深蓝学院
shenlanxueyuan.com

机器人学中的状态估计

Timothy Barfoot 著 高翔，谢晓佳等 译
Slides by Xiang Gao

2020年春

第7讲 位姿估计问题

- 点云对准
- 点云跟踪
- 位姿图松驰化

内容回顾

- 前面我们介绍了
  - 通常向量空间的状态估计理论
  - 三维空间的矩阵李群

- 本节课与下节课
  - 该领域内的一些典型问题比如点云配准/SLAM等

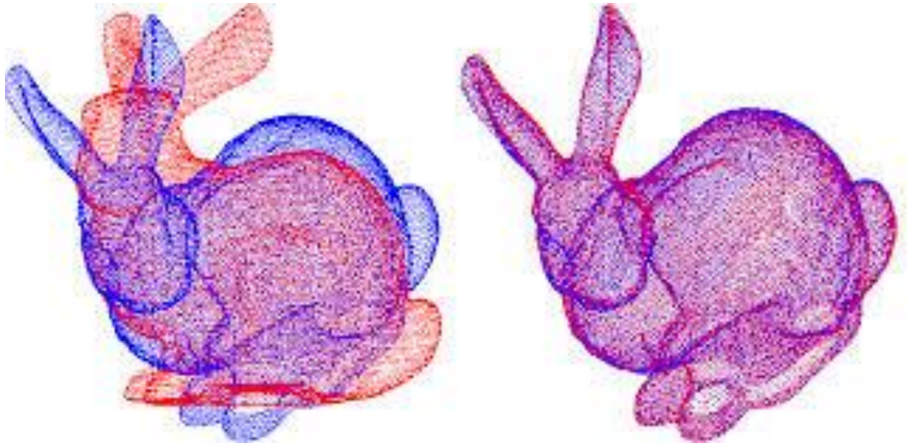
  - 本章重点将放在前两个问题上，尤其是第二个问题
  - 由于书本没有课后习题，我们将这里的一些推导作为习题

第7讲 位姿估计问题

- 点云对准
- 点云跟踪
- 位姿图松驰化

点云对准

- 点云对准：通过最小二乘来解决两个三维点云之间的对准问题
- 常见的场景：激光Odom，点云拼接等



常见的解法是ICP
不过旋转部分可以用各种不同方式来表达
- 四元数
- 旋转矩阵
- 变换矩阵

点云对准

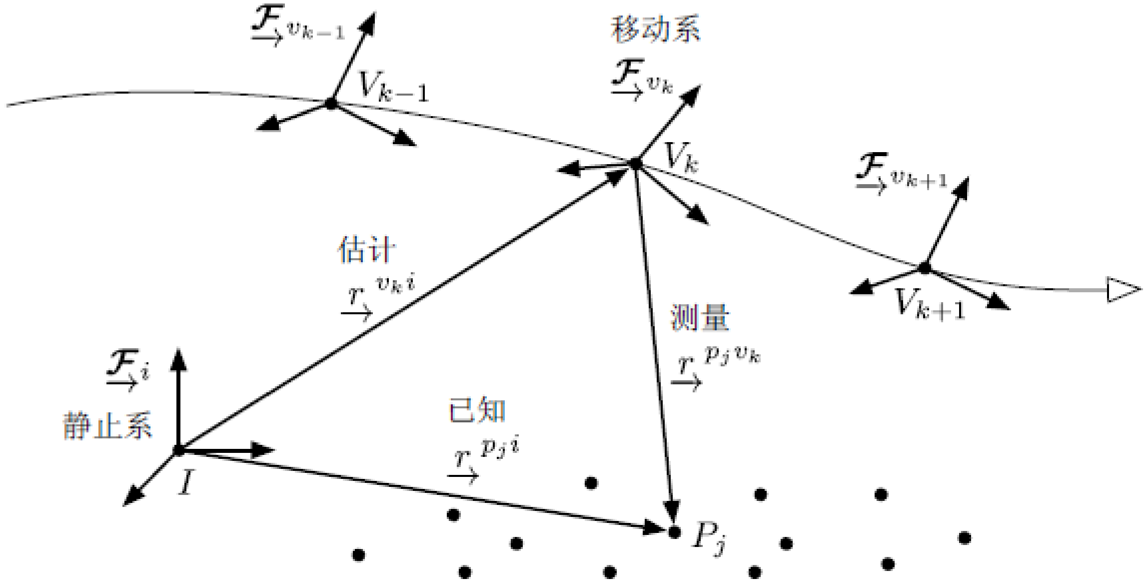
- 点云对准问题
- 背景：

惯性系： $\underrightarrow{\mathcal{F}}_i$
机器人系： $\underrightarrow{\mathcal{F}}_{v_k}$ 并随时间运动

