2002) whereby robots can propagate their state independently and only need to share information at a time where exteroceptive measurements are taken.

Robot Measurement Update In the robot measurement update step, we are required to decouple (67) and (72). This would be a straightforward task for (72) if P was known, however given that the propagation step has been computed in terms of Σ, this would require a full matrix-inversion of Σ, which is only possible in a centralised system — not in our system where each robot is only tracking a sub-matrix. Thus, the update step must also be reformulated and then decoupled in terms of Σ.

$$\Sigma^{+\!\!+} = (\boldsymbol{I}_{6n} + \Sigma W)^{-1}\,\Sigma \tag{85}$$

Recall the definition of W from (73) through (76). We observe that, because of the sparsity of W, the only elements of Σ that need to be known in order to compute $(\boldsymbol{I}_{6n} + \Sigma W)^{-1}$ are Σ_{ki} and Σ_{kj}, $k \in N$. This corresponds to the elements of Σ that are being tracked by robot i and j, respectively, and means that the inverse can be computed locally between robot i and j. Once calculated, this term can then be shared with all other robots in the network to calculate the value for $\Sigma^{+\!\!+}$.

Similarly, (69) and (70) can be computed locally between robots i and j, which also allows (68) to be computed locally. Each component of $\boldsymbol{\Xi}$ can then be communicated to the relevant robot such that (67) can be computed locally to each robot.

$$\hat{X}_i^{+\!\!+} = \hat{X}_i \Xi_i \quad \forall i \in N \tag{86}$$

One of the issues with the current formulation is that (85) requires inverting a $6n \times 6n$ matrix. However, we observe that $\operatorname{rank}(W) \leqslant 12$ and we perform a singular value decomposition on ΣW as follows.

$$TSV^{\top} = \Sigma W \tag{87}$$

$$TT^{\top} = VV^{\top} = \boldsymbol{I}_{6n}, \quad S \in \mathbb{D}^{12}, \quad T, V \in \mathbb{R}^{6n \times 12} \tag{88}$$

We can then apply the matrix inversion lemma (Woodbury matrix identity) which reduces the size of the matrix that is inverted from $6n \times 6n$ to a maximum of 12×12, depending on the actual rank of W. S is diagonal and can be trivially inverted.

$$\Sigma^{+\!\!+} = \left(\boldsymbol{I}_{6n} - T(S^{-1} + V^{\top}T)^{-1}V^{\top}\right)\Sigma \tag{89}$$

This also reduces the size of messages that need to be communicated, as instead of sending a $6n \times 6n$ matrix, T, S, and V can be sent individually, which is only $6n \times 24 + 12$ elements.

Landmark Measurement Update The decoupling of the landmark measurement update equations follows in a similar way to the previous section, giving

$$\Sigma^{+} = (\boldsymbol{I}_{6n} + \Sigma Q)^{-1}\Sigma. \tag{90}$$

Calculating ΣQ only requires Σ_{ki}, $k \in N$ to be known. Thus robot i can perform the matrix inversion locally, and then communicate the required information for all other robots to update their state and respective components of Σ^{+}. Similarly, $\boldsymbol{\Theta}$ can be calculated by robot i and distributed to each robot to perform the update of the state estimate locally by

$$\hat{X}_i^{+} = \hat{X}_i \Theta_i \quad \forall i \in N. \tag{91}$$

If we perform an SVD of ΣQ in a similar way to (87), we note that $\operatorname{rank}(Q) \leq 6$ and thus the resulting decomposition produces matrices maximally of size $6n \times 6$ and a maximum total message size of $6n \times 12 + 6$.

5. SIMULATIONS

We demonstrate the performance of the resulting filter in two Python simulations.[1] The first considers the case where the robots' poses are constrained to a 2-D plane, such as in the case of a network of ground based robots, while the second scenario considers the more general case of 3-D trajectories.

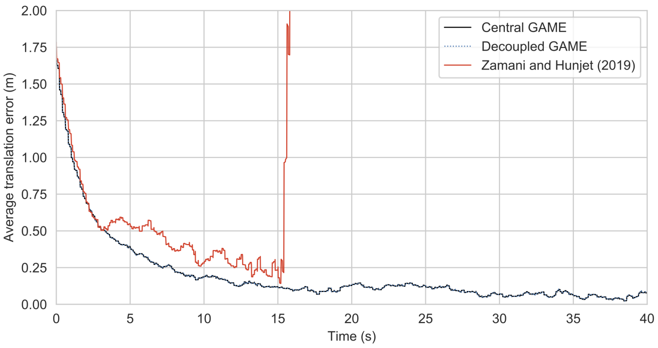

Fig. 1. Average translation error across all robots for 2-D Scenario. Best viewed in colour.

5.1 2-D Case

We consider a network of $n = 4$ robots moving along circular trajectories within an approximately 20m × 20m area. There are $r = 4$ landmarks in the environment and each robot is only able to take measurements of one distinct landmark at a rate of 10 Hz. Velocity measurements are available to each robot at a rate of 100 Hz. Robots can observe only a single other distinct robot at a rate of 5 Hz but can communicate freely to all. To be specific, Robot 1 can observe Robot 2, R2 can observe R3, R3 can observe R4, and R4 can observe R1. The sensor properties are defined as

$$B = 0.05\boldsymbol{I}_6, \qquad C = 0.5\boldsymbol{I}_3, \qquad D = 0.5\boldsymbol{I}_3, \tag{92}$$

$$\epsilon \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_6), \qquad \delta \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_3), \qquad \eta \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_3). \tag{93}$$

Note, the sensor errors are constrained appropriately in the 2-D case.

