R-RANSAC with Lie Groups

I. Introduction

This note assumes the reader has a basic understanding of the theory relating to Matrix Lie Groups and their corresponding Lie algebras. This includes the matrix exponential and logarithm map, the vector space isomorphic to specific Lie algebras, the adjoint of the group and the algebra, and the left and right Jacobians. It also assumes that the reader has a basic understanding of R-RANSAC.

II. Terminology, Parameters, and Notation

The terminology, parameters, and notation used is previous version of R-RANSAC have varied which has created confusion and inconsistencies. In this section, we hope to solidify certain terminology, parameters and notation that are suitable for expanding R-RANSAC to work with Lie Groups, and incorporating other improvements we have made to the general algorithm.

A. Terminology

R-RANSAC has very specific terminology that we that will facilitate the discussion.

- **Phenomenon**: Something that produces an observable signal. In the case of target tracking, the phenomenon is referred to as a **target**, which is an object that exists in physical space.
- **Measurement Source**: A sensor equipped with an algorithm that captures information from the environment and produces meaningful measurements used to observe the target.
- **Surveillance Region**: The portion of the environment that is observable by the measurement sources. There is a local surveillance region (LSR) for each measurement source and a global surveillance region (GSR) that is a union of all the local surveillance regions.
- **Frame of reference**: Consists of an abstract coordinate system and the set of physical reference points that uniquely fix (locate and orient) the coordinate system and standardize measurements within that frame. We will often refer to a frame of reference as just **frame**.
- **Local Frame**: The frame that coincides with a local surveillance region.
- **Global Frame**: The frame that coincides with the global surveillance region. It is possible that the global frame is the same as a local frame.
- **Sensor Scan**: When a sensor observes its surveillance region and extracts meaningful data. For example, the sensor scan of a camera occurs when the camera produces a new image of its surveillance region.
- **False Measurement**: A measurement extracted from a sensor scan that does not correspond to a phenomenon of interest. For example, motion in a camera can generate false measurements due to parallax depending on the algorithm. Another example is just noisy sensors.
- **True Measurement**: A measurement extracted from a sensor scan that corresponds to a phenomenon of interest.
- **Model**: This is simply a model of the phenomenon. In regards to target tracking, a model is referred to as a **track**.
- **Model Hypothesis**: This is a hypothetical model of the phenomenon (i.e. a possible model) created by the RANSAC algorithm. A model hypothesis that meets certain criteria becomes a model. In regards to target tracking, a model hypothesis is referred to as a **track hypothesis**. We will often abbreviate the term and mention only **hypothesis**.
- **Model Likelihood**: The probability that a model represents an actual target.
- **Good Model**: A model that is deemed very likely to correctly describe a phenomenon, based on predefined criteria, becomes a good model. In regards to target tracking, a good model is referred to as a **good track**.
- **Poor Model**: A model that is not a good model. In regards to target tracking, a poor model is referred to as a **poor track**.
- **Time Window**: An interval of time extending into the past from the current time.
- **Expired Measurement**: A measurement that was observed in the past outside the time window.
- **Measurement Source**: An algorithm that takes sensor data and produces a measurement. We will often refer to a measurement source as just a source.

Don't worry if some of the terms do not make much sense now. We will elaborate on them later in the text.

B. Parameters

There are various parameters used throughout R-RANSAC. A parameter can either be a scalar or a tuple.

In case the context doesn't make it clear, we will specify which one is a scalar and a tuple.

RANSAC parameters

- ℓ - max iterations
- τ_E - RANSAC early termination threshold
- τ_I - RANSAC inlier threshold

Cluster parameters

- τ_D - neighborhood distance threshold
- τ_C - cluster min cardinality threshold

Model Management Parameters

- τ_S - Similarity merge threshold
- τ_ρ - Good track threshold
- τ_α- Missed detection threshold

PDAF

- P_D - Probability of detection
- P_G - Probability of being in the validation region
- λ - The spacial density of false measurements in a local surveillance region per sensor scan

Other

- T_W - Time window
- M - The number of models

C. Notation

In this note we are assuming a discrete system whose measurements arrive at non-fixed time intervals. We denote the current time using a subscript k, the previous time using a subscript k^-, and a previous time using the subscript m such that $m \leq k$. We denote the time at the beginning of the time window using 0. For example x_0 is the state of the system at the beginning of the time window. Let δ_k denote the time elapsed from k^- to k, and $\delta_{m:k}$ denote the time elapsed from m to k. Note that $\delta_{m:k}, \delta_{k^-} \geq 0$. Lastly, let $\delta_{-k} = -\delta_k$ and $\delta_{k:m} = -\delta_{m:k}$.

- Let G denote a Lie group
- Let $\mathfrak{g}$ denote the Lie algebra of G
- Let E denote the Euclidean vector space isomorphic to $\mathfrak{g}$
- Let $x \in G \times \mathfrak{g}$ denote the state of the system
- Let N denote the measurement space which is a Lie group
- Let $p(x)$ denote the probability of x, $p(x, y)$ the joint probability of x and y, and $p(x|y)$ the probability of x conditioned on y.
- ψ_k^i The set of measurements received at time k from the i^{th} measurement sensor.
- Ψ_k The set of measurements received at time k. $\Psi_k = \cup_{i \in J} \psi_k^i$ where J is an index associated with the measurement sources.
- $\Psi_{m:k}$ The set of measurements received from time m to time k.
- Ψ_{T_W} The set of measurements received over the current time window.
- Y^i The set of measurements pertaining to the i^{th} cluster during the time window.
- $y_{0:k}^i$ A minimum subset of Y^i such that the model can be observed using the measurements from $y_{0:k}^i$. One measurement from $y_{0:k}^i$ must be a new measurement and the others are randomly sampled from Y^i.

III. Lie Theory Review

The objective of this section is to provide a review of the pertinent concepts of Lie theory that we will use. We assume that the reader is familiar with Lie group and Lie algebra theory. For those unfamiliar with this material, we refer the interested reader primarily to [22]. Solá offers a very gentle introduction to Lie theory that covers the majority of information needed to understand this paper. We also follow much of the notation prescribed by Solá and Hertzberg in[22] and [10]. For those who are interested in a more rigorous treatment of Lie theory, we refer the reader to [3], [23], [9], [8], [12], [1]. In this document we will focus on targets with a discrete system model and constant velocity, this is merely to simplify the presentation of the material. With no loss in generality, everything we present can be extended to targets with a continuous system model and constant acceleration by using a semidirect product group formed from a Lie group and its Lie algebra to form a new Lie group. A good discussion of the semidirect product group formed from a Lie group and its Lie algebra can be found in [7]. If using a continuous system that requires numerical integration on the manifold, we refer the reader to [16], [17], [18] which describe the Runge-Kutta-Muthe-Kass numerical integration technique.

Let G denote a Lie group and $\mathfrak{g}$ denote it's corresponding Lie algebra. The exponential function is a surjective function that maps an element of the Lie algebra to an element of the Lie group, and the logarithmic map is the inverse of the exponential map. We denote both as

$$\exp : \mathfrak{g} \to G$$
$$\log : G \to \mathfrak{g}.$$

The definition of these maps is dependent on the Lie group. For matrix Lie groups, the exponential and logarithm maps are defined as the matrix exponential and matrix logarithm.

The Lie algebra is isomorphic to the Euclidean space $\mathbb{R}^n$ where n is the dimension of the Lie algebra. We will denote this Euclidean space as E. The Wedge function maps an element from the Euclidean space to the Lie

algebra, and the Vee function is the inverse function which we denote as

$$\begin{aligned} \cdot^{\wedge} &: E \to \mathfrak{g} \quad (v) \mapsto v^{\wedge} \\ \cdot^{\vee} &: \mathfrak{g} \to V \quad (v^{\wedge})^{\vee} \mapsto v. \end{aligned}$$

. The definition of these maps is dependent on the Lie algebra. We will rely on context to distinguish between elements of E and $\mathfrak{g}$.

Let $\text{Exp} : E \to G$ and its inverse be defined as the composite function

$$\begin{aligned} \text{Exp} &: E \to G \quad (v) \mapsto \exp(v^{\wedge}) \\ \text{Log} &: G \to V \quad (g) \mapsto \log(g)^{\vee}. \end{aligned}$$

Other functions of importance are the box-plus/minus and the o-plus/minus defined as

$$\begin{aligned} \boxplus &: G \times \mathfrak{g} \to G \quad (g, u) \mapsto g \bullet \exp(u) \\ \boxminus &: G \times G \to \mathfrak{g} \quad (g_1, g_2) \mapsto \log\left(g_2^{-1} \bullet g_1\right) \\ \oplus &: G \times E \to G \quad (g, v) \mapsto g \bullet \text{Exp}(v) \\ \ominus &: G \times G \to E \quad (g_1, g_2) \mapsto \text{Log}\left(g_2^{-1} \bullet g\right). \end{aligned}$$

where $\bullet$ represents the group operator which we will omit in the future, and g_2^{-1} is the inverse element of g_2. We have based the definition of these function on left trivializations (using vector fields that are left invariant) since we will present the material using the left trivialization. Note that we could easily use the right trivialization as well.

The adjoint of G is a representation of G that acts on $\mathfrak{g}$, and is denoted and generically defined as

$$Ad_g : \mathfrak{g} \to \mathfrak{g} \quad (u) \mapsto gug^{-1}$$

where $g \in G$. Since the adjoint is a linear function, we can find a matrix version that operates on the associated Euclidean space generically defined as

$$\mathbf{Ad}_g : E \to E \quad (v) \mapsto \mathbf{Ad}_g v$$

where $\mathbf{Ad}_g$ is the matrix adjoint representation of G.