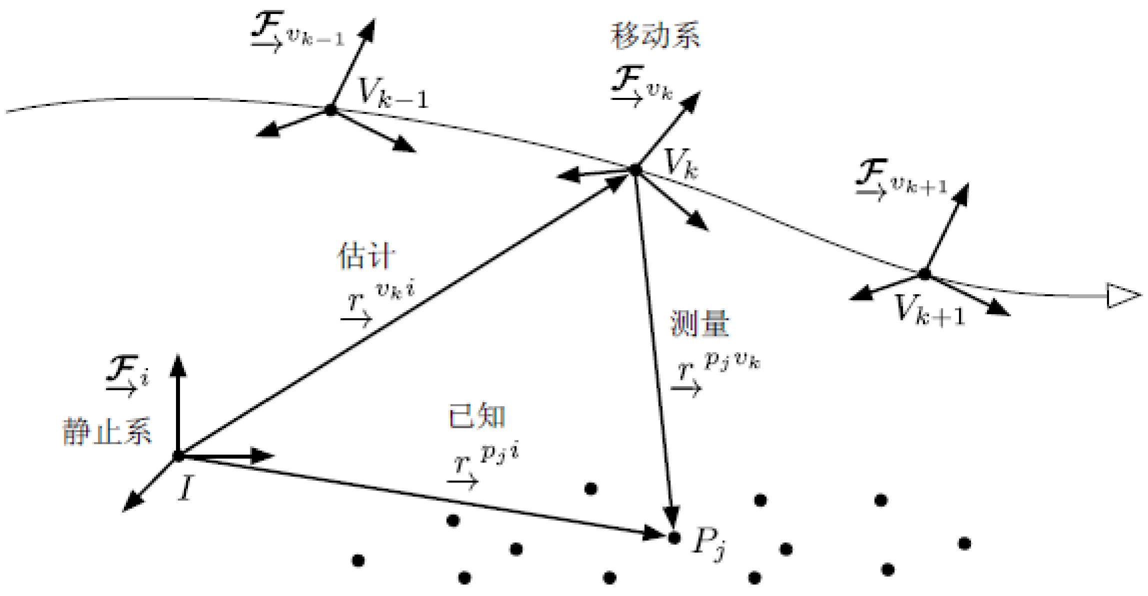
存在M个点：P_j ，j=1, ..., M

这些点在i系下坐标已知：$\mathbf{r}_i^{p_j i}$
机器人对M个三维点观测： $\mathbf{r}_{v_k}^{p_j v_k}$

希望推断机器人本体的旋转和平移



点云对准



- 单位四元数解法
- 定义齐次坐标

$$y_j = \begin{bmatrix} r_{v_k}^{p_j v_k} \\ 1 \end{bmatrix}, \quad p_j = \begin{bmatrix} r_i^{p_j i} \\ 1 \end{bmatrix}$$

- 那么它们满足：

$$\underbrace{\begin{bmatrix} r_{v_k}^{p_j v_k} \\ 1 \end{bmatrix}}_{y_j} = \underbrace{\begin{bmatrix} C_{v_k i} & 0 \\ 0^T & 1 \end{bmatrix}}_{q^{-1^+} q^{\oplus}} \left(\underbrace{\begin{bmatrix} r_i^{p_j i} \\ 1 \end{bmatrix}}_{p_j} - \underbrace{\begin{bmatrix} r_i^{v_k i} \\ 0 \end{bmatrix}}_{r} \right)$$

- 写成四元数坐标变换形式：

$$y_j = q^{-1^+} (p_j - r)^+ q$$

- 此问题中，四元数q和平移r可看成v系到i系的变换
- +算符把四元数乘法变成矩阵乘法

$$q = \begin{bmatrix} \varepsilon \\ \eta \end{bmatrix} \quad q^+ = \begin{bmatrix} \eta 1 - \varepsilon^\times & \varepsilon \\ -\varepsilon^T & \eta \end{bmatrix}, \quad q^\oplus = \begin{bmatrix} \eta 1 + \varepsilon^\times & \varepsilon \\ -\varepsilon^T & \eta \end{bmatrix}$$