We implement three filters, the centralised GAME filter described in Section 4.1, the decoupled GAME filter described in Section 4.2, and the collaborative GAME filter proposed by Zamani and Hunjet (2019), using the Covariance Intersection estimation method. The average translation error over all robots is shown in Figure 1.

We can observe that the centralised GAME filter and the decoupled GAME filter provide identical state estimates, demonstrating that there is no loss of information when the centralised filter is decoupled. Our filter is able to accurately localise the network of robots from an initial average translation error of 1.8m down to a long term average of 0.08m. This simulation also highlights a weakness in the filter from Zamani and Hunjet (2019) which stems from the asymmetry of the robot observations. As it does not share the information gained from measurements to other robots, the filter is not able to accurately localise and it diverges after approximately 15 seconds.

5.2 3-D Case

We present a different scenario to highlight the difference in filter performance even when measurements are readily available to all robots. We again consider a network of $n =$

[1] Code is available at `jackhenderson.com.au`

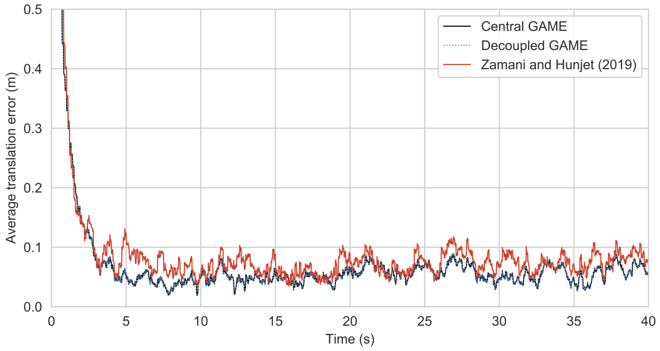

Fig. 2. Average translation error across all robots for 3-D Scenario. Best viewed in colour.

4 robots with $r = 4$ fixed landmarks in the environment. Velocity measurements are available to each robot at a rate of 100 Hz. In contrast to the previous scenario, each robot can observe all 4 landmarks at a rate of 10 Hz, and can observe all other robots at a rate of 10 Hz. Robots move with continuously changing random velocities in an approximately 20m × 20m × 20m volume. The sensor properties are the same as defined in (92) and (93). The average translation error of the three different filters is shown in Figure 2. The covariance intersection method in the Zamani and Hunjet (2019) filter was coarsely tuned to a value of $\omega = 0.03$.

Given the abundance of landmark measurements, both filters localise rapidly from the original initialisation error of 1.8m. However, the Zamani and Hunjet (2019) filter converges to a long-term average error of 0.074m, compared to 0.053m for our filter.

6. CONCLUSION

In this paper, we have shown how minimum energy filtering can be applied to the collaborative localisation problem. We demonstrate how our centralised filter can be decoupled and distributed among the robots in the network. The simulations presented verify that no information is lost when distributing the filters and demonstrates superior performance compared with previous work.

Planned future work includes a comparison to similar EKF filters and a demonstration on real hardware. While the all-to-all communication requirements for this algorithm may be too restrictive for many scenarios, we can use the algorithm developed here to guide further research. In the same way that Roumeliotis and Bekey (2002) has been used as the benchmark for further improvements to EKF localisation algorithms, we can use our filter as the benchmark to quantify the reduction in performance that comes with reducing communication constraints for minimum energy filters.

REFERENCES

Carrillo-Arce, L.C., Nerurkar, E.D., Gordillo, J.L., and Roumeliotis, S.I. (2013). Decentralized multi-robot cooperative localization using covariance intersection. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1412–1417.

Howard, A., Mataric, M., and Sukhatme, G. (2003). Putting the 'I' in 'team': An ego-centric approach to cooperative localization. In *2003 IEEE International Conference on Robotics and Automation*, 868–874.

Luft, L., Schubert, T., Roumeliotis, S.I., and Burgard, W. (2018). Recursive decentralized localization for multi-robot systems with asynchronous pairwise communication. *The International Journal of Robotics Research*, 37(10), 1152–1167.

Markley, F.L. (2003). Attitude Error Representations for Kalman Filtering. *Journal of Guidance, Control, and Dynamics*, 26(2), 311–317.

Mortensen, R.E. (1968). Maximum-likelihood recursive nonlinear filtering. *Journal of Optimization Theory and Applications*, 2(6), 386–394.

Roumeliotis, S. and Bekey, G. (2002). Distributed multi-robot localization. *IEEE Transactions on Robotics and Automation*, 18(5), 781–795.

Zamani, M. and Hunjet, R. (2019). Collaborative Pose Filtering Using Relative Measurements and Communications. In *12th Asian Control Conference (ASCC)*, 919–924.

Zamani, M. and Trumpf, J. (2019). Discrete update pose filter on the special Euclidean group SE(3). In *Proceedings of the 55th IEEE Conference on Decision and Control (CDC)*.

Zamani, M., Trumpf, J., and Mahony, R. (2013). Minimum-Energy Filtering for Attitude Estimation. *IEEE Transactions on Automatic Control*, 58(11), 2917–2921.