The adjoint of $\mathfrak{g}$ is a representation of $\mathfrak{g}$ that acts on $\mathfrak{g}$. It is also the Lie bracket $[\cdot, \cdot]$ which we will denoted and generically define as

$$ad_u : \mathfrak{g} \to \mathfrak{g} \quad (v) \mapsto [u, v]$$

where $u \in \mathfrak{g}$. Since the adjoint is a linear function, we can find a matrix version that operates on the associated Euclidean space generically defined as

$$\mathbf{ad}_u : E \to E \quad (v) \mapsto \mathbf{ad}_u v$$

where $u \in \mathfrak{g}$ and $\mathbf{ad}_u$ is the matrix adjoint representation of $\mathfrak{g}$.

As stated in [11], we can defined the differential of the exponential mapping as the 'left trivialized' tangent of the exponential map or as the right trivialized' tangent of the exponential map. These differentials are also commonly called the right and left Jacobians. The right Jacobian, $J_r : E \to GL(E)$, is defined as

$$J_r(v) = \frac{^r\partial \text{Exp}(v)}{\partial v},$$

and the left Jacobian is defined as

$$J_l(v) = \frac{^l\partial \text{Exp}(v)}{\partial v}.$$

Their inverses are defined as

$$J_r^{-1}(v) = \frac{^r\partial \text{Log}(v)}{\partial v}$$

and

$$J_l^{-1}(v) = \frac{^l\partial \text{Log}(v)}{\partial v}.$$

The right Jacobian has the property that for small $\delta v \in E$ given $v \in E$ we get that

$$\begin{aligned} \text{Exp}(\delta v + u) &\approx \text{Exp}(v)\,\text{Exp}(J_r(v)\,\delta v) \\ \text{Exp}(v)\,\text{Exp}(\delta v) &\approx \text{Exp}\left(v + J_r^{-1}(v)\,\delta v\right) \\ \text{Log}(\text{Exp}(v)\,\text{Exp}(\delta v)) &\approx v + J_r^{-1}(v)\,\delta v. \end{aligned}$$

The left Jacobian has the similar property that for small $\delta v \in E$ given $v \in E$ we get that

$$\begin{aligned} \text{Exp}(v + \delta v) &\approx \text{Exp}(J_l(v)\,\delta v)\,\text{Exp}(v) \\ \text{Exp}(\delta v)\,\text{Exp}(v) &\approx \text{Exp}\left(v + J_l^{-1}(v)\,\delta v\right) \\ \text{Log}(\text{Exp}(\delta v)\,\text{Exp}(v)) &\approx v + J_l^{-1}(v)\,\delta v. \end{aligned}$$

The derivation of the left and right Jacobians stems from the Baker-Campbell-Hausdorff formula and can be studied in [9] and [3].

IV. Problem Description

The objective is to estimate the states of multiple dynamic targets given a set of measurements from multiple sensors and a dynamic model without a prior information about the number of targets in the global surveillance region. Let the state of the system at time k be given by $x_k = (g_k, u_k) \in G \times E$ where $g_k \in G$ and $u_k \in E$. We assume the system model to be continuous, near constant velocity, and time invariant defined as

$$\begin{aligned} \dot{g}_k &= \xi_{u_k}(g_k) \\ \dot{u}_k &= w_k^u, \end{aligned}$$

where $\xi_{u_k}(g_k)$ is an infinitesimal generator constructed from u_k, $w_k = \begin{bmatrix} w_k^g \\ w_k^u \end{bmatrix}$ is process noise sampled from a zero-mean, white-noise, Gaussian distribution with covariance Q. Under the assumption that w_k stays constant

over a time period δ_k we can discretize the model to get the discrete system model described as

$$g_k = g_{k^-} \oplus \left(\delta_k u_{k^-} + \delta_k w_k^g + \frac{\delta_k^2}{2} w_k^u \right) \tag{5a}$$

$$u_k = u_{k^-} + \delta_k w_k^u. \tag{5b}$$

This form is similar to the one found in [**?**], and is an exact solution provided that the Lie group is commutative. If the Lie group is not commutative, the exact numeric solution is cumbersome if not impossible to derive, thus we will use the system model as an approximate numerical solution.

The complete discrete system is defined as

$$x_k = f\left(x_{k^-}, w_k, \delta_k\right) \tag{6a}$$

$$y_k = h\left(x_k, v_k\right), \tag{6b}$$

where f is the system model defined in (5), $y_k \in N$ is the output and N is a Lie group, $h : G \to N$ is the observation model, and v_k is measurement noise sampled from a zero-mean, white-noise, Gaussian distribution with covariance R.

The state x is a new Lie group generated from the Cartesian product of G and E. The Lie algebra of $G \times E$ is simply $\mathfrak{g} \times E$ which is isomorphic to $\mathbb{R}^{2n}$ with n being the dimension of G. We will denote this space as E^*. All of the operations previously defined are simply inherited. For example, let $x \in G \times \mathfrak{g}$ and $v = (a, b)^\vee \in E^*$, then

$$\begin{aligned} x \oplus v &= (g, u) \oplus v \\ &= (g, u) \boxplus (a, b) \\ &= \left(g \exp(a), u + b\right), \end{aligned}$$

where $g \in G$, $a \in \mathfrak{g}$, and $u, b \in E$.

V. Lie Group Probability

In this section we define the prior distribution of the state x, the state transition distribution, the likelihood distribution of the measurement and other distributions in a Lie group framework necessary to develop the theory. We follow the convention, notation and terminology from [21], [20], [24] adapted for the Lie group setting. Others have defined these distributions similarly as in [**?**], [4]; however, our derivation is suited for our system model.

We denote the state estimate at time m as $\hat{x}_m \in G \times E$ and the actual state as $x_m \in G \times E$. Let $\delta x_m \in E^*$ denote a local perturbation around $\hat{x}_m$ that is a zero-mean, Gaussian, random variable with error covariance P_m such that

$$x_m = \hat{x}_m \oplus \delta x_m.$$

We denote the prior distribution, or the belief distribution as $p(x_m)$ and define it as

$$\begin{aligned} p(x_m) &= \eta \exp\left(-\frac{1}{2} \left(x_m \ominus \hat{x}_m\right)^\top P_m^{-1} \left(x_m \ominus \hat{x}_m\right) \right) \\ &= \eta \exp\left(-\frac{1}{2} \delta x_m^\top P_m^{-1} \delta x \right), \end{aligned}$$

where η is a normalizing coefficient corresponding to the Gaussian distribution. We will useη to denote any normalizing coefficient. We denote the state transition probability as $p(x_m \mid x_{m^-})$ and define it as

$$p(x_m \mid x_{m^-}) = \eta \exp\left(-\frac{1}{2} \left(x_m \ominus f\right)^\top Q^{-1} \left(x_m \ominus f\right) \right),$$

where $x_m \ominus f = x_m \ominus f\left(x_{m^-}, w_k, \delta_m\right)$.

We can approximate $p(x_m \mid x_{m^-})$ with a Gaussian distribution by using the first order Taylor series expansion of $x_m \ominus f$ and treating δx_{m^-} and w_m as perturbations; this is known as linearizing. Doing so, yields

$$p(x_m \mid x_{m^-}) \approx \eta \exp\left(-\frac{1}{2} \left(x_m \ominus \tilde{f}\right)^\top \bar{Q}^{-1} \left(x_m \ominus \tilde{f}\right) \right),$$

where

$$x_m \ominus \tilde{f} = x_m \ominus f\left(\hat{x}_m, 0, \delta_m\right) + F_m \delta x_{m^-} + G_m w_m,$$

$$\begin{aligned} F_m &= \left. \frac{\partial x_m \ominus f\left(x_{m^-}, w_k, \delta_m\right)}{\partial x_{m^-}} \right|_{\hat{x}_{m^-}, \hat{w}_m, \delta_m} \\ &= \begin{bmatrix} \mathbf{Ad}_{\mathrm{Exp}\left(\delta_m \hat{u}_{m^-}\right)^{-1}} \\ J_r\left(\delta_m \hat{u}_{m^-}\right) \end{bmatrix}, \end{aligned}$$

$$\begin{aligned} G_m &= \left. \frac{\partial x_m \ominus f\left(x_{m^-}, w_k, \delta_m\right)}{\partial w_m} \right|_{\hat{x}_{m^-}, \hat{w}_m, \delta_m} \\ &= \begin{bmatrix} J_r\left(\delta_m \hat{u}_{m^-}\right) \delta_m & J_r\left(\delta_m \hat{u}_{m^-}\right) \frac{\delta_m^2}{2} \\ I \delta_m & 0 \end{bmatrix}, \end{aligned}$$

and $\hat{w}$ is the expected value of the process noise which is zero.

Let $\bar{p}(x_m)$ denote the probability of x_m after state propagation. It is calculated as

$$\bar{p}(x_m) = \int p(x_m \mid x_{m^-}) \, p(x_{m^-}) \, dx_{m^-}.$$

We approximate $\bar{p}(x_m)$ by linearizing $p(x_m \mid x_m)$ so that $\bar{p}(x_m)$ is a Gaussian distribution which is parametrized by the new state estimate $\hat{\bar{x}}_m$ and error covariance $\bar{P}_m$. We will not include the tedious derivation, but refer the reader to [24]. The new state estimate and error covariance is calculated as

$$\hat{\bar{x}}_m = f\left(\hat{x}_{m^-}, \hat{w}_m, \delta_m\right) \tag{7a}$$

$$\bar{P}_m = F_m P_{m^-} F_m^\top + G_m Q G_m^\top; \tag{7b}$$

this calculation is called the propagation or the prediction step.

