An Introduction to Lie Groups for State Estimation

Johann Laconte

March 30, 2026

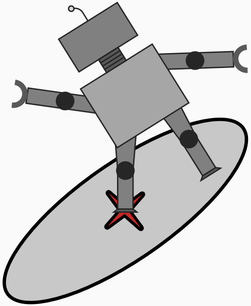
Context

- In robotics, **rotations** and **poses** are everywhere in the theory
- Localization, SLAM, Calibration, ...

Primer on State Estimation

Primer on State Estimation
The Rotation Issue
Lie Groups and Algebras
Probability and Statistics
Kinematics
Filtering on Lie Groups
Summary

State Estimation Principles



State estimation consists in inferring a “state” $\boldsymbol{x}$ from

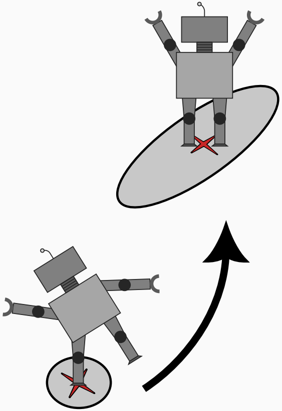
1. Some measurements $\boldsymbol{y}$
2. A prior on the system dynamics $\boldsymbol{v}$

We represent uncertainty with a probability distribution $p(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{v})$.

In the Gaussian case:

$$\boldsymbol{x} \sim \mathcal{N}(\hat{\boldsymbol{x}}, \mathbf{P})$$

Prior



We assume the robot follow a linear motion model:

$$\boldsymbol{x}_k = \underbrace{\mathbf{A}_{k-1}\boldsymbol{x}_{k-1}}_{\text{Dynamics}} + \underbrace{\boldsymbol{v}_k}_{\text{Input}} + \underbrace{\boldsymbol{w}_k}_{\text{Noise}}, \quad \boldsymbol{w}_k \sim \mathcal{N}(\mathbf{0}, \boldsymbol{Q}_k)$$

It can be put in a batch form

$$\boldsymbol{x} = \mathrm{A}(\boldsymbol{v} + \boldsymbol{w})$$

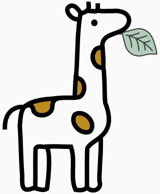
where $\boldsymbol{x}$ contains all the states $\boldsymbol{x}_k$ staked together, as

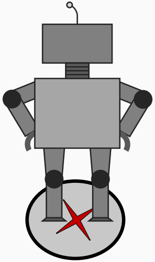
$$\boldsymbol{x} = \left[\boldsymbol{x}_0\ \boldsymbol{x}_1 \cdots \boldsymbol{x}_N\right]^T$$

The prior have the distrubution $p(\boldsymbol{x}|\boldsymbol{v}) = \mathcal{N}(\check{\mathbf{x}}, \check{\mathbf{P}})$ with

$$\check{\boldsymbol{x}} = \mathrm{A}\boldsymbol{v}$$
$$\check{\mathbf{P}} = \mathrm{AQA}^T$$

Measurements





We assume a linear measurement model:

$$\boldsymbol{y}_k = \mathbf{C}_k \boldsymbol{x}_k + \boldsymbol{n}_k \quad \boldsymbol{n}_k \sim \mathcal{N}(\mathbf{0}, \mathbf{R}_k)$$

It can the put in a batch form:

$$\boldsymbol{y} = \mathbf{C}\boldsymbol{x} + \boldsymbol{n}$$

where $\boldsymbol{y}$ contains all the measurements $\boldsymbol{y}_k$ staked together, as

$$\boldsymbol{y} = \begin{bmatrix} \boldsymbol{y}_0 \; \boldsymbol{y}_1 \cdots \boldsymbol{y}_N \end{bmatrix}^T$$

What about the full posterior $p(\boldsymbol{x}|\boldsymbol{y}, \boldsymbol{v})$?

Bayesian Inference

With a bit of maths, we get the full posterior as

$$p(\boldsymbol{x}|\boldsymbol{v},\boldsymbol{y}) = \mathcal{N}\left(\underbrace{\left(\check{\mathbf{P}}^{-1} + \mathbf{C}^T\mathbf{R}^{-1}\mathbf{C}\right)^{-1}\left(\check{\mathbf{P}}^{-1}\check{\boldsymbol{x}} + \mathbf{C}^T\mathbf{R}^{-1}\boldsymbol{y}\right)}_{\text{Mean } \hat{\boldsymbol{x}}}, \underbrace{\left(\check{\mathbf{P}}^{-1} + \mathbf{C}^T\mathbf{R}^{-1}\mathbf{C}\right)^{-1}}_{\text{Covariance } \hat{\boldsymbol{P}}}\right) \tag{1}$$