点云对准

□ 由于q为单位四元数，这个处理不影响目标函数

- 误差项： $\boldsymbol{e}_j = \boldsymbol{y}_j - \boldsymbol{q}^{-1^+}(\boldsymbol{p}_j - \boldsymbol{r})^+\boldsymbol{q}$ 处理一下： $\boldsymbol{e}'_j = \boldsymbol{q}^+\boldsymbol{e}_j = \left(\boldsymbol{y}_j^\oplus - (\boldsymbol{p}_j - \boldsymbol{r})^+\right)\boldsymbol{q}$
- 那么目标函数为：

$$J(\boldsymbol{q}, \boldsymbol{r}, \lambda) = \frac{1}{2}\sum_{j=1}^{M} w_j \boldsymbol{e}_j'^{\mathrm{T}} \boldsymbol{e}'_j - \underbrace{\frac{1}{2}\lambda(\boldsymbol{q}^{\mathrm{T}}\boldsymbol{q} - 1)}_{\text{拉格朗日乘子项}}$$

□ w为权重

- 写开：

$$J(\boldsymbol{q}, \boldsymbol{r}, \lambda) = \frac{1}{2}\sum_{j=1}^{M} w_j \boldsymbol{q}^{\mathrm{T}}\left(\boldsymbol{y}_j^\oplus - (\boldsymbol{p}_j - \boldsymbol{r})^+\right)^{\mathrm{T}}\left(\boldsymbol{y}_j^\oplus - (\boldsymbol{p}_j - \boldsymbol{r})^+\right)\boldsymbol{q} - \frac{1}{2}\lambda(\boldsymbol{q}^{\mathrm{T}}\boldsymbol{q} - 1)$$

拉格朗日乘子
使得q为单位四元数

点云对准

- 求上式对各变量导数并令其为零：

$$J(\boldsymbol{q},\boldsymbol{r},\lambda)=\frac{1}{2}\sum_{j=1}^{M}w_j\boldsymbol{q}^{\mathrm{T}}\left(\boldsymbol{y}_j^{\oplus}-(\boldsymbol{p}_j-\boldsymbol{r})^{+}\right)^{\mathrm{T}}\left(\boldsymbol{y}_j^{\oplus}-(\boldsymbol{p}_j-\boldsymbol{r})^{+}\right)\boldsymbol{q}$$
$$-\frac{1}{2}\lambda(\boldsymbol{q}^{\mathrm{T}}\boldsymbol{q}-1)$$

$$\frac{\partial J}{\partial \boldsymbol{q}^{\mathrm{T}}}=\sum_{j=1}^{M}w_j\left(\boldsymbol{y}_j^{\oplus}-(\boldsymbol{p}_j-\boldsymbol{r})^{+}\right)^{\mathrm{T}}\left(\boldsymbol{y}_j^{\oplus}-(\boldsymbol{p}_j-\boldsymbol{r})^{+}\right)\boldsymbol{q}-\lambda\boldsymbol{q}$$

$$\frac{\partial J}{\partial \boldsymbol{r}^{\mathrm{T}}}=\boldsymbol{q}^{-1^{\oplus}}\sum_{j=1}^{M}w_j\left(\boldsymbol{y}_j^{\oplus}-(\boldsymbol{p}_j-\boldsymbol{r})^{+}\right)\boldsymbol{q}$$

$$\frac{\partial J}{\partial \lambda}=-\frac{1}{2}(\boldsymbol{q}^{\mathrm{T}}\boldsymbol{q}-1)$$

- 这个并不是非常直观，留作习题
- 提示：使用书本(6.19)-(6.21)帮助证明

- 第二式为零得到： $\boldsymbol{r}=\boldsymbol{p}-\boldsymbol{q}^{+}\boldsymbol{y}^{+}\boldsymbol{q}^{-1}$ 其中： $\boldsymbol{y}=\frac{1}{w}\sum_{j=1}^{M}w_j\boldsymbol{y}_j,\quad \boldsymbol{p}=\frac{1}{w}\sum_{j=1}^{M}w_j\boldsymbol{p}_j,\quad w=\sum_{j=1}^{M}w_j$

- 物理意义：y为观测质心，p为世界质心

点云对准

$$\frac{\partial J}{\partial \boldsymbol{q}^{\mathrm{T}}} = \sum_{j=1}^{M} w_j \left(\boldsymbol{y}_j^{\oplus} - (\boldsymbol{p}_j - \boldsymbol{r})^{+} \right)^{\mathrm{T}} \left(\boldsymbol{y}_j^{\oplus} - (\boldsymbol{p}_j - \boldsymbol{r})^{+} \right) \boldsymbol{q} - \lambda \boldsymbol{q}$$

$$\frac{\partial J}{\partial \boldsymbol{r}^{\mathrm{T}}} = \boldsymbol{q}^{-1^{\oplus}} \sum_{j=1}^{M} w_j \left(\boldsymbol{y}_j^{\oplus} - (\boldsymbol{p}_j - \boldsymbol{r})^{+} \right) \boldsymbol{q}$$

$$\frac{\partial J}{\partial \lambda} = -\frac{1}{2}(\boldsymbol{q}^{\mathrm{T}}\boldsymbol{q} - 1)$$

$$\boldsymbol{r} = \boldsymbol{p} - \boldsymbol{q}^{+}\boldsymbol{y}^{+}\boldsymbol{q}^{-1}$$