The probability of a measurement conditioned on the state is called the measurement probability or likelihood denoted as $p\left(y_m \mid x_m\right)$, and defined as

$$p\left(y_m \mid x_m\right) = \eta \exp\left(-\frac{1}{2} e^\top R^{-1} e\right),$$

where

$$e = y_m \ominus h\left(x_m, v_m\right).$$

We can linearize the measurement probability to get

$$p\left(y_m \mid x_m\right) \approx \eta \exp\left(-\frac{1}{2} \tilde{e}^\top R^{-1} \tilde{e}\right)$$

where

$$\tilde{e} = y_m \ominus h\left(\hat{x}_m, \hat{v}_m\right) + H_m \delta x_m + V_m v_m,$$

$$\begin{aligned} H_m &= \left.\frac{\partial y_m \ominus h\left(x_m, v_m\right)}{\partial x_m}\right|_{\hat{x}_m, \hat{v}_m}, \\ &= \left.\frac{\partial \log\left(\hat{e}\right)}{\partial z} \frac{\partial z}{\partial h^{-1}\left(x_m, v_m\right)} \frac{\partial h\left(x_m, v_m\right)}{\partial x_m}\right|_{\hat{x}_m, \hat{v}_m} \\ &= J_r^{-1}\left(\hat{e}\right) \mathbf{Ad}_{y_m^{-1}} \left.\frac{\partial h\left(x_m, v_m\right)}{\partial x_m}\right|_{\hat{x}_m, \hat{v}_m} \end{aligned}$$

$$\begin{aligned} V_m &= \left.\frac{\partial y_m \ominus h\left(x_m, v_m\right)}{\partial v_m}\right|_{\hat{x}_m, \hat{v}_m}, \\ &= J_r^{-1}\left(\hat{e}\right) \mathbf{Ad}_{y_m^{-1}} \left.\frac{\partial h\left(x_m, v_m\right)}{\partial v_m}\right|_{\hat{x}_m, \hat{v}_m} \end{aligned}$$

$\hat{e} = y_m \ominus h\left(\hat{x}_m, \hat{v}_m\right)$ and $\hat{v}_m$ is the expected value of the measurement noise which is zero.

The probability of the state given a measurement is called the posterior denoted $p\left(x \mid y\right)$ and defined as

$$p\left(x \mid y\right) = \frac{p\left(y \mid x\right) \bar{p}\left(x\right)}{p\left(y\right)}.$$

We improve the state estimate $\hat{x}_m$ by maximizing the posterior distribution. This process is outlined in [24] which gives us the standard Kalman filter update step:

$$\text{Innovation :} \quad \hat{e} = y_m \ominus h\left(\hat{x}_m, \hat{v}_m\right) \tag{8a}$$

$$\text{Innovation cov. :} \quad S = H_m \bar{P}_m H_m^\top + V_m R V_m^\top \tag{8b}$$

$$\text{Kalman gain :} \quad K = \bar{P}_m H_m^\top S^{-1} \tag{8c}$$

$$\text{Observed error :} \quad \delta x_m = K\hat{e} \tag{8d}$$

$$\text{State update :} \quad \hat{x}_m = \hat{\bar{x}}_m \oplus \delta x_m \tag{8e}$$

$$\text{Cov. update :} \quad P_m = P_m - KSK^\top. \tag{8f}$$

Depending on the data associating filter used, the update step will be different; however, the underlying basic idea is the same which will require computing the Jacobians F_m, G_m, H_m and V_m.

VI. RANSAC

The objective of RANSAC is to estimate the parameters of a model given a set of measurements which contains both false and true measurements. We follow a scheme similar to the one presented in [25]; that is, we perform RANSAC on every cluster seeding it with a measurement obtained from the current time. This process is described for a single cluster.

Let Y denote the set of measurements pertaining to a cluster, and let $y_{0:k}$ be a minimum subset of Y that contains a measurement from the current time k and other measurements randomly sampled from different times such that the system can be observed by the measurements in $y_{0:k}$. Using the measurements $y_{0:k}$, we create a model hypothesis by estimating a current hypothetical state x_k that fit the measurements according to they system model described in (5). The hypothetical state x_k is estimated using a log maximum likelihood estimate (LMLE) method described in section VII. Once x_k is estimated, we construct a consensus set for the model hypothesis. The consensus set contains all of the measurements that are inliers to the model hypothesis. An inlier is a measurement that is within some distance of the estimated measurement obtained from x_k. Let $d_I : N \times N \to \mathbb{R}$ be a metric on the measurement space N, as described in [15], and let $\tau_I \in \mathbb{R}$ be a threshold, then if

$$d_I\left(y_m^j, \hat{y}_m\right) < \tau_I,$$

where $\hat{y}_m$ is the estimated measurement at time m, and y_m^j is the j^{th} measurement obtained at time m, then the measurement y_m^j is considered an inlier to the model hypothesis and is added to the consensus set. The estimated measurement $\hat{y}_m$ is calculated using model inversion described in subsection VII-A. The metric that we use is defined as

$$d_I\left(y_m^j, \hat{y}_m\right) = \left(y_m^i \ominus \hat{y}_m\right)^\top \left(R_{k:m}^j\right)^{-1} \left(y_m^i \ominus \hat{y}_m\right),$$

where $R_{k:m}^j$ is defined in (10). For help choosing τ_I, see section (VIII).

This process is repeated at most ℓ times so that we have at most ℓ model hypothesis from each cluster. The model hypothesis with the largest consensus set from each cluster is kept and used to create a model using a filtering method that incorporates all of the measurements in the consensus set to calculate a current state estimate and error covariance. The size of the consensus set is not necessarily the cardinality of the consensus set. We define the size of the consensus set as the number of measurements from different sources at each time step. For example, if a source provided ten measurements at time k and four of them were added to the consensus set, all four of them would count as one. The purpose of this definition is to ensure that

one source is not weighted more heavily than another. RANSAC will terminate early if the size of a consensus set is greater than or equal to the threshold τ_E, and the measurements in a model hypothesis's consensus set are removed from the data tree if the model hypothesis become a model.

The filtering method that we employ uses the Kalman propagation step described in equation (7) and an update step using centralized measurement fusion described in section IX. The centralized measurement fusion allows multiple measurements from different sources to be fused together.

VII. Log Maximum Likelihood Estimate

We use LMLE to generate model hypotheses in RANSAC. A model hypotheses is created by taking a subset of measurements from a cluster $y_{0:k} \subseteq Y$ which contains one measurement from the current time and other measurements which are randomly sampled from different times with enough measurements in order to estimate the state, i.e. the $|y_{0:k}|$ is greater than or equal to the observability index of the system. Using the measurements $y_{0:k}$ we employ the LMLE method to calculate a hypothetical state estimate, i.e. a possible state estimate of an actual target.

LMLE estimates the joint probability of all the states in the time window, $x_{0:k}$, conditioned on the measurements $y_{0:k}$ by maximizing the posterior probability. The posterior probability is

$$p\left(x_{0:k} \mid y_{0:k}\right).$$

Using Bayes rule we get

$$p\left(x_{0:k} \mid y_{0:k}\right) = \frac{p\left(y_k \mid x_{0:k}, y_{0:k^-}\right) p\left(x_{0:k} \mid y_{0:k^-}\right)}{p\left(y_k \mid y_{0:k^-}\right)}.$$

Since $p\left(y_k \mid y_{0:k^-}\right)$ does not depend on the parameters we are trying to estimate, we can replace it with a constant $\frac{1}{\eta}$ so that the posterior becomes

$$p\left(x_{0:k} \mid y_{0:k}\right) = \eta p\left(y_k \mid x_{0:k}, y_{0:k^-}\right) p\left(x_{0:k} \mid y_{0:k^-}\right).$$

Under the assumption that the system is a first order Markov process, we can simplify the posterior to

$$p\left(x_{0:k} \mid y_{0:k^-}\right) = \eta p\left(y_k \mid x_k\right) p\left(x_k \mid x_{k^-}\right) p\left(x_{0:k^-} \mid y_{0:k^-}\right).$$

Repeating the process recursively, we get

$$p\left(x_{0:k} \mid y_{0:k}\right) = \eta p\left(x_0\right) \prod_{m=1}^{k} \left[p\left(x_m \mid x_{m^-}\right)\right] \prod_{m=0}^{k} \prod_{j=1}^{\ell(m)} \left[p\left(y_m^j \mid x_m\right)\right],$$

where $\ell\left(m\right)$ is the number of measurements in $y_{0:k}$ that were received at time m.

Maximizing the posterior is equivalent to minimizing the negative log posterior. The negative log posterior is

$$\log\left(-p\left(x_{0:k} \mid y_{1:k}\right)\right) = \log\left(-\eta\right) + \log\left(-p\left(x_0\right)\right) + \sum_{m=1}^{k} \log\left(-p\left(x_m \mid x_{m^-}\right)\right) + \sum_{m=0}^{k} \sum_{j=1}^{\ell(m)} \log\left(-p\left(y_m^j \mid x_m\right)\right).$$

We can drop the term $\log\left(-\eta\right)$ since it will have no impact on the optimization problem. The term $\log\left(-p\left(x_0\right)\right)$ is negligible compared to the rest of the optimization problem since its covariance is big indicating that we do not know the initial state x_0. We thus simplify the negative log posterior to

$$\sum_{m=1}^{k} \log\left(-p\left(x_m \mid x_{m^-}\right)\right) + \sum_{m=0}^{k} \sum_{j=1}^{\ell(m)} \log\left(-p\left(y_m^j \mid x_m\right)\right).$$

Since we are only interested in estimating the current state x_k and because there is no input to the system, we simplify the negative log posterior to

$$\sum_{m=0}^{k} \sum_{j=1}^{\ell(m)} \log\left(-p\left(y_m^j \mid x_k\right)\right).$$

Using these simplifications, the minimization problem is defined as

$$\arg\min_{x_k} \left(\sum_{m=0}^{k} \sum_{j=1}^{\ell(m)} \log\left(-p\left(y_m^j \mid x_k\right)\right) \right).$$

This requires constructing a map from the current state x_k to a previous state x_m which is done by inverting the model.