For the mean, a more compact form is

$$\mathbf{H}^T\mathbf{W}^{-1}\mathbf{H}\,\hat{\boldsymbol{x}} = \mathbf{H}^T\mathbf{W}^{-1}\boldsymbol{z}$$

with

$$\boldsymbol{z} = \begin{bmatrix}\boldsymbol{v}\\ \boldsymbol{y}\end{bmatrix}, \quad \mathbf{H} = \begin{bmatrix}\mathbf{A}^{-1}\\ \mathbf{C}\end{bmatrix}, \quad \mathbf{W} = \begin{bmatrix}\mathbf{Q} & \mathbf{0}\\ \mathbf{0} & \mathbf{R}\end{bmatrix}$$

RTS smoother

we can manipulate the matrices to get to a better form for solving efficiently

Forward pass (k=1...N)

$$
\begin{aligned}
\check{\mathbf{P}}_{k,f} &= \mathbf{A}_{k-1}\hat{\mathbf{P}}_{k-1,f}\mathbf{A}_{k-1}^T + \mathbf{Q}_k \\
\check{\boldsymbol{x}}_{k,f} &= \mathbf{A}_{k-1}\hat{\boldsymbol{x}}_{k-1,f} + \boldsymbol{v}_k \\
\mathbf{K}_k &= \check{\mathbf{P}}_{k,f}\mathbf{C}_k^T\left(\mathbf{C}_k\check{\mathbf{P}}_{k,f}\mathbf{C}_k^T + \mathbf{R}_k\right)^{-1} \\
\hat{\mathbf{P}}_{k,f} &= (\mathbf{I} - \mathbf{K}_k\mathbf{C}_k)\check{\mathbf{P}}_{k,f} \\
\hat{\boldsymbol{x}}_{k,f} &= \check{\boldsymbol{x}}_{k,f} + \mathbf{K}_k(\boldsymbol{y}_k - \mathbf{C}_k\check{\boldsymbol{x}}_{k,f})
\end{aligned}
$$

Kalman Filter

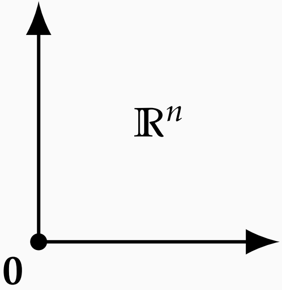
Backward pass (k=N...1)

$$
\begin{aligned}
\hat{\boldsymbol{x}}_{k-1} &= \hat{\boldsymbol{x}}_{k-1,f} + \left(\hat{\mathbf{P}}_{k-1,f}\mathbf{A}_{k-1}^T\check{\mathbf{P}}_{k,f}^{-1}\right)(\hat{\boldsymbol{x}}_k - \check{\boldsymbol{x}}_{k,f}) \\
\hat{\mathbf{P}}_{k-1} &= \hat{\mathbf{P}}_{k-1,f} + \left(\hat{\mathbf{P}}_{k-1,f}\mathbf{A}_{k-1}^T\check{\mathbf{P}}_{k,f}^{-1}\right)\left(\hat{\mathbf{P}}_k - \check{\mathbf{P}}_{k,f}\right) \\
&\quad \times \left(\hat{\mathbf{P}}_{k-1,f}\mathbf{A}_{k-1}^T\check{\mathbf{P}}_{k,f}^{-1}\right)^T
\end{aligned}
$$

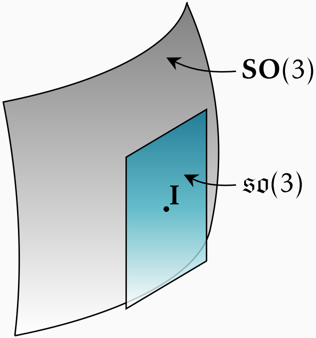
The Rotation Issue

Primer on State Estimation
The Rotation Issue
Lie Groups and Algebras
Probability and Statistics
Kinematics
Filtering on Lie Groups
Summary

Dealing with rotations and poses



- Find a parametrization $p \in \mathbb{R}^n$ of the rotation and the pose
- A parametrization either brings constraints or singularities

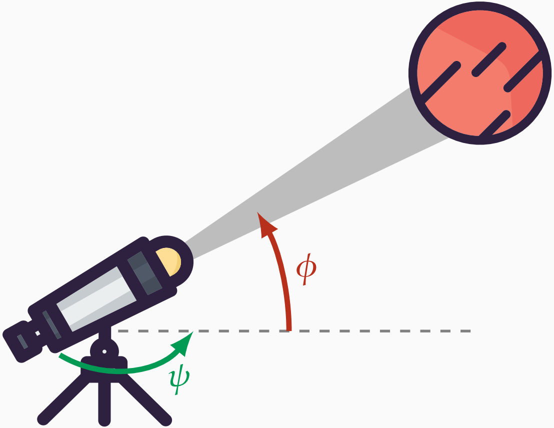


- Directly work with the full objects $\boldsymbol{C}$, $\boldsymbol{T}$
- Classier, more intuitive, but more complex...

A quick example in 2D: Gimbal Lock

Imagine a telescope tracking the trajectory of a planet. At the precise moment the planet passes directly overhead, the azimuth angle ψ must instantaneously jump by 180° to continue following the planet's motion.