- r代入第1式并使之为零，得：

$$\boldsymbol{W}\boldsymbol{q} = \lambda \boldsymbol{q}$$

- 其中：

$$\boldsymbol{W} = \frac{1}{w} \sum_{j=1}^{M} w_j \left((\boldsymbol{y}_j - \boldsymbol{y})^{\oplus} - (\boldsymbol{p}_j - \boldsymbol{p})^{+} \right)^{\mathrm{T}} \left((\boldsymbol{y}_j - \boldsymbol{y})^{\oplus} - (\boldsymbol{p}_j - \boldsymbol{p})^{+} \right)$$

□ 这步可以用上一步结果

- 可见这是一个特征值问题，q是特征向量，lambda为特征值
- 如果W特征值为正且无重根，取q为最小特征值对应的单位特征向量，使得q长度为1
  - W是半正定的，它的特征值至少非负

点云对准

- 代入原目标函数：

$$J(\boldsymbol{q},\boldsymbol{r},\lambda)=\frac{1}{2}\sum_{j=1}^{M}w_j\boldsymbol{q}^{\mathrm{T}}\left(\boldsymbol{y}_j^{\oplus}-(\boldsymbol{p}_j-\boldsymbol{r})^{+}\right)^{\mathrm{T}}\left(\boldsymbol{y}_j^{\oplus}-(\boldsymbol{p}_j-\boldsymbol{r})^{+}\right)\boldsymbol{q} -\frac{1}{2}\lambda(\boldsymbol{q}^{\mathrm{T}}\boldsymbol{q}-1)$$

$$J(\boldsymbol{q},\boldsymbol{r},\lambda)=\frac{1}{2}\boldsymbol{q}^{\mathrm{T}}\underbrace{\boldsymbol{W}\boldsymbol{q}}_{\lambda\boldsymbol{q}}-\frac{1}{2}\lambda(\boldsymbol{q}^{\mathrm{T}}\boldsymbol{q}-1)=\frac{1}{2}\lambda$$

- 因此，选择最小特征值时使得原问题最小化
- 当W奇异时，问题退化，将在后面讨论
- 解得q后，转为矩阵表达即得旋转矩阵：$\begin{bmatrix}\hat{\mathbf{C}}_{v_k i} & \mathbf{0}\\ \mathbf{0}^{\mathrm{T}} & 1\end{bmatrix}=\boldsymbol{q}^{-1^{+}}\boldsymbol{q}^{\oplus},\quad \begin{bmatrix}\hat{\boldsymbol{r}}_i^{v_k i}\\ 0\end{bmatrix}=\boldsymbol{r}$

点云对准

- 旋转矩阵解法
- 类似前文，定义点云的观测和世界坐标：

$$\mathbf{y}_j = \mathbf{r}_{v_k}^{p_j v_k}, \quad \mathbf{p}_j = \mathbf{r}_i^{p_j i}, \quad \mathbf{r} = \mathbf{r}_i^{v_k i}, \quad \mathbf{C} = \mathbf{C}_{v_k i}$$

- 同时把质心、权重和也定义出来：

$$\mathbf{y} = \frac{1}{w}\sum_{j=1}^{M} w_j \mathbf{y}_j, \quad \mathbf{p} = \frac{1}{w}\sum_{j=1}^{M} w_j \mathbf{p}_j, \quad w = \sum_{j=1}^{M} w_j$$

- 这和上文大同小异，但符号使用3x1的非齐次坐标。

点云对准

- 每个点的误差： $e_j = y_j - C(p_j - r)$
- 整体误差：

$$J(C, r) = \frac{1}{2}\sum_{j=1}^{M} w_j e_j^{\mathrm{T}} e_j = \frac{1}{2}\sum_{j=1}^{M} w_j (y_j - C(p_j - r))^{\mathrm{T}} (y_j - C(p_j - r))$$

s.t. C为旋转矩阵

- 首先分离平移量： $d = r + C^{\mathrm{T}} y - p$
- 写成r的表达式并代入目标函数： □ 代入后有一个交叉项，但求和后为零

$$J(C, d) = \underbrace{\frac{1}{2}\sum_{j=1}^{M} w_j ((y_j - y) - C(p_j - p))^{\mathrm{T}} ((y_j - y) - C(p_j - p))}_{\text{仅依赖于}C} + \underbrace{\frac{1}{2} d^{\mathrm{T}} d}_{\text{仅依赖于}d}$$