A. Model Inversion

Given the current state x_k we are interested in calculating the state and output at time $m < n$. We can easily invert the system model defined in (5). Ignoring the noise terms, we can propagate the pose of the system g_m to g_k by

$$g_k = g_m \exp\left(\delta_{m:k} u\right).$$

Solving for g_m yields

$$\begin{aligned} g_m &= \exp^{-1}\left(\delta_{m:k} u\right) g_k \\ &= \exp\left(-\delta_{m:k} u\right) g_k \\ &= \exp\left(\delta_{k:m} u\right) g_k. \end{aligned}$$

Thus, the inverse of the system model $f^{-1}\left(x_k, w_k, \delta_{k:m}\right)$ is defined as

$$g_m = g_k \oplus \left(\delta_{k:m} u_k + \delta_{k:m} w_k^g + \frac{\delta_{k:m}^2}{2} w_k^u \right) \tag{9a}$$

$$u_m = u_k + \delta_{k:m} w_k^u. \tag{9b}$$

and the output at time m is

$$y_m = h\left(f^{-1}, v_m\right),$$

where

$$f^{-1} = f^{-1}\left(x_k, w_k, \delta_{k:m}\right)$$

We are interested in the linearized likelihood $p\left(y_m^j \mid x_k\right)$. Since the likelihood is conditioned on the state, the state is not a random variable. This simplifies the Taylor series of

$$y_m^j \ominus h\left(f^{-1}\left(x_k, w_k, \delta_{k:m}\right), v_m\right),$$

which is

$$y_m^j \ominus h\left(f^{-1}, \hat{v}_m\right) + G_{k:m}^j w_k + V v_m,$$

where

$$G_{k:m}^j = \left.\frac{\partial y_m \ominus h\left(f^{-1}, v_m\right)}{\partial w_k}\right|_{x_k, \hat{w}_k, \hat{v}_m} =$$

$$= J_r^{-1}\left(\hat{e}_{k:m}\right) \mathbf{Ad}_{y_m^{j-1}} \left.\frac{\partial h\left(f^{-1}, v_m\right)}{\partial w_k}\right|_{x_k, \hat{w}_k, \hat{v}_m},$$

and

$$\hat{e}_{k:m}^j = y_m^j \ominus h\left(f^{-1}\left(x_k, \hat{w}_k, \delta_{k:m}\right), \hat{v}_m\right)$$

which is the expected value of $p\left(y_m^j \mid x_k\right)$. The covariance of $p\left(y_m^j \mid x_k\right)$ is

$$R_{k:m}^j = G_{k:m}^j Q \left(G_{k:m}^j\right)^\top + V_m R V_m^\top; \qquad (10)$$

therefore, the linearized probability of $p\left(y_m \mid x_k\right)$ is

$$\eta \exp\left(-\frac{1}{2}\left(\hat{e}_{k:m}^j\right)^\top \left(R_{k:m}^j\right)^{-1} \hat{e}_{k:m}^j\right). \qquad (11)$$

Using the linearized probability, we can write the minimization problem as

$$\arg\min_{x_k}\left(\sum_{m=0}^{k}\sum_{j=1}^{\ell(m)}\left(\hat{e}_{k:m}^j\right)^\top \left(R_{k:m}^j\right)^{-1} \hat{e}_{k:m}^j\right). \qquad (12)$$

There are many methods that can be used to solve the optimization problem. If the system is linear, the optimization problem reduces to the one described in [19]. If the system is non-linear, the optimization problem can be solved using the Gauss-newton method or some other optimization method. Depending on the method used, you may need compute the derivative of $y_m^j \ominus h\left(f^{-1}\left(x_k, w_k, \delta_{k:m}\right), v_m\right)$ with respect to the state. We denote this derivative as $F_{k:m}$ which is defined as

$$F_{k:m} = \left.\frac{\partial y_m \ominus h\left(f^{-1}, v_m\right)}{\partial x_k}\right|_{\hat{x}_k, \hat{w}_k, \hat{v}_m}$$

$$= J_r^{-1}\left(\hat{e}_{k:m}\right) \mathbf{Ad}_{y_m^{-1}} \left.\frac{\partial h\left(f^{-1}, v_m\right)}{\partial x_k}\right|_{\hat{x}_k, \hat{w}_k, \hat{v}_m}.$$

If the optimization is non-linear, a scheme to seed the optimization will drastically help. In the examples that we will present later, we will show same possible seeding methods.

Some of these calculations can be very costly. It may be worth increasing computation at a cost of quality by letting $R_{k:m} = R$.

VIII. Validation Region

When new measurements are received, they are tested to see if they are inside a model's validation region. The validation region is a volume around an estimated state used for data association. Measurements that fall within the validation region are associated to the model and are used to update the model. We will use the validation region described in [**?**]. We assume that the probability of the j^{th} measurement of dimension n conditioned on the previous state, $p\left(y_k^j \mid x_{k^-}\right)$, is Normally distributed with mean $\hat{y}_k = h\left(f\left(\hat{x}_{k^-}, \delta_k\right)\right)$ and covariance S where S is the innovation covariance described in equation (8). Let z_k^j be the random variable defined as

$$z_k^j = S^{-\frac{1}{2}} y_k^j \ominus \hat{y}_k,$$

then z_k^j is a standard multivariate Gaussian distribution. For clarity, we will drop the subscripts and superscripts. Let $d_v : N \times N \to \mathbb{R}$ be the metric defined as

$$d_v\left(y, \hat{y}\right) = z^\top z$$
$$= \left(y \ominus \hat{y}\right)^\top S^{-1}\left(y \ominus \hat{y}\right).$$

Note that the metric d_v is simply the sum of the square of n Gaussian distributions. Thus, the values of the metric d_v form a chi-square distribution with probability density function (PDF)

$$p\left(d\left(y, \hat{y}\right)\right) = \frac{d\left(y, \hat{y}\right)^{n/2-1} \exp^{-n/2}}{2^{n/2}\Gamma\left(n/2\right)},$$

where $\Gamma\left(n/2\right)$ is the gamma function.

The validation region is defined as

$$\nu\left(\hat{y}, \gamma\right) = \left\{y \in N \,:\, d\left(y, \hat{y}\right) \leq \gamma\right\},$$

where parameter γ is called the gate threshold. The gate probability P_G is the probability that a measurement produced by a target is within the validation region. This probability is defined as $p\left(d\left(y, \hat{y}\right) \leq \gamma\right)$, thus P_G is the value of the cumulative distribution function (CDF) of the chi-square distribution with parameter γ. New measurements that fall within the validation region of a model are used to update the corresponding model using centralized measurement fusion described in section (IX).

Some data association techniques require knowing the volume of the validation region V_{vol}. The volume is defined as

$$V_{vol} = c_n \gamma^{n/2} |S|^{1/2}, \tag{13}$$

where c_n is the volume of the unit hypersphere of dimension n (dimension of the measurement space) calculated as

$$c_n = \frac{\pi^{n/2}}{\Gamma(n/2+1)}.$$

IX. Centralized Measurement Fusion

Validated measurements, or measurements associated to a model, can come from multiple different measurement sources at different times. In order to update the state estimate, we use the centralized measurement fusion method, as discussed in [2] and [14], which assumes that measurements are statistically independent. We briefly review this method here only to adapt it to the Lie group setting. Let N_s denote the number of measurement sources, $\theta_m^i = \{y_m^1, y_m^2, \cdots, y_m^n\}$ denote the set of measurements associated with the model that were received at time m from the i^{th} measurement source, β_m^j be the weight associated with measurement $y_m^j \in \theta_m^i$, Z^i be defined as

$$Z^i = \begin{cases} \sum_{j=1}^{n} \beta^j H_m^\top \left(R^i\right)^{-1} y_m^j \ominus h(x_m) & \text{if } \left|\theta_m^i\right| > 0 \\ 0 & \text{else} \end{cases},$$

and P_m^i be the updated measurement covariance after incorporating only the measurements from the i^{th} source, then the covariance update is

$$P_m^{-1} = \bar{P}_m^{-1} + \sum_{i=1}^{N_s} \left(\left(P_m^i\right)^{-1} - \bar{P}_m^{-1} \right),$$

and the state update is

$$\hat{x}_m = \hat{\bar{x}}_m \oplus \left(P_m \sum_{i=1}^{N_s} Z^i \right).$$

Recall that H_m is the Jacobian of the measurement mode, R^i the measurement covariance of the i^{th} measurement source, and $\bar{P}_m$ the error covariance after propagation but before any measurement updates. The weights β_m^j are calculated from the data association filter used such as a PDAF.

X. Track to Track Fusion

In the case where R-RANSAC initializes two different tracks and later decides that the two tracks represent the same target, the tracks need to be fused (merged) together. Calculating the cross covariance needed to optimally fuse the two tracks can be computationally complex and time expensive, or in our case unknown. Fortunately, the covariance intersection (CI) method can be used to fuse two tracks together without calculating the cross covariance. A good introduction to the CI method is presented in [**?**]. Bar-Shalom, in [**?**], pointed out that the original CI method is so conservative in its estimation that it can indicate degradation of the state estimate due to fusion. An improved version of the CI method was presented in [**?**] called the sampled covariance intersection (SCI) method. This is the method we will follow. In order to see if two tracks need to be fused together, we use a track association method prescribed in [**?**]. In this section we will briefly present the track association and fusion methods adapted for the Lie group setting.