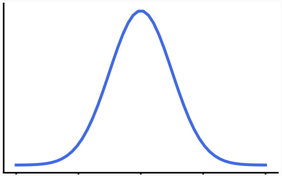
When $\phi = 180^\circ$, changes in the azimuth angle ψ no longer affect the state of the telescope. This results in a local loss of one degree of freedom.



The nice properties of the real numbers

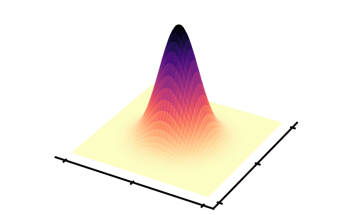
The real numbers $\mathbb{R}$

- Field with operations $(+, \times)$
- lots of nice properties: commutative, associative, ...



The N-dimensional space $\mathbb{R}^N$

- Vector space over $\mathbb{R}$
- lots of nice properties: commutative, associative, ...



Rotation matrices

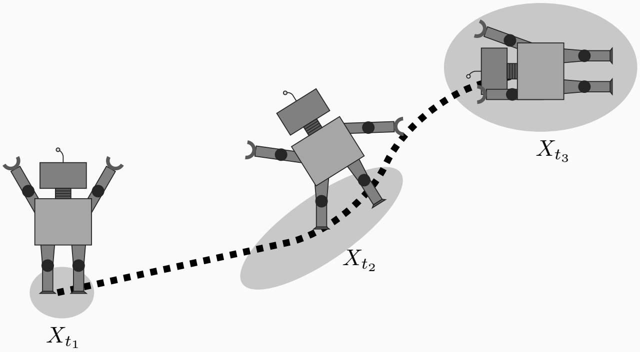
- Continuous Group
- **noncommutative**

?

Using probability distributions on SO(3)

- $\boldsymbol{C}_1 + \boldsymbol{C}_2$ is not a rotation matrix
- $\boldsymbol{C} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ does not make sense

We cannot use standard mathematical tools since **the rotations and poses do not form vector spaces.**



Lie Groups and Algebras

Primer on State Estimation
The Rotation Issue
Lie Groups and Algebras
Probability and Statistics
Kinematics
Filtering on Lie Groups
Summary

Group

A group is a non-empty set G together with a binary operation on G, denoted "$\cdot$", such that the following four requirements are satisfied:

1. **Closure:** For all a, b in G, we have
$$a \cdot b \in G$$
2. **Identity element:** There exists an unique element e in G (identity element) such that, for every a in G, one has
$$e \cdot a = a \quad \text{and} \quad a \cdot e = a.$$
3. **Associativity:** For all a, b, c in G, one has
$$(a \cdot b) \cdot c = a \cdot (b \cdot c).$$
4. **Inverse element:** For each a in G, there exists an element b in G such that
$$a \cdot b = e \quad \text{and} \quad b \cdot a = e,$$
where e is the identity element. b is commonly denoted a^{-1}.

The 3D rotation group $\mathbf{SO}(3)$

The Special Orthogonal group $\mathbf{SO}(3)$ represents the rotations in 3D:

$\mathbf{SO}(3)$ group

$$\mathbf{SO}(3) = \left\{ \boldsymbol{C} \in \mathbb{R}^{3\times 3} \middle| \boldsymbol{C}\boldsymbol{C}^T = \boldsymbol{I}, \det \boldsymbol{C} = 1 \right\}$$

with the standard matrix multiplication as the group operation.

Prove that $\mathbf{SO}(3)$ follows the group axioms and thus is a group.

The 3D pose group $\mathbf{SE}(3)$

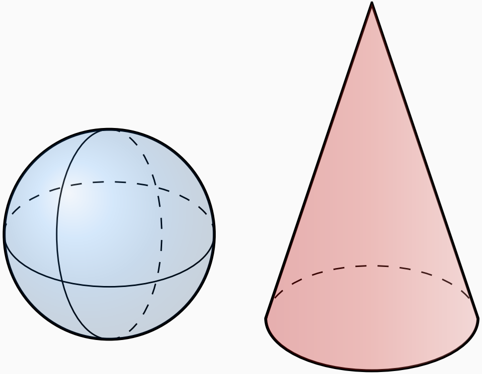
The Special Euclidean group $\mathbf{SE}(3)$ represent the spatial transformations in 3D:

$\mathbf{SE}(3)$ group

$$\mathbf{SE}(3) = \left\{ \begin{bmatrix} \boldsymbol{C} & \boldsymbol{r} \\ \mathbf{0}^T & 1 \end{bmatrix} \in \mathbb{R}^{4\times 4} \;\middle|\; \boldsymbol{C} \in \mathbf{SO}(3), \boldsymbol{r} \in \mathbb{R}^3 \right\}$$

with the standard matrix multiplication as the group operation.

Prove that $\mathbf{SE}(3)$ follows the group axioms and thus is a group.