- 这允许我们：(1)找C使得第一项最小化；(2)计算d使得第二项为零；

点云对准

- 对C的部分，有：

$$((\boldsymbol{y}_j-\boldsymbol{y})-\boldsymbol{C}(\boldsymbol{p}_j-\boldsymbol{p}))^{\mathrm{T}}((\boldsymbol{y}_j-\boldsymbol{y})-\boldsymbol{C}(\boldsymbol{p}_j-\boldsymbol{p}))$$
$$=\underbrace{(\boldsymbol{y}_j-\boldsymbol{y})^{\mathrm{T}}(\boldsymbol{y}_j-\boldsymbol{y})}_{\text{独立于}\boldsymbol{C}}-2\underbrace{((\boldsymbol{y}_j-\boldsymbol{y})^{\mathrm{T}}\boldsymbol{C}(\boldsymbol{p}_j-\boldsymbol{p}))}_{\operatorname{tr}(\boldsymbol{C}(\boldsymbol{p}_j-\boldsymbol{p})(\boldsymbol{y}_j-\boldsymbol{y})^{\mathrm{T}})}+\underbrace{(\boldsymbol{p}_j-\boldsymbol{p})^{\mathrm{T}}(\boldsymbol{p}_j-\boldsymbol{p})}_{\text{独立于}\boldsymbol{C}}$$

- 中间项求和后：

$$\frac{1}{w}\sum_{j=1}^{M} w_j((\boldsymbol{y}_j-\boldsymbol{y})^{\mathrm{T}}\boldsymbol{C}(\boldsymbol{p}_j-\boldsymbol{p}))=\frac{1}{w}\sum_{j=1}^{M} w_j\operatorname{tr}(\boldsymbol{C}(\boldsymbol{p}_j-\boldsymbol{p})(\boldsymbol{y}_j-\boldsymbol{y})^{\mathrm{T}})$$
$$=\operatorname{tr}\left(\boldsymbol{C}\frac{1}{w}\sum_{j=1}^{M} w_j(\boldsymbol{p}_j-\boldsymbol{p})(\boldsymbol{y}_j-\boldsymbol{y})^{\mathrm{T}}\right)=\operatorname{tr}(\boldsymbol{C}\boldsymbol{W}^{\mathrm{T}})$$

其中：

$$\boldsymbol{W}=\frac{1}{w}\sum_{j=1}^{M} w_j(\boldsymbol{y}_j-\boldsymbol{y})(\boldsymbol{p}_j-\boldsymbol{p})^{\mathrm{T}}$$

点云对准

- 那么，定义仅关于C的代价函数，既能最小化原问题，又保证C满足旋转矩阵约束：

$$J(\boldsymbol{C},\boldsymbol{\Lambda},\gamma) = -\operatorname{tr}(\boldsymbol{C}\boldsymbol{W}^{\mathrm{T}}) + \underbrace{\operatorname{tr}(\boldsymbol{\Lambda}(\boldsymbol{C}\boldsymbol{C}^{\mathrm{T}} - \mathbf{1})) + \gamma(\det \boldsymbol{C} - 1)}_{\text{拉格朗日乘子项}}$$

- 求目标函数关于各变量导数：

$$\frac{\partial J}{\partial \boldsymbol{C}} = -\boldsymbol{W} + 2\boldsymbol{\Lambda}\boldsymbol{C} + \gamma \underbrace{\det \boldsymbol{C}}_{1} \underbrace{\boldsymbol{C}^{-\mathrm{T}}}_{\boldsymbol{C}} = -\boldsymbol{W} + \boldsymbol{L}\boldsymbol{C}$$

其中 $\boldsymbol{L} = 2\boldsymbol{\Lambda} + \gamma\mathbf{1}$

$$\frac{\partial J}{\partial \boldsymbol{\Lambda}} = \boldsymbol{C}\boldsymbol{C}^{\mathrm{T}} - \mathbf{1}$$

$$\frac{\partial J}{\partial \gamma} = \det \boldsymbol{C} - 1$$

□ 用到的矩阵求导公式：

$$\frac{\partial}{\partial \boldsymbol{A}} \det \boldsymbol{A} = \det(\boldsymbol{A})\boldsymbol{A}^{-\mathrm{T}}$$

$$\frac{\partial}{\partial \boldsymbol{A}} \operatorname{tr}(\boldsymbol{A}\boldsymbol{B}^{\mathrm{T}}) = \boldsymbol{B}$$

$$\frac{\partial}{\partial \boldsymbol{A}} \operatorname{tr}(\boldsymbol{B}\boldsymbol{A}\boldsymbol{A}^{\mathrm{T}}) = (\boldsymbol{B} + \boldsymbol{B}^{\mathrm{T}})\boldsymbol{A}$$