A. Track Association

Let $\hat{x}^i$ denote the state estimate of the i^{th} track and P^i be the corresponding error covariance. The estimation error between two tracks is denoted Δ^{ij} and defined as

$$\Delta^{ij} = \hat{x}^i \ominus \hat{x}^j.$$

Assuming that the two tracks are independent, the covariance of the estimation error is

$$T^{ij} = P^i + P^j.$$

We next define a metric $d_T : (G \times \mathfrak{g}) \times (G \times \mathfrak{g}) \to \mathbb{R}$ to be

$$d_T\left(\hat{x}^i, \hat{x}^j\right) = \left(\Delta^{ij}\right)^\top \left(T^{ij}\right)^{-1} \Delta^{ij}.$$

If $d_T(\cdot,\cdot) < \tau_S$ where $\tau_S \in \mathbb{R}$, then the two tracks are deemed to be the same and should be merged.

B. Track Fusion

Let $\hat{x}^1, \hat{x}^2, \cdots, \hat{x}^n$ denote the state estimates of the tracks that need to be fused together and P^i be their corresponding error covariance. The fused estimate in SCI is calculated as if the tracks to be fused are independent; therefore,

$$P^{-1} = \sum_{i=1}^{n} \left(P^i\right)^{-1}$$

$$\hat{x}_{SCI} = P \left(\sum_{i=1}^{n} \left(P^i\right)^{-1} \hat{x}^i \right).$$

The error covariance P^{-1} is overly optimistic, so we must adjust the size as follows:

- Generate N (about 100 for a good distribution) random samples x^j, $j = 1, 2, \ldots, N$ from $x \sim \mathcal{N}(0, P)$
- Find

$$r_{\max} = \max_{j=1,\ldots,N} \frac{\left(x^j\right)^\top P^{-1} x^j}{\max\limits_{j=1,\ldots,N} \left\{ \left(x^j\right)^\top P^{-1} x^j \right\}}$$

and

$$r_{\min} = \min_{j=1,\dots,N} \frac{\left(x^j\right)^\top P^{-1} x^j}{\max_{j=1,\dots,N} \left\{\left(x^j\right)^\top P^{-1} x^j\right\}}$$

- Then set

$$P_{SCI} = \frac{P}{u r_{\min} + (1-u)\, r_{\max}}$$

where $u \in [0,1]$ is used to adjust the performance of the SCI algorithm. When $u = 1$, the fused covariance is conservative and when $u = 0$, the fused covariance is optimistic. The authors of [**?**] received good results by setting $u = 0.5$.

The fused track will have the state estimate $\hat{x}_{SCI}$ with error covariance P_{SCI}.

XI. Model Likelihood

R-RANSAC stores up to $\mathcal{M}$ models. Some of the models might not represent actual targets and could have been initialized from clustered false measurements. To account for this, we need a way to measure how likely a model represents an actual target. We will refer to this measure or probability as the model's likelihood. In the previous versions of R-RANSAC, the inlier ratio and the model's lifetime were used to indicate the model's likelihood. In this section, we present a different approach using a binary Bayes filter with static states.

Let P_D^i denote the probability that the i^{th} measurement source produced a true measurement of the target during the sensor scan, let P_G^i denote the probability that the true measurement was validated, i.e. associated with the target during the data association process, then the probability that a true measurement was received and validated is $P_D^i P_G^i$. Let λ^i denote the spatial density of the expected number of false measurements from the i^{th} measurement source during a sensor scan. For example, let V_s^i denote the volume of a surveillance region and Λ^i denote the expected number of false measurements per sensor scan from the i^{th} measurement source, then $\lambda^i = \frac{\Lambda^i}{V_s^i}$. The expected number of false measurements within a validation region is $\lambda^i V_{vol}$ where V_{vol} is the volume of the validation region defined in (13) and is depending on the measurement source and state estimate.

Let θ_m^i denote the number of validated measurements from the i^{th} measurement source at time m, $\theta_{m:k}^i$ denote the number of validated measurements from the i^{th} measurement source from time m to time k where $m < k$, and $\theta_{m:k}$ the number of all validated measurements from all the measurement sources from time m to time k. Let z be a Bernoulli random variable with $z = 1$ meaning that the model represents an actual target and $z = 0$ meaning that the model doesn't represent a target. For notation purposes, we will denote $p(z = 1)$ as $p(z)$ and $p(z = 0)$ as $p(\neg z)$.

We are interested in calculating the probability that the model represents an actual target conditioned on the validated measurements. We denote this probability as $p(z \mid \theta_{m:k})$. Using Bayes rule we get

$$\begin{aligned} p(z \mid \theta_{m:k}) &= \frac{p(\theta_k \mid z, \theta_{m:k^-})\, p(z \mid \theta_{m:k^-})}{p(\theta_k \mid \theta_{m:k^-})} \\ &= \frac{p(\theta_k \mid z)\, p(z \mid \theta_{m:k^-})}{p(\theta_k \mid \theta_{m:k^-})}. \end{aligned}$$

The probability $p(\theta_k \mid \theta_{m:k^-})$ can be difficult to calculate; however, we can work around needing to know this probability by using a log odds ratio. For this method, we need $p(\neg z \mid \theta_{m:k})$ which is

$$p(\neg z \mid \theta_{m:k}) = \frac{p(\theta_k \mid \neg z)\, p(\neg z \mid \theta_{m:k^-})}{p(\theta_k \mid \theta_{m:k^-})}.$$

Taking the log (base 10) ratio of the probabilities $p(z \mid \theta_{m:k})$ and $p(\neg z \mid \theta_{m:k})$, we get

$$\begin{aligned} \log\left(\frac{p(z \mid \theta_{m:k})}{p(\neg z \mid \theta_{m:k})}\right) &= \log\left(\frac{p(\theta_k \mid z)\, p(z \mid \theta_{m:k^-})}{p(\theta_k \mid \neg z)\, p(\neg z \mid \theta_{m:k^-})}\right) \\ &= \log\left(\frac{p(\theta_k \mid z)}{p(\theta_k \mid \neg z)}\right) + \sum_{n=m}^{k^-} \log\left(\frac{p(\theta_n \mid z)}{p(\theta_n \mid \neg z)}\right). \end{aligned}$$

Whenever new validated measurements are received, we calculate the term

$$l_k = \log\left(\frac{p(\theta_k \mid z)}{p(\theta_k \mid \neg z)}\right)$$

and add it to the previous log odds ratio

$$\begin{aligned} l_{k^-} &= \log\left(\frac{p(z \mid \theta_{m:k^-})}{p(\neg z \mid \theta_{m:k^-})}\right) \\ &= \sum_{n=m}^{k^-} \log\left(\frac{p(\theta_n \mid z)}{p(\theta_n \mid \neg z)}\right). \end{aligned}$$

The term l_k can be written as

$$\log\left(\frac{p(\theta_k \mid z)}{p(\theta_k \mid \neg z)}\right) = \sum_{i \in 1}^{s} \log\left(\frac{p\left(\theta_k^i \mid z\right)}{p\left(\theta_k^i \mid \neg z\right)}\right),$$

where s is the number of measurement sources. Thus we are left with calculating $p\left(\theta_k^i \mid z\right)$ and $p\left(\theta_k^i \mid \neg z\right)$ for each measurement source. We assume that a measurement source can only produce at most one true measurement during a sensor scan. Therefore, we have the two possibilities for $p\left(\theta_k^i \mid z\right)$, that all θ_k^i measurements are false or all but one are false. Thus

$$\begin{aligned} p\left(\theta_k^i \mid z\right) = P_G^i P_D^i \exp\left(\lambda^i V_{vol}\right) \frac{\left(\lambda^i V_{vol}\right)^{\theta_k^i - 1}}{\left(\theta_k^i - 1\right)!} \\ + \left(1 - P_G^i P_D^i\right) \exp\left(\lambda^i V_{vol}\right) \frac{\left(\lambda^i V_{vol}\right)^{\theta_k^i}}{\left(\theta_k^i\right)!}. \end{aligned}$$

There is only one possibility for $p\left(\theta_k^i \mid \neg z\right)$, that is the probability of all of the measurements being false

$$p\left(\theta_k^i \mid \neg z\right) = \exp\left(\lambda^i V_{vol}\right) \frac{\left(\lambda^i V_{vol}\right)^{\theta_k^i}}{\left(\theta_k^i\right)!}.$$

Taking the ratio of the two, we get

$$\frac{p\left(\theta_k^i \mid z\right)}{p\left(\theta_k^i \mid \neg z\right)} = 1 + P_G^i P_D^i \left(\frac{\theta_k^i}{\lambda^i V_{vol}} - 1\right).$$

The model likelihood, $p\left(z \mid \theta_{m:k}\right)$, can be recovered from the log odds ratio by

$$p\left(z \mid \theta_{m:k}\right) = 1 - \frac{1}{1 + 10^{l_k}}.$$

Note that if a model is outside the surveillance region of the measurement source, then P_D should be zero and $p\left(z \mid \theta_{m:k}\right)$ will not be affected since there won't be any measurements and the update to the log odds ratio is 0. In the case that the target leaves the global surveillance region, we need a way to kill the model in order to make room for other models to be initialized. Thus, we assume that a model that has not received a measurement within τ_α seconds is no longer the the GSR and will be removed. A model is considered a good model if the $p\left(z \mid \theta_{m:k}\right) > \tau_\rho$; otherwise, it is considered a poor model. The parameters λ^i and P_D^i can be easily estimated and P_G^i is defined by the user.