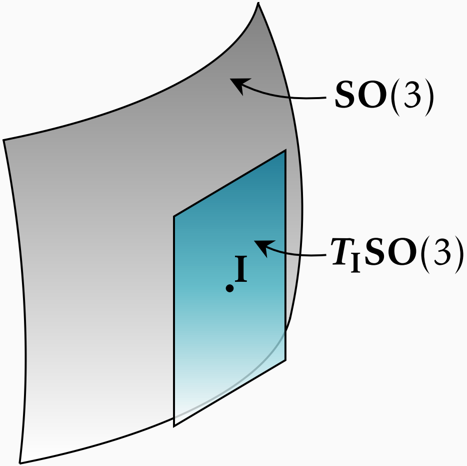
Lie Group

A Lie group is a group that is also a differentiable manifold.

Differentiable Manifold

A differentiable manifold is a type of manifold that is locally similar enough to a vector space to apply calculus.

Cartan's theorem if H is a closed subgroup of a Lie group G, then H is a Lie group



If $\mathbf{SO}(3)$ and $\mathbf{SE}(3)$ are Lie groups, we can locally approximate them with vector spaces...

Lie Algebra

A Lie Algebra is a vector space V with a binary operation $[\cdot, \cdot]$, called the **Lie bracket**, that satisfies:

1. **Closure:**

$$[X, Y] \in V, \text{ for all } X, Y \in V.$$

2. **Bilinearity:**

$$[aX + bY, Z] = a[X, Z] + b[Y, Z],$$

$$[Z, aX + bY] = a[Z, X] + b[Z, Y],$$

where $a, b \in \mathbb{F}$ (the underlying field), and $X, Y, Z \in V$.

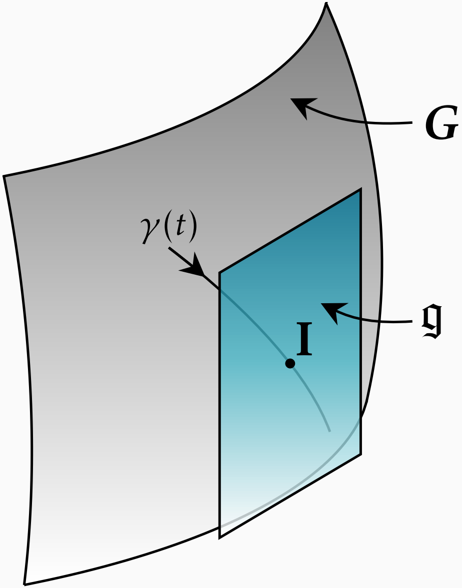
3. **Alternating:**

$$[X, X] = 0, \text{ for all } X \in V.$$

4. **Jacobi Identity:**

$$[X, [Y, Z]] + [Y, [Z, X]] + [Z, [X, Y]] = 0$$

for all $X, Y, Z \in V$.



A Lie algebra $\mathfrak{g}$ of a Lie group G is the tangent space of G at the identity element (and equipped with the Lie bracket $[\cdot, \cdot]$).
The tangent space at the identity is defined as

$$\mathfrak{g} = \{\gamma'(0) \,|\, \gamma(t) \in G, t \in \mathbb{R}, \gamma(0) = \boldsymbol{I}\}$$

where $\gamma(\cdot)$ is a **one-parameter subgroup**.

Rotations

Lie algebra $\mathfrak{so}(3)$ associated with SO(3)

vectorspace: $\mathfrak{so}(3) = \{\Phi = \boldsymbol{\phi}^\wedge \in \mathbb{R}^{3\times 3} \mid \boldsymbol{\phi} \in \mathbb{R}^3\}$,

Lie bracket: $[\Phi_1, \Phi_2] = \Phi_1\Phi_2 - \Phi_2\Phi_1$, with

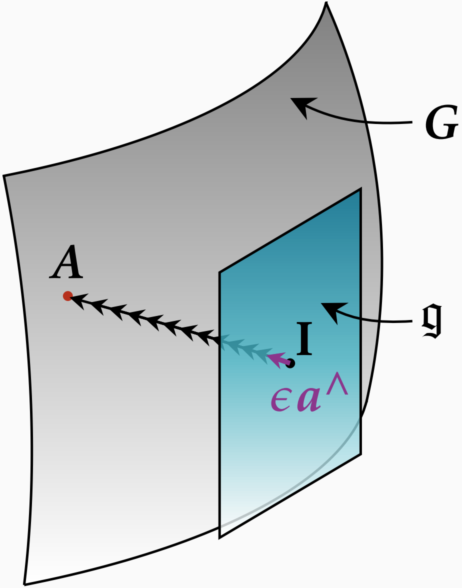
$$\boldsymbol{\phi}^\wedge = \begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \end{bmatrix}^\wedge = \begin{bmatrix} 0 & -\phi_3 & \phi_2 \\ \phi_3 & 0 & -\phi_1 \\ -\phi_2 & \phi_1 & 0 \end{bmatrix} \in \mathbb{R}^{3\times 3}, \boldsymbol{\phi} \in \mathbb{R}^3.$$