- 第一项为零给出： $\boldsymbol{L}\boldsymbol{C} = \boldsymbol{W}$

点云对准

- 也可以不使用拉格朗日乘子，得到李群的方法
- 设旋转矩阵左扰动： $\mathbf{C}' = \exp(\boldsymbol{\phi}^\wedge)\mathbf{C}$
- 利用左扰动的定义式：

$$
\begin{aligned}
\frac{\partial J}{\partial \phi_i} &= \lim_{h\to 0} \frac{J(\mathbf{C}') - J(\mathbf{C})}{h} \\
&= \lim_{h\to 0} \frac{-\operatorname{tr}(\mathbf{C}'\mathbf{W}^T) + \operatorname{tr}(\mathbf{C}\mathbf{W}^T)}{h} \\
&= \lim_{h\to 0} \frac{-\operatorname{tr}(\exp(h\mathbf{1}_i^\wedge)\mathbf{C}\mathbf{W}^T) + \operatorname{tr}(\mathbf{C}\mathbf{W}^T)}{h} \\
&\approx \lim_{h\to 0} \frac{-\operatorname{tr}((\mathbf{1} + h\mathbf{1}_i^\wedge)\mathbf{C}\mathbf{W}^T) + \operatorname{tr}(\mathbf{C}\mathbf{W}^T)}{h} \\
&= \lim_{h\to 0} \frac{-\operatorname{tr}(h\mathbf{1}_i^\wedge\mathbf{C}\mathbf{W}^T)}{h} \\
&= -\operatorname{tr}(\mathbf{1}_i^\wedge\mathbf{C}\mathbf{W}^T)
\end{aligned}
$$

设导数为零，得到：

$$(\forall i)\ \operatorname{tr}(\mathbf{1}_i^\wedge \underbrace{\mathbf{C}\mathbf{W}^T}_{\mathbf{L}}) = 0$$

此式说明L为对称矩阵
取转置右乘C后得到：

$$\mathbf{L}\mathbf{C} = \mathbf{W}$$

点云对准

- 之后的推导有简略版和完整版两个版本，我们课上推导简略版，完整版留给同学自己阅读
- 假设：$\det \boldsymbol{W} > 0$ 根据：$\boldsymbol{L}\boldsymbol{C} = \boldsymbol{W}$ ，那么它自乘自己转置时，得：

$$\boldsymbol{L}\underbrace{\boldsymbol{C}\boldsymbol{C}^{\mathrm{T}}}_{1}\boldsymbol{L}^{\mathrm{T}} = \boldsymbol{W}\boldsymbol{W}^{\mathrm{T}}$$

- 相当于对WW^T做Cholesky分解，于是：

$$\boldsymbol{L} = (\boldsymbol{W}\boldsymbol{W}^{\mathrm{T}})^{\frac{1}{2}}$$

- 于是：$\boldsymbol{C} = (\boldsymbol{W}\boldsymbol{W}^{\mathrm{T}})^{-\frac{1}{2}}\boldsymbol{W}$
- 这和我们之前提到把R转到SO(3)上的投影是一致的：$\boldsymbol{R} = \left(\boldsymbol{C}\boldsymbol{C}^{\mathrm{T}}\right)^{-\frac{1}{2}}\boldsymbol{C}$

- ❑ 矩阵平方根：将矩阵对角化，然后对特征值矩阵求平方根即可
- ❑ 实对称矩阵有效

点云对准

- 完整版（思路）
- 不假设det(W)>0的条件（det(W)>0意味着点云的形状正常且不退化）
- 讨论W的奇异值情况

- 完整版的详细推导留给同学课后完成

点云对准

- 完整版结论
- 如果W满足下列三种条件之一：
  1. det(W) >0
  2. det(W)<0, W的最小奇异值不重复
  3. rank(W)=2
- 那么解法为：
  1. W作奇异值分解：$W = UDV^{\mathrm{T}}$
  2. 设 $S = \mathrm{diag}(1, 1, \det U \det V)$
  3. 那么 $C = USV^{\mathrm{T}}$