XII. Example $\mathbb{R}^2$

Euclidean spaces equipped with the standard topology and the group operation of addition form a Lie group. We will give an example of this using $\mathbb{R}^2$. Let $\mathbb{R}^2$ be a group equipped with addition. Note that this is an abelian group; thus the adjoint map is the identity map. Its corresponding Lie algebra is simply $\mathbb{R}^2$. Therefore, the exponential map and its inverse is simply the identity map. This implies that the left and right Jacobians of the exponential map is also the identity map. The box plus/minus and o plus/minus mapping are trivially derived; for example

$$g \oplus u = g + u$$
$$g_1 \ominus g_2 = g_1 - g_2$$

where $g, g_1, g_2 \in G = \mathbb{R}^2$ and $u \in \mathfrak{g} = \mathbb{R}^2$.

The system model is then

$$g_k = g_{k^-} + \delta_k \left(u_{k^-} + w_k^g\right) + \frac{\delta_k^2}{2} w_k^u$$
$$u_k = u_{k^-} + \delta_k w_k^u.$$

We can write this in the familiar matrix notation as

$$x_k = A_k x_{k^-} + B_k w_k,$$

where $x_k = \left[g_k^\top, u_k^\top\right]^\top$,

$$A_k = \begin{bmatrix} I_{2\times 2} & \delta_k I \\ 0 & I_{2\times 2} \end{bmatrix},$$

and

$$B_k = \begin{bmatrix} \delta_k I & \frac{\delta_k^2}{2} I \\ 0 & I \end{bmatrix}.$$

XIII. Constrained Problem for SE(n)

The LMLE presented in the previous section relied on the measurement being an element of the group. This is not always possible depending on the sensors available. In this section we assume that the system can be modeled on the manifold $SE\left(n\right)$, only the position of the system is observable, system has constant angular and translational velocity.

Let $x \in SE\left(n\right)$, then x can be written in matrix form as

$$x = \begin{bmatrix} R & P \\ 0_{n\times 1} & 1 \end{bmatrix},$$

where $R \in SO\left(n\right)$ represents the attitude and $P \in \mathbb{R}^n$ represents the position.

Let $u \in \mathfrak{se}\left(n\right)$, then u can be written in matrix form as

$$u = \begin{bmatrix} \omega & \rho \\ 0_{n\times 1} & 0 \end{bmatrix},$$

where $\omega \in \mathfrak{so}\left(n\right)$ represents the angular velocity in the body frame, and $\rho \in \mathbb{R}^n$ denotes the translational velocity in the body frame.

The system can be described as

$$x_k = x_{k-1} \boxplus \left(\delta_k u + \delta_k w_k\right) \tag{14a}$$
$$y_k = x_k C + v_k, \tag{14b}$$

where

$$C = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

so that $y_k \in \mathbb{R}^{n+1}$. We can write the output at time k as

$$\begin{aligned} y_k &= x_{k-1} \boxplus \left(\delta_k u + \delta_k w_k\right) C + v_k \\ &= x_{k-1} \exp\left(\delta_k u + \delta_k w_k\right) C + v_k. \end{aligned}$$

Since $w_k \in \mathfrak{se}\left(n\right)$ we can represent it as

$$w_k = \begin{bmatrix} \nu & \eta \\ 0 & 0 \end{bmatrix},$$

where $\omega \in \mathfrak{so}\left(n\right)$ and $\eta \in \mathbb{R}^n$.

Using the exponential map, we can write

$$x_{k-1} \exp\left(\delta_k u + \delta_k w_k\right)$$

as

$$\begin{bmatrix} R_{k-1} & P_{k-1} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \exp\left(\delta_k\left(\omega_k+\nu_k\right)\right) & V\left(\delta_k\left(\omega_k+\nu_k\right)\right)\delta_k\left(\rho_k+\eta_k\right) \\ 0 & 1 \end{bmatrix}$$
$$= \begin{bmatrix} R_{k-1}\exp\left(\delta_k\left(\omega_k+\nu_k\right)\right) & P_{k-1}+R_{k-1}V\left(\delta_k\left(\omega_k+\nu_k\right)\right)\delta_k\left(\rho_k+\eta_k\right) \\ 0 & 1 \end{bmatrix}$$

Let $v \in \mathfrak{so}(n)$, $\theta = \|v\|$, and I be the identity matrix, then

$$V(v) = I + \left(\frac{1-\cos(\theta)}{\theta^2}\right)v + \left(\frac{\theta-\sin(\theta)}{\theta^3}\right)v^2.$$

Using the expanded from, we get that

$$x_{k-1}\exp\left(\delta_k u + \delta_k w_k\right)C = P_{k-1}+R_{k-1}V\left(\delta_k\left(\omega_k+\nu_k\right)\right)\delta_k\left(\rho_k+\eta_k\right),$$

thus

$$y_k = P_{k-1} + R_{k-1}V\left(\delta_k\left(\omega_k+\nu_k\right)\right)\delta_k\left(\rho_k+\eta_k\right) + v_k.$$

Notice that ν_k acts like a small deviation from ω_k. We can approximate $V\left(\delta_k\left(\omega_k+\nu_k\right)\right)$ by taking it's Taylor series expansion. The series form for $V(v)$ is

$$V = I + \frac{v}{2!} + \frac{v^2}{3!} + \frac{v^3}{4!} + \cdots,$$

thus

$$\begin{aligned}\frac{\partial V}{\partial v} &= \frac{I}{2!} + \frac{2v}{3!} + \frac{3v^2}{4!} + \frac{4v^3}{5!} + \cdots \\ &= \frac{I}{2} + \sum_{i=0}^{\infty}\frac{(2i+2)\,v^{2i+1}}{(2i+3)!} + \sum_{i=0}^{\infty}\frac{(2i+3)\,v^{2i+2}}{(2i+4)!} \\ &= \frac{I}{2} + \sum_{i=0}^{\infty}\frac{(2i+2)\,(-1)^i\,\theta^{2i}}{(2i+3)!}v + \sum_{i=0}^{\infty}\frac{(2i+3)\,(-1)^i\,\theta^{2i}}{(2i+4)!}v^2.\end{aligned}$$

Notice that

$$\sum_{i=0}^{\infty}\frac{(2i+2)\,(-1)^i\,\theta^{2i}}{(2i+3)!}$$

is similar to the Taylor Series of the $\sin$ function. We are going to manipulate it to get a closed form by first multiplying it by $(-1)(-1)\frac{\theta^2}{\theta^2}$ and adding and subtracting $\frac{\sin(\theta)}{\theta^3}$ to it.

$$\begin{aligned}(-1)(-1)\frac{\theta^3}{\theta^3}\sum_{i=0}^{\infty}\frac{(2i+2)\,(-1)^i\,\theta^{2i}}{(2i+3)!} &= \frac{-1}{\theta^3}\sum_{i=0}^{\infty}\frac{(2i+2)\,(-1)^{i+1}\,\theta^{2i+3}}{(2i+3)!} \\ &= \frac{\sin(\theta)}{\theta^3} - \frac{\sin(\theta)}{\theta^3} - \frac{1}{\theta^3}\sum_{i=0}^{\infty}\frac{(2i+2)\,(-1)^{i+1}\,\theta^{2i+3}}{(2i+3)!} \\ &= \frac{\sin(\theta)}{\theta^3} - \frac{1}{\theta^3}\left(\sum_{i=0}^{\infty}\frac{(-1)^i\,x^{2i+1}}{(2n+1)!} + \sum_{i=0}^{\infty}\frac{(2i+2)\,(-1)^{i+1}\,\theta^{2i+3}}{(2i+3)!}\right) \\ &= \frac{\sin(\theta)}{\theta^3} - \frac{1}{\theta^3}\theta\sum_{i=0}^{\infty}\frac{(-1)^i\,\theta^{2i}}{2i!} \\ &= \frac{\sin(\theta)}{\theta^3} - \frac{1}{\theta^2}\cos(\theta).\end{aligned}$$

Next we are going to do a similar process to

$$\sum_{i=0}^{\infty}\frac{(2i+3)\,(-1)^i\,\theta^{2i}}{(2i+4)!}$$

in order to get a closed form solution. We are going to first multiply it by $\frac{\theta^4}{\theta^4}$ and then adding and subtracting $\frac{\cos(\theta)}{\theta^4}$ to it

$$\begin{aligned}\frac{\theta^4}{\theta^4}\sum_{i=0}^{\infty}\frac{(2i+3)\,(-1)^i\,\theta^{2i}}{(2i+4)!} &= \frac{1}{\theta^4}\sum_{i=0}^{\infty}\frac{(2i+3)\,(-1)^i\,\theta^{2i+4}}{(2i+4)!} \\ &= \frac{-\cos(\theta)}{\theta^4} + \frac{\cos(\theta)}{\theta^4} + \frac{1}{\theta^4}\sum_{i=0}^{\infty}\frac{(2i+3)\,(-1)^i}{(2i+4)} \\ &= \frac{-\cos(\theta)}{\theta^4} + \frac{1}{\theta^4}\left(\sum_{i=0}^{\infty}\frac{(-1)^i\,\theta^{2i}}{2i!} + \sum_{i=0}^{\infty}\frac{(2i+}\right. \\ &= \frac{-\cos(\theta)}{\theta^4} + \frac{1}{\theta^4}\left(1 + \frac{\theta^2}{2} - \theta\sin(\theta)\right) \\ &= \frac{1-\cos(\theta)}{\theta^4} - \frac{\sin(\theta)}{\theta^3} + \frac{1}{2\theta^2}.\end{aligned}$$

Combining everything together we get

$$\frac{\partial V}{\partial v} = \frac{I}{2} + \left(\frac{\sin(\theta)}{\theta^3} - \frac{1}{\theta^2}\cos(\theta)\right)v + \left(\frac{1-\cos(\theta)}{\theta^4} - \frac{\sin(\theta)}{\theta^3} + \frac{1}{2\theta^2}\right)v^2$$