Lie algebra $\mathfrak{se}(3)$ associated with $\mathrm{SE}(3)$

vectorspace: $se(3) = \{\Xi = \boldsymbol{\xi}^\wedge \in \mathbb{R}^{4\times4} \mid \boldsymbol{\xi} \in \mathbb{R}^6\}$,

Lie bracket: $[\Xi_1, \Xi_2] = \Xi_1\Xi_2 - \Xi_2\Xi_1$, with

$$\boldsymbol{\xi}^\wedge = \begin{bmatrix} \boldsymbol{\rho} \\ \boldsymbol{\phi} \end{bmatrix}^\wedge = \begin{bmatrix} \boldsymbol{\phi}^\wedge & \boldsymbol{\rho} \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{4\times4}, \boldsymbol{\rho}, \boldsymbol{\phi} \in \mathbb{R}^3.$$



Close enough to the identity, we have the linear approximation

$$\boldsymbol{I} + \epsilon \boldsymbol{a}^{\wedge} \in \boldsymbol{G}, \quad \boldsymbol{a}^{\wedge} \in \mathfrak{g}$$

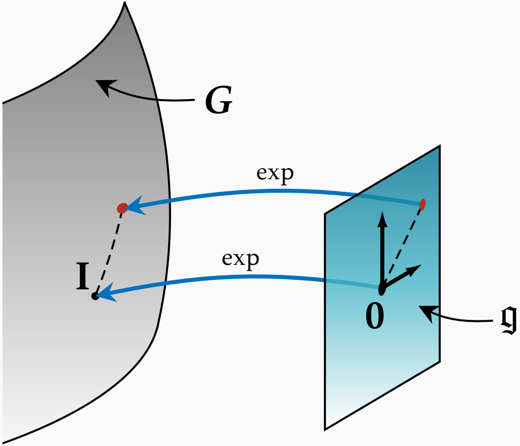
with ϵ being a very small number.

If we apply a large number of time this infinitesimal action, we can reach other elements of the group:

$$\begin{aligned} \boldsymbol{A} &= \lim_{n\to\infty} \left(\boldsymbol{I} + \frac{1}{n}\boldsymbol{a}^{\wedge} \right)^{n} \\ &= \exp\left(\boldsymbol{a}^{\wedge}\right) \end{aligned}$$

Exponential Map

The key property of a Lie algebra is that it **completely describes the structure** of its associated Lie Group. In the case of $\mathbf{SO}(3)$ and $\mathbf{SE}(3)$, we can find a local bijection between the Lie Group and the Lie Algebra, called the **exponential map**.



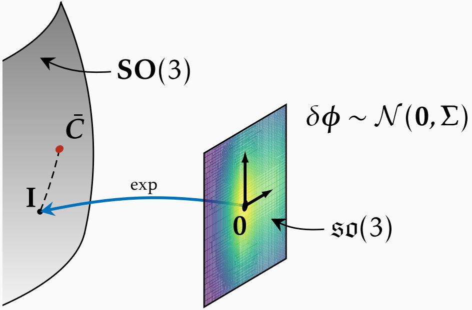
$$\exp(\mathbf{X}) = 1 + \mathbf{X} + \frac{1}{2!}\mathbf{X}^2 + \frac{1}{3!}\mathbf{X}^3 + \cdots = \sum_{n=0}^{\infty} \frac{\mathbf{X}^n}{n!},$$

$$\ln(\mathbf{A}) = \sum_{n=1}^{\infty} \frac{(-1)^{n-1}}{n}(\mathbf{A} - \mathbf{I})^n$$

Probability and Statistics

Primer on State Estimation
The Rotation Issue
Lie Groups and Algebras
Probability and Statistics
Kinematics
Filtering on Lie Groups
Summary

Probabilistic Rotation

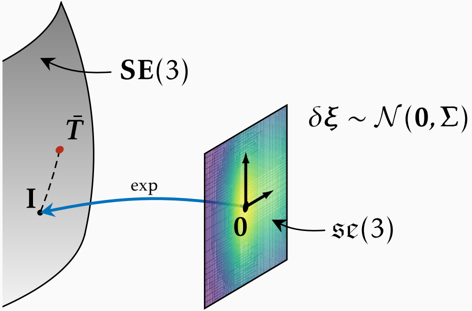


We want to model an uncertainty on a rotation $\boldsymbol{C} \in \mathbf{SO}(3)$:

$$\boldsymbol{C} = \exp\left(\delta\boldsymbol{\phi}^\wedge\right)\bar{\boldsymbol{C}}$$

with $\delta\boldsymbol{\phi} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$

Probabilistic Poses

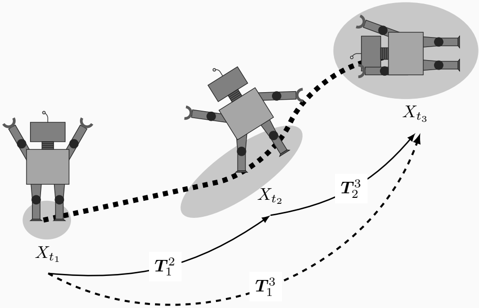


We want to model an uncertainty on a pose $\boldsymbol{T} \in \mathbf{SE}(3)$:

$$\boldsymbol{T} = \exp\left(\delta\boldsymbol{\xi}^{\wedge}\right) \bar{\boldsymbol{T}}$$

with $\delta\boldsymbol{\xi} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma})$