- ☐ 如果C存在唯一全局解，那么它的解必为此式
- ☐ 如果不存在唯一全局解，那么C存在无数种解

- 平移部分由质心给出

点云对准

- 变换矩阵解法
- 使用之前定义好的变量：

$$\mathbf{y}_j = \begin{bmatrix} \mathbf{y}_j \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{r}_{v_k}^{p_j v_k} \\ 1 \end{bmatrix}, \quad \mathbf{p}_j = \begin{bmatrix} \mathbf{p}_j \\ 1 \end{bmatrix} = \begin{bmatrix} \mathbf{r}_i^{p_j i} \\ 1 \end{bmatrix} \quad \mathbf{T} = \mathbf{T}_{v_k i} = \begin{bmatrix} \mathbf{C}_{v_k i} & -\mathbf{C}_{v_k i}\mathbf{r}_i^{v_k i} \\ \mathbf{0}^{\mathrm{T}} & 1 \end{bmatrix}$$

- 注意使用SE(3)时，要用齐次坐标
- 误差定义为： $\mathbf{e}_j = \mathbf{y}_j - \mathbf{T}\mathbf{p}_j$
- 目标函数：

$$J(\mathbf{T}) = \frac{1}{2}\sum_{j=1}^{M} w_j \mathbf{e}_j^{\mathrm{T}}\mathbf{e}_j = \frac{1}{2}\sum_{j=1}^{M} w_j (\mathbf{y}_j - \mathbf{T}\mathbf{p}_j)^{\mathrm{T}}(\mathbf{y}_j - \mathbf{T}\mathbf{p}_j)$$

点云对准

- 使用SE(3)上的左扰动，在工作点附近添加左扰动为：

$$\boldsymbol{T} = \exp(\boldsymbol{\epsilon}^{\wedge})\boldsymbol{T}_{\text{op}} \approx (\boldsymbol{1} + \boldsymbol{\epsilon}^{\wedge})\boldsymbol{T}_{\text{op}}$$

- 那么目标函数变为：

$$J(\boldsymbol{T}) \approx \frac{1}{2}\sum_{j=1}^{M} w_j \left((\boldsymbol{y}_j - \boldsymbol{z}_j) - \boldsymbol{z}_j^{\odot}\boldsymbol{\epsilon}\right)^{\mathrm{T}}\left((\boldsymbol{y}_j - \boldsymbol{z}_j) - \boldsymbol{z}_j^{\odot}\boldsymbol{\epsilon}\right)$$

- 其中 $\boldsymbol{z}_j = \boldsymbol{T}_{\text{op}}\boldsymbol{p}_j$ ，并且 $\boldsymbol{\epsilon}^{\wedge}\boldsymbol{z}_j = \boldsymbol{z}_j^{\odot}\boldsymbol{\epsilon}$
- 整个问题是一个无约束优化问题（且是二次的）

点云对准

- 直接用目标函数对扰动求导： $\frac{\partial J}{\partial \epsilon^{\mathrm{T}}} = -\sum_{j=1}^{M} w_j \boldsymbol{z}_j^{\odot^{\mathrm{T}}} \left((\boldsymbol{y}_j - \boldsymbol{z}_j) - \boldsymbol{z}_j^{\odot} \epsilon \right)$
- 令其为零，得到最优的扰动量： $\left(\frac{1}{w} \sum_{j=1}^{M} w_j \boldsymbol{z}_j^{\odot^{\mathrm{T}}} \boldsymbol{z}_j^{\odot} \right) \epsilon^{\star} = \frac{1}{w} \sum_{j=1}^{M} w_j \boldsymbol{z}_j^{\odot^{\mathrm{T}}} (\boldsymbol{y}_j - \boldsymbol{z}_j)$
- 该式给出了迭代步长的解法，但公式里z是依赖于T的工作点的
- 下面进行化简

点云对准

$$\left(\frac{1}{w}\sum_{j=1}^{M} w_j \mathbf{z}_j^{\odot^{\mathrm{T}}} \mathbf{z}_j^{\odot}\right)\boldsymbol{\epsilon}^{\star} = \frac{1}{w}\sum_{j=1}^{M} w_j \mathbf{z}_j^{\odot^{\mathrm{T}}} (\mathbf{y}_j - \mathbf{z}_j)$$