We can now approximate the observed position

$$y_k \approx P_{k-1} + R_{k-1}V\left(\delta_k\left(\omega_k\right)\right)\delta_k\left(\rho_k+\eta_k\right) + \frac{\partial V}{\partial v}\left(\delta\omega_k\right)\delta_k\nu_k\delta_k\left(\rho_k+\eta_k\right) +$$

We will make one last simplification to get the final output by assuming that $\delta_k^2 v_k \eta_k$ is negligible relative to everything else.

$$\begin{aligned}y_k &\approx P_{k-1} + R_{k-1}V\left(\delta_k\omega_k\right)\delta_k\left(\rho_k+\eta_k\right) + \frac{\partial V}{\partial v}\left(\delta_k\omega_k\right)\delta_k^2\nu_k\rho_k + v_k \\ &= P_{k-1} + R_{k-1}V\left(\delta_k\omega_k\right)\delta_k\left(\rho_k+\eta_k\right) - \frac{\partial V}{\partial v}\left(\delta_k\omega_k\right)\delta_k^2\rho_k^{\wedge}\nu_k^{\vee} + v_k\end{aligned}$$

Calculating the mean and covariance under the assumption that $E\left[\nu_k\eta_k^{\top}\right] = 0$ yields

$$E\left[y_k \mid x_{k-1}\right] = P_{k-1} + R_{k-1}V\left(\delta_k\left(\omega_k\right)\right)\delta_k\rho_k$$

$$\operatorname{cov}\left(y_k, y_k \mid x_{k-1}\right) = \delta_k^2 R_{k-1}^{\top}V\left(\delta_k\omega_k\right)E\left[\eta_k\eta_k^{\top}\right]V\left(\delta_k\omega_k\right)^{\top}R_{k-1}^{\top} + \delta_k^4\frac{\partial}{\partial}$$

In order to calculate the LMLE, we need $p\left(y_{k-n} \mid x_k\right)$. This easily follows from the results above.

$$E\left[y_{k-n} \mid x_k\right] = P_{k-n} - \Delta_{k-n}R_kV\left(-\Delta_{k-n}\omega\right)\rho$$

$$\operatorname{cov}\left(y_{k-n}, y_{k-n} \mid x_k\right) = \bar{Q} + \Delta_{k-n}^2 R_kV\left(-\Delta_{k-n}\omega_k\right)E\left[\eta_k\eta_k^{\top}\right]V\left(-\Delta_{k-n}\omega_k\right.$$

Once again we can represent R_k in terms on an element in the Lie algebra so that $R_k = \exp(\tau_k)$. Thus the LMLE is

$$\arg\min_{P_k,\tau_k,\omega_k,\rho_k}\sum_{n=1}^{k}\left(y_{k-n}-\hat{y}_{k-n}\right)^{\top}\left(\bar{Q}+R\right)^{-1}\left(y_{k-n}-\hat{y}_{k-n}\right).$$

The question remains under what conditions is the system observable.

A. Observability

In order to calculate the observability of the system, we are going to model the system in an alternate, continuous-time, equivalent form. Let the state of the system at time t be

$$x = \begin{bmatrix} \tau \\ P \\ \omega \\ \rho_x \end{bmatrix},$$

where $R = \exp(\tau^\wedge)$ is the attitude, P is the position, ω is the angular velocity in the body frame, and ρ_x is the translational velocity along the x-axis. We assume that $\rho_y = \rho_z = 0$. The kinematics for $SE(3)$ is

$$\begin{bmatrix} \dot{R} & \dot{P} \\ 0 & 0 \end{bmatrix} = \begin{bmatrix} \exp(\tau^\wedge) & P \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \omega^\wedge & \rho \\ 0 & 0 \end{bmatrix},$$

whose solution is

$$\begin{bmatrix} \exp(\tau^\wedge)\exp(\delta\omega^\wedge) & P + \delta RV(\delta\omega^\wedge)\rho \\ 0 & 1 \end{bmatrix}.$$

An equivariant flow on the Lie algebra for the rotation is

$$\log(\exp(\tau^\wedge)\exp(\delta\omega^\wedge)),$$

taking the derivative with respect to time yields

$$\left(J_r^{-1}(\tau^\wedge)\omega\right)^\wedge.$$

Thus, the state derivative is

$$\dot{x} = \begin{bmatrix} J_r^{-1}(\tau^\wedge)\omega \\ \exp(\tau^\wedge)\rho \\ 0 \\ 0 \end{bmatrix}.$$

We are assuming that we can only observe the position of the system, thus

$$y = Cx,$$

where

$$C = \begin{bmatrix} 0_{3\times3} & I_{3\times3} & 0_{3\times3} & 0_{3\times1} \end{bmatrix}.$$

The observability of a nonlinear system can only be determined locally using the Lie derivatives. Let

$$\dot{x} = f(x),$$
$$y = h(x),$$

then the observability matrix is calculated as follows:

$$A = \begin{bmatrix} L_f^0 h \\ L_f^1 h \\ L_f^2 h \\ L_f^3 h \\ \vdots \\ L_f^n h \end{bmatrix},$$

and

$$\mathcal{O} = \frac{\partial A}{\partial x}.$$

If the matrix $\mathcal{O}$ is full rank after, then the system is locally observable. We are going to calculate the first four terms of the matrix A

$$A = \begin{bmatrix} P \\ \exp(\tau^\wedge)\rho \\ G(\tau,\rho)J_r^{-1}(\alpha)\omega \\ L_f^3 h \end{bmatrix}.$$

The calculations become very tedious by hand, so we used MATLAB to calculate $L_f^3 h$ and the matrix $\mathcal{O}$ for us. We found that the system is observable under the conditions that $\rho_x \neq 0$, and ω_y or ω_z are non zero.

B. Seeding

When seeding the algorithm , we are going to ignore the probability aspect of the system. We also assume that the measurement y_k is available, that the body velocity is $\rho = [x, 0, 0]^\top$ with x being positive, and that we have at least three distinct measurements. Under these assumptions, the system will either move in a straight line if the angular velocity is 0 or orbit if it is non zero.

$$z_1 = (y_k - y_{k-1})/\Delta_{k-1} = R_k V(\Delta_{k-1}\omega)\rho$$
$$z_2 = (y_k - y_{k-2})/\Delta_{k-2} = R_k V(\Delta_{k-2}\omega)\rho$$
$$z_3 = (y_k - y_{k-3})/\Delta_{k-3} = R_k V(\Delta_{k-3}\omega)\rho$$

XIV. Derivative

Let

$$v = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}, \quad p = \begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix}, \quad \theta = \sqrt{v^\top v}.$$

Consider the Rodriquez formula for the exponential matrix

$$\exp(v^\wedge) = I + \frac{\sin(\theta)}{\theta}v^\wedge + \frac{1-\cos(\theta)}{\theta^2}(v^\wedge)^2,$$

then

$$\frac{d\exp(v^\wedge)p}{dv} = \alpha(\theta)(-p)^\wedge + \beta(\theta)v^\wedge pv^\top + \gamma(\theta)G + \delta(\theta)(v^\wedge)^2 pv^\top,$$

with

$$\alpha(\theta) = \frac{\sin(\theta)}{\theta},$$
$$\beta(\theta) = \frac{\cos(\theta)}{\theta^2} - \frac{\sin(\theta)}{\theta^3},$$
$$\gamma(\theta) = \frac{1-\cos(\theta)}{\theta^2},$$
$$\delta(\theta) = \frac{\sin(\theta)}{\theta^3} - 2\frac{1-\cos(\theta)}{\theta^4}$$
$$G = \begin{bmatrix} v_2p_2 + v_3p_3 & -2v_2p_1 + v_1p_2 & -2v_3p_1 + v_1p_3 \\ v_2p_1 - 2v_1p_2 & v_1p_1 + v_3p_3 & -2v_3p_2 + v_2p_3 \\ v_3p_1 - 2v_1p_3 & v_3p_2 - 2v_2p_3 & v_1p_1 + v_2p_2 \end{bmatrix}$$

XV. SE(2) Observability

Assume the state

$$x = \begin{bmatrix} \alpha \\ P \\ \omega \\ \rho \end{bmatrix},$$

where $R = \exp(\alpha^\wedge)$, P is the position, ω is the angular velocity and ρ is the body x velocity. The derivative is

$$\dot{x} = \begin{bmatrix} \omega \\ \cos(\alpha)\rho \\ \sin(\alpha)\rho \\ 0 \\ 0 \end{bmatrix}.$$

Then the observability matrix is

$$O = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ -\sin(\alpha)\rho & 0 & 0 & 0 & \cos(\alpha) \\ \cos(\alpha)\rho & 0 & 0 & 0 & \sin(\alpha) \\ -\cos(\alpha)\rho\omega & 0 & 0 & -\sin(\alpha)\rho & -\sin(\alpha)\omega \\ \sin(\alpha)\rho\omega & 0 & 0 & -\cos(\alpha)\rho & -\cos(\alpha)\omega \end{bmatrix},$$

which is full rank provided that $\rho, \omega \neq 0$. Thus the system is locally observable provided $\rho, \omega \neq 0$.

XVI. Formulas

$$x_{k+1} = f(x_k, u) + w_k$$
$$y_{k+1} = h(x_{k+1}) + v_k$$

$$\arg\max_{x_0, u} p(x_0) \prod_{j=0}^{m} p(y_j \mid x_j)$$

Appendix

A. Probability

B. Euclidean Group $\mathbb{R}^n$

C. Special Orthogonal Group $SO(n)$

Suppose that we can measure the pose of the system. That is

$$y_k = x_k \oplus v_k.$$

where y_k is the output, x_k is the state and v_k is measurement noise. The probability of the output condition on the state is denoted $p(y \mid x)$ and is defined as

$$p(y \mid x) = \eta \exp\left(-\frac{1}{2}(y \ominus x)^\top R^{-1}(y \ominus x)\right).$$

Recall that $x = \hat{x}\exp(\delta x)$. We do not know x exactly but we have our estimate $\hat{x}$. Thus we want to estimate $(y \ominus x)$ using the Taylor series expansion around $\hat{x}$.