Compounding Poses



We want to model an uncertainty on a compounded pose $\boldsymbol{T} \in \mathbf{SE}(3)$: Given

$$\boldsymbol{T}_1^2 = \exp\left(\delta\boldsymbol{\xi}_1^\wedge\right)\bar{\boldsymbol{T}}_1^2, \quad \boldsymbol{\xi}_1 \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_1)$$
$$\boldsymbol{T}_2^3 = \exp\left(\delta\boldsymbol{\xi}_2^\wedge\right)\bar{\boldsymbol{T}}_2^3, \quad \boldsymbol{\xi}_2 \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_2)$$

find the mean and covariance of the compounded pose

$$\begin{aligned} \boldsymbol{T}_1^3 &= \boldsymbol{T}_1^2\boldsymbol{T}_2^3 \\ &= \exp\left(\delta\boldsymbol{\xi}_1^\wedge\right)\bar{\boldsymbol{T}}_1^2\exp\left(\delta\boldsymbol{\xi}_2^\wedge\right)\bar{\boldsymbol{T}}_2^3 \\ &\approx \exp\left(\Big[\underbrace{\delta\boldsymbol{\xi}_1 + \boldsymbol{\mathcal{T}}_1^2\delta\boldsymbol{\xi}_2}_{\sim\mathcal{N}(\mathbf{0},\Sigma_1+\boldsymbol{\mathcal{T}}_1^2\Sigma_2\boldsymbol{\mathcal{T}}_1^{2\,T})}\Big]^\wedge\right)\underbrace{\bar{\boldsymbol{T}}_1^2\bar{\boldsymbol{T}}_2^3}_{\text{Mean}} \end{aligned}$$

Kinematics

Primer on State Estimation
The Rotation Issue
Lie Groups and Algebras
Probability and Statistics
Kinematics
Filtering on Lie Groups
Summary

Suppose we have a perturbed rotation matrix $\boldsymbol{C}'$ such that

$$\boldsymbol{C}' = \exp\left(\delta\boldsymbol{\phi}^\wedge\right)\boldsymbol{C}$$

Assuming the perturbation is small, we simplify this into

$$\begin{aligned}
&\boldsymbol{C}' \approx (\boldsymbol{I} + \delta\boldsymbol{\phi}^\wedge)\boldsymbol{C} \\
\Rightarrow &\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{C}' \approx \frac{\mathrm{d}}{\mathrm{d}t}\left(\boldsymbol{I} + \delta\boldsymbol{\phi}^\wedge\right)\boldsymbol{C} \\
\Leftrightarrow &(\boldsymbol{\omega} + \delta\boldsymbol{\omega})^\wedge(\boldsymbol{I} + \delta\boldsymbol{\phi}^\wedge)\boldsymbol{C} = \frac{\mathrm{d}}{\mathrm{d}t}\left(\boldsymbol{I} + \delta\boldsymbol{\phi}^\wedge\right)\boldsymbol{C} \\
\Rightarrow &\delta\dot{\boldsymbol{\phi}} = \boldsymbol{\omega}^\wedge\delta\boldsymbol{\phi} + \delta\boldsymbol{\omega}
\end{aligned}$$

Linearized kinematics on SO(3)

$$\begin{cases} \text{Nominal:} & \dot{\boldsymbol{C}} = \boldsymbol{\omega}^\wedge\boldsymbol{C} \\ \text{Perturbation:} & \delta\dot{\boldsymbol{\phi}} = \boldsymbol{\omega}^\wedge\delta\boldsymbol{\phi} + \delta\boldsymbol{\omega} \end{cases}$$

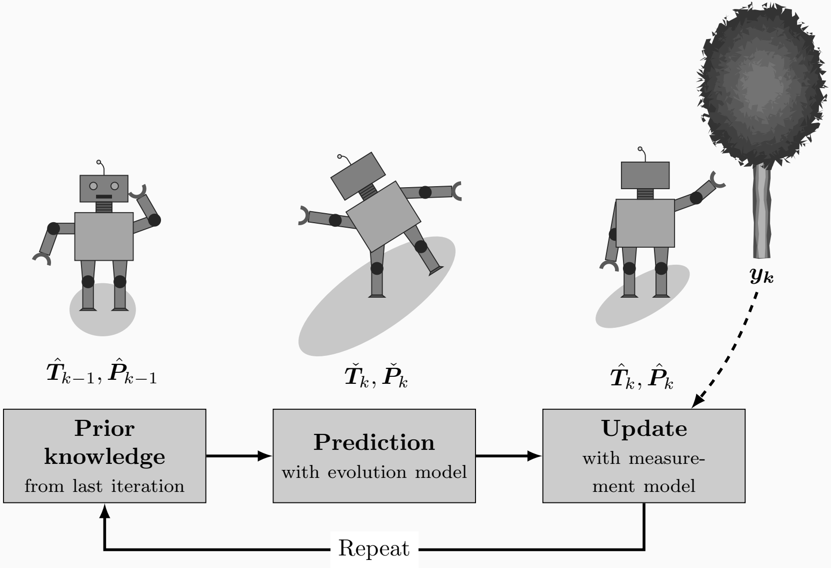
Linearized kinematics on SE(3)

$$\begin{cases} \text{Nominal:} & \dot{\boldsymbol{T}} = \boldsymbol{\varpi}^\wedge\boldsymbol{T} \\ \text{Perturbation:} & \delta\dot{\boldsymbol{\xi}} = \boldsymbol{\varpi}^\curlywedge\delta\boldsymbol{\xi} + \delta\boldsymbol{\varpi} \end{cases}$$

Filtering on Lie Groups

Primer on State Estimation
The Rotation Issue
Lie Groups and Algebras
Probability and Statistics
Kinematics
Filtering on Lie Groups
Summary

Kalman Filter on Lie Groups



Kalman Filter - Prediction Step

Let $\boldsymbol{C}' = \exp\left(\delta\boldsymbol{\phi}^\wedge\right)\boldsymbol{C}, \quad \delta\boldsymbol{\phi} \sim \mathcal{N}\left(\mathbf{0}, \boldsymbol{P}\right)$

1) Use the nominal equation to propagate the mean

$$\dot{\boldsymbol{C}} = \boldsymbol{\omega}^\wedge \boldsymbol{C}$$
$$\Rightarrow \check{\boldsymbol{C}}_k = \underbrace{\exp\left(\Delta t \cdot \boldsymbol{\omega}_k^\wedge\right)}_{\boldsymbol{F}_k} \boldsymbol{C}_{k-1}$$

2) Use the perturbation equation to propagate the co-variance

$$\delta\dot{\boldsymbol{\phi}} = \boldsymbol{\omega}^\wedge \delta\boldsymbol{\phi} + \delta\boldsymbol{\omega}$$
$$\Rightarrow \delta\boldsymbol{\phi}_k = \underbrace{\exp\left(\Delta t \cdot \boldsymbol{\omega}_k^\wedge\right)}_{\boldsymbol{F}_k} \delta\boldsymbol{\phi}_{k-1} + \boldsymbol{w}_k$$

Thus

$$\check{\boldsymbol{P}}_k = \boldsymbol{F}_k \boldsymbol{P}_{k-1} \boldsymbol{F}_k^T + \boldsymbol{Q}_k$$

KF Filter - Prediction step

$$\begin{cases} \check{\boldsymbol{C}}_k &= \boldsymbol{F}_k \boldsymbol{C}_{k-1} \\ \check{\boldsymbol{P}}_k &= \boldsymbol{F}_k \boldsymbol{P}_{k-1} \boldsymbol{F}_k^T + \boldsymbol{Q}_k \end{cases}$$

Kalman Filter - Update Step

We write the measurement model as

$$\boldsymbol{y}_k = \boldsymbol{G}_k \delta\boldsymbol{\phi}_k + \boldsymbol{n}_k + \text{cst}, \quad \boldsymbol{n}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{R}_k)$$

The Kalman gain and covariance update equations are then unchanged from the generic case:

$$\begin{aligned} \boldsymbol{K}_k &= \check{\boldsymbol{P}}_k \boldsymbol{G}_k^T \left(\boldsymbol{G}_k \check{\boldsymbol{P}}_k \boldsymbol{G}_k^T + \boldsymbol{R}_k \right)^{-1} \\ \boldsymbol{\mu}_{\delta\boldsymbol{\phi}_k} &= \boldsymbol{K}_k \left(\boldsymbol{y}_k - \check{\boldsymbol{y}}_k \right) \\ \boldsymbol{P}_{\delta\boldsymbol{\phi}_k} &= \left(\boldsymbol{I} - \boldsymbol{K}_k \boldsymbol{G}_k \right) \check{\boldsymbol{P}}_k \end{aligned}$$