We can approximate the probability by computing the Taylor series of $(y \ominus x)$.

$$\begin{aligned} \frac{\partial y \ominus x}{\partial x} &= \frac{\partial \log\left(x^{-1}y\right)}{\partial x} \\ &= J_r^{-1}\left(\log\left(x^{-1}y\right)\right)\left(\mathbf{Ad}_y^{-1}\right)(-\mathbf{Ad}_x) \\ &= -J_r^{-1}\left(\log\left(x^{-1}y\right)\right)\mathbf{Ad}y^{-1}x \\ &= -J_r^{-1}\left(\log\left(x^{-1}y\right)\mathbf{Ad}_{(xy)^{-1}}\right) \\ &= -J_l^{-1}\left(\log\left(x^{-1}y\right)\right). \end{aligned}$$

Using a first order Taylor series expansion, we can approximate $p(y \mid x)$ as

$$p(y \mid x) \approx \eta \exp\left(-\frac{1}{2}\left(y \ominus x - J_l^{-1}(y \ominus x)\delta x\right)^\top R^{-1}\left(y \ominus x - J_l^{-1}(y\right.\right.$$

Our objective is to calculate the posterior $p(x \mid y)$. Using Bayes rule we get

$$p(x \mid y) = \frac{p(y \mid x)p(x)}{p(y)}.$$

Since we are trying to maximize the posterior probability we can write $p(x \mid y)$ as

$$p(x \mid y) = \eta p(y \mid x)p(x).$$

Note that

$$\begin{aligned} p(y \mid x)p(x) &\approx \eta \exp\left(-\frac{1}{2}\left(y \ominus x - J_l^{-1}(y \ominus x)\delta x\right)^\top R^{-1}\left(y \ominus x - J_l^{-1}\right.\right. \\ &= \eta \exp\left(-\frac{1}{2}\left(y \ominus x - J_l^{-1}(y \ominus x)\delta x\right)^\top R^{-1}\left(y \ominus x - J_l^{-1}\right.\right. \end{aligned}$$

We are interested in maximizing the posterior distribution. To do this we

$$\frac{\partial L}{\partial \delta x} = e^\top R^{-1} J_l^{-1}(e) + \delta x^\top \left(J_l^{-1}(e)\right)^\top R^{-1} J_l^{-1}(e) + \delta x^\top P^{-1}.$$

where $e = y \ominus x$. Setting it to zero and solving for δx we get

$$\delta x = \left(\left(J_l^{-1}(e) \right)^\top R^{-1} J_l^{-1}(e) + P^{-1} \right) \left(J_l^{-1}(e) \right)^\top R^{-1} e.$$

Which can be written as

$$\delta x = Ke$$

where

$$S = \left(J_l^{-1}(e) \right) P \left(J_l^{-1}(e) \right)^\top + R$$
$$K = P \left(J_l^{-1}(e) \right)^\top S^{-1}.$$

D. Special Euclidean Group $SE(n)$

We begin with the definition of a Lie Group taken from [13].

Definition 1. A **Lie group** is a smooth manifold G that is also a group in the algebraic sens, with the property that the multiplication map $m : G \times G \to G$ and inversion map $i : G \to G$, given by

$$m(g, h) = g \bullet h, \quad i(g) = g^{-1}$$

are both smooth where $\bullet$ denotes the group operation.

Some examples of Lie groups that we focus on are the Euclidean space $\mathbb{R}^n$ with the group operation being addition and the inverse being the additive inverse, the special orthogonal group of dimension n, $SO(n)$ with the group operation being matrix multiplication and the inverse being the matrix inverse. , and the special Euclidean group of dimension n,$SE(n)$ with the group operation being matrix multiplication and the inverse being the matrix inverse.

Every Lie group has a corresponding Lie algebra. Depending on the author, the Lie algebra is defined differently but equivalently. We define it according to [1] and [13].

Definition 2. The **Lie algebra** $\mathfrak{g}$ of a Lie group G is the set of all left invariant vector fields endowed with the Lie bracket. The Lie algebra is isomorphic to the tangent space of the Lie group at identity.

The Lie algebra is also isomorphic to the canonical coordinates of the 1st kind [6] or exponential coordinates of the first kind [5]. We will overload the term Lie algebra to mean the set of all left invariant vector fields, the tangent space of the Lie group at identity or the canonical coordinates of the 1st kind and we rely on the context to make clear which isomorphic space of the Lie algebra we are referring to.

Some examples of Lie algebras that we focus on is the Lie algebra of $\mathbb{R}^n$ which is $\mathbb{R}^n$ equipped with the Lie bracket $[\cdot, \cdot] : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^n$ defined by $(a, b) \mapsto [a]_\times b$; the Lie algebra of $SO(n)$ which is the set of skew symmetric matrices of dimension n denoted as $\mathfrak{so}(n)$ equipped with the Lie bracket $[\cdot, \cdot] : \mathfrak{so}(n) \times \mathfrak{so}(n) \to \mathfrak{so}(n)$ defined by $(a, b) \mapsto ab - ba$; the Lie algebra of $SE(n)$ which is $\mathfrak{se}(n) = \left\{ \begin{bmatrix} \omega & \rho \\ 0_{n\times 1} & 0_{1\times 1} \end{bmatrix} : \omega \in \mathfrak{so}(\mathfrak{n}) \text{ and } \rho \in \mathbb{R}^n \right\}$.

The exponential map is the function $\exp : \mathfrak{g} \to G$. In the case of $\mathbb{R}^n$, the

References

[1] Ralph Abraham, Jerrold Mardsen, and Tudor Ratiu. *Manifolds, Tensor Analysis, and Applications*. Springer-Verlag, New York, first edition, 1998.

[2] Yaakov Bar-Shalom and Fortmaan Thomas E. *Tracking and Data Association*. Academin Press, Inc., 1988.

[3] Timothy Barfoot. *State Estimation For Robotics*. Cambridge University Press, 2019.

[4] Guillaume Bourmaud, Rémi Mégret, Marc Arnaudon, and Audrey Giremus. Continuous-Discrete Extended Kalman Filter on Matrix Lie Groups Using Concentrated Gaussian Distributions. *Journal of Mathematical Imaging and Vision*, 51(1):209–228, 2014.

[5] Francesco Bullo and Andrew D. Lewis. *Geometric Control of Mechanical Systems*. Springer Science+Business Media, LCC, New York, 2005.

[6] Kenth Engø. On the construction of geometric integrators in the RKMK class. *BIT Numerical Mathematics*, 40(1):41–61, 2000.

[7] Kenth Engø. Partitioned Runge-Kutta Methods in Lie-Group Setting. *BIT*, 43:21–39, 2003.

[8] Bill Goodwine. *Geometric Control of Mechanical Systems*, volume 50. Springer Science+Business Media, LCC, New York, 2005.

[9] Brian C. Hall. *Lie Groups, Lie Algebras, and Representations: An Elementary Introduction*. Springer-Verlag New York, Inc, 2003.

[10] Christoph Hertzberg, René Wagner, Udo Frese, and Lutz Schröder. Integrating generic sensor fusion algorithms with sound state representations through encapsulation of manifolds. *Information Fusion*, 2013.

[11] Arieh Iserles, Hans Z Munthe-kaas, and Antonella Zanna. Lie-group methods. pages 215–365, 2000.

[12] John M. Lee. *An Introduction to Smooth Manifolds*. Springer Science+Business Media, LCC, 2013.

[13] Taeyoung Lee, Melvin Leok, and N Harris Mcclamroch. Geometric Tracking Control of a Quadrotor UAV on SE(3). Technical report.

[14] Jeffrey Millard. Multiple Target Tracking in Realistic Environments Using Recursive-RANSAC in a Data Fusion Framework. page 82, 2017.

[15] Todd K Moon and Wynn C. Stirling. *Mathematical Methods and Algorithms for Signal Processing*. Prentice-Hall, Uppder Saddle River, New Jersey, 2000.

[16] Hans Munthe-Kaas. Lie-Butcher theory for Runge-Kutta methods. *BIT Numerical Mathematics*, 35(4):572–587, 1995.

[17] Hans Munthe-Kaas. Runge-Kutta methods on Lie groups. *BIT Numerical Mathematics*, 38(1):92–111, 1998.

[18] Hans Munthe-Kaas. High order Runge-Kutta methods on manifolds. *Applied Numerical Mathematics*, 29(1):115–127, 1999.

[19] Peter C. Niedfeldt and Randal W. Beard. Multiple target tracking using recursive RANSAC. *Proceedings of the American Control Conference*, pages 3393–3398, 2014.

[20] Ian Reid. Estimation I. Technical report, 2001.

[21] Ian Reid. Estimation II 1 Discrete-time Kalman filter. Technical report, 2001.

[22] Joan Solà, Jeremie Deray, and Dinesh Atchuthan. A micro Lie theory for state estimation in robotics. pages 1–17, 2018.

[23] John Stillwell. *Naive Lie Theory*. Springer Science+Business Media, LCC, 2008.

[24] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic Robotics*. Massachusetts Institue of Technology, Massachusetts, 2006.

[25] Feng Yang, Weikang Tang, and Hua Lan. A density-based recursive RANSAC algorithm for unmanned aerial vehicle multi-target tracking in dense clutter. *IEEE International Conference on Control and Automation, ICCA*, (k 1):23–27, 2017.