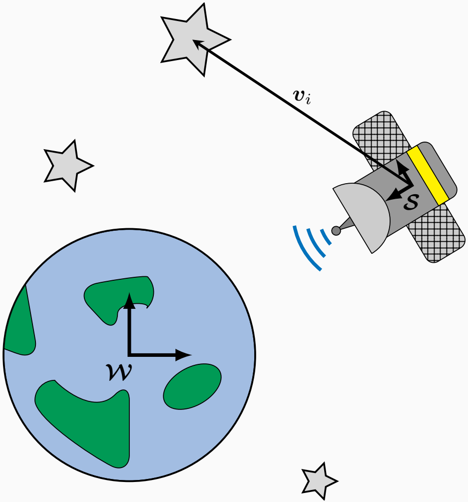
From this, we have the updated estimate

$$\begin{aligned} \hat{\boldsymbol{C}}_k &= \exp\left(\delta\hat{\boldsymbol{\phi}}_k + \boldsymbol{\mu}_{\delta\boldsymbol{\phi}_k} \right) \boldsymbol{C}_k, \quad \delta\hat{\boldsymbol{\phi}}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{P}_{\delta\boldsymbol{\phi}_k}) \\ &\approx \underbrace{\exp\left(\boldsymbol{J}(\boldsymbol{\mu}_{\delta\boldsymbol{\phi}_k}) \delta\hat{\boldsymbol{\phi}}_k \right)}_{\text{uncertainty}} \underbrace{\exp\left(\boldsymbol{\mu}_{\delta\boldsymbol{\phi}_k} \right) \boldsymbol{C}_k}_{\text{updated mean}} \end{aligned}$$

KF Filter - Update step

$$\begin{cases} \boldsymbol{K}_k &= \check{\boldsymbol{P}}_k \boldsymbol{G}_k^T \left(\boldsymbol{G}_k \check{\boldsymbol{P}}_k \boldsymbol{G}_k^T + \boldsymbol{R}_k \right)^{-1} \\ \boldsymbol{\mu}_{\delta\boldsymbol{\phi}_k} &= \boldsymbol{K}_k \left(\boldsymbol{y}_k - \check{\boldsymbol{y}}_k \right) \\ \boldsymbol{P}_{\delta\boldsymbol{\phi}_k} &= \left(\boldsymbol{I} - \boldsymbol{K}_k \boldsymbol{G}_k \right) \check{\boldsymbol{P}}_k \\ \hat{\boldsymbol{C}}_k &= \exp\left(\boldsymbol{\mu}_{\delta\boldsymbol{\phi}_k}^{\wedge} \right) \check{\boldsymbol{C}}_k \\ \hat{\boldsymbol{P}}_k &= \boldsymbol{J}(\boldsymbol{\mu}_{\delta\boldsymbol{\phi}_k}) \boldsymbol{P}_{\delta\boldsymbol{\phi}_k} \boldsymbol{J}(\boldsymbol{\mu}_{\delta\boldsymbol{\phi}_k})^T \end{cases}$$

Example: Attitude Estimation



A satellite estimates its orientation $\boldsymbol{C} \in \mathbf{SO}(3)$.

We estimate the rotation via

$$\boldsymbol{C} = \exp(\delta\boldsymbol{\phi})\hat{\boldsymbol{C}}$$

with $\delta\boldsymbol{\phi} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{P})$.

It measures known star directions $\boldsymbol{v}_k$:

$$\boldsymbol{y}_k = \boldsymbol{C}\boldsymbol{v}_k + \boldsymbol{n}_k$$

Kalman Equations

Prior

The continuous model is

$$\dot{\boldsymbol{C}} = \boldsymbol{\omega}^{\wedge}\boldsymbol{C}$$

Integrating the ODE, we get

$$\check{\boldsymbol{C}}_k = \exp(\boldsymbol{\omega}_k \Delta t)\hat{\boldsymbol{C}}_{k-1}$$

and thus

$$\boldsymbol{F}_k = \exp(\boldsymbol{\omega}_k \Delta t)$$

Update

From the measurement model

$$\boldsymbol{y}_k = \boldsymbol{C}_k \boldsymbol{v}_k + \boldsymbol{n}_k$$

We linearize

$$\begin{aligned} \boldsymbol{y}_k &= (\boldsymbol{I} + \delta\boldsymbol{\phi}_k^{\wedge})\check{\boldsymbol{C}}_k \boldsymbol{v}_k + \boldsymbol{n}_k \\ \boldsymbol{y}_k &= \check{\boldsymbol{C}}_k \boldsymbol{v}_k - (\check{\boldsymbol{C}}_k \boldsymbol{v}_k)^{\wedge}\delta\boldsymbol{\phi} + \boldsymbol{n}_k \end{aligned}$$

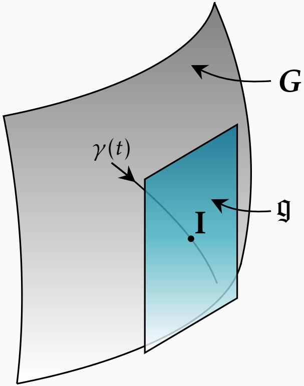
and thus

$$\boldsymbol{G}_k = -(\check{\boldsymbol{C}}_k \boldsymbol{v}_k)^{\wedge}$$

Summary

Primer on State Estimation
The Rotation Issue
Lie Groups and Algebras
Probability and Statistics
Kinematics
Filtering on Lie Groups
Summary

Summary



- Lie theory has numerous uses in robotics, such as control, state estimation, and modeling.
- Using this theory, we can define probability distributions on rotations and poses, as well as perform optimization.
- State Estimation is cleaner and “singularity-free”.

To dive further in the theory:

Barfoot, T. D. (2024). State estimation for robotics. Cambridge University Press.