- 左侧:

$$\frac{1}{w}\sum_{j=1}^{M} w_j \mathbf{z}_j^{\odot^{\mathrm{T}}} \mathbf{z}_j^{\odot} = \underbrace{\boldsymbol{\mathcal{T}}_{\mathrm{op}}^{-\mathrm{T}}}_{>0} \underbrace{\left(\frac{1}{w}\sum_{j=1}^{M} w_j \mathbf{p}_j^{\odot^{\mathrm{T}}} \mathbf{p}_j^{\odot}\right)}_{\boldsymbol{\mathcal{M}}} \underbrace{\boldsymbol{\mathcal{T}}_{\mathrm{op}}^{-1}}_{>0}$$

- 其中:

$$\boldsymbol{\mathcal{T}}_{\mathrm{op}} = \mathrm{Ad}(\mathbf{T}_{\mathrm{op}}), \quad \boldsymbol{\mathcal{M}} = \begin{bmatrix} \mathbf{1} & \mathbf{0} \\ -\mathbf{p}^{\wedge} & \mathbf{1} \end{bmatrix} \begin{bmatrix} \mathbf{1} & \mathbf{0} \\ \mathbf{0} & \mathbf{1} \end{bmatrix} \begin{bmatrix} \mathbf{1} & \mathbf{p}^{\wedge} \\ \mathbf{0} & \mathbf{1} \end{bmatrix},$$

$$w = \sum_{j=1}^{M} w_j, \quad \mathbf{p} = \frac{1}{w}\sum_{j=1}^{M} w_j \mathbf{p}_j, \quad \mathbf{I} = -\frac{1}{w}\sum_{j=1}^{M} w_j (\mathbf{p}_j - \mathbf{p})^{\wedge}(\mathbf{p}_j - \mathbf{p})^{\wedge}$$

- 花体M形如广义质量矩阵，为常量

点云对准

$$\left(\frac{1}{w}\sum_{j=1}^{M} w_j \boldsymbol{z}_j^{\odot^{\mathrm{T}}} \boldsymbol{z}_j^{\odot}\right) \boldsymbol{\epsilon}^{\star} = \frac{1}{w}\sum_{j=1}^{M} w_j \boldsymbol{z}_j^{\odot^{\mathrm{T}}} (\boldsymbol{y}_j - \boldsymbol{z}_j)$$

- 右侧: $$\boldsymbol{a} = \frac{1}{w}\sum_{j=1}^{M} w_j \boldsymbol{z}_j^{\odot^{\mathrm{T}}} (\boldsymbol{y}_j - \boldsymbol{z}_j) = \begin{bmatrix} \boldsymbol{y} - \boldsymbol{C}_{\mathrm{op}}(\boldsymbol{p} - \boldsymbol{r}_{\mathrm{op}}) \\ \boldsymbol{b} - \boldsymbol{y}^{\wedge}\boldsymbol{C}_{\mathrm{op}}(\boldsymbol{p} - \boldsymbol{r}_{\mathrm{op}}) \end{bmatrix}$$

- 其中: $$\boldsymbol{b} = [\mathrm{tr}(\mathbf{1}_i^{\wedge} \boldsymbol{C}_{\mathrm{op}} \boldsymbol{W}^{\mathrm{T}})]_i, \quad \boldsymbol{T}_{\mathrm{op}} = \begin{bmatrix} \boldsymbol{C}_{\mathrm{op}} & -\boldsymbol{C}_{\mathrm{op}}\boldsymbol{r}_{\mathrm{op}} \\ \mathbf{0}^{\mathrm{T}} & 1 \end{bmatrix}$$

$$\boldsymbol{W} = \frac{1}{w}\sum_{j=1}^{M} w_j (\boldsymbol{y}_j - \boldsymbol{y})(\boldsymbol{p}_j - \boldsymbol{p})^{\mathrm{T}}, \quad \boldsymbol{y} = \frac{1}{w}\sum_{j=1}^{M} w_j \boldsymbol{y}_j$$

- 那么每一次迭代可由: $\boldsymbol{\epsilon}^{\star} = \boldsymbol{\mathcal{T}}_{\mathrm{op}} \boldsymbol{\mathcal{M}}^{-1} \boldsymbol{\mathcal{T}}_{\mathrm{op}}^{\mathrm{T}} \boldsymbol{a}$ 给出
- 该解法只和当时工作点取值相关，因此可以不断迭代至收敛

点云对准

- 小结

- 关于点云对准问题，我们从四元数、旋转矩阵、SE3流形三种角度来进行了推导
- 它们的结果大同小异，四元数和旋转矩阵可以一步得到，SE3流形则以迭代方式得到
- 常用的推导以旋转矩阵为主，我们需要关心它通常的解与退化条件

第7讲 位姿估计问题

- 点云对准
- 点云跟踪
- 位姿图松驰化

点云跟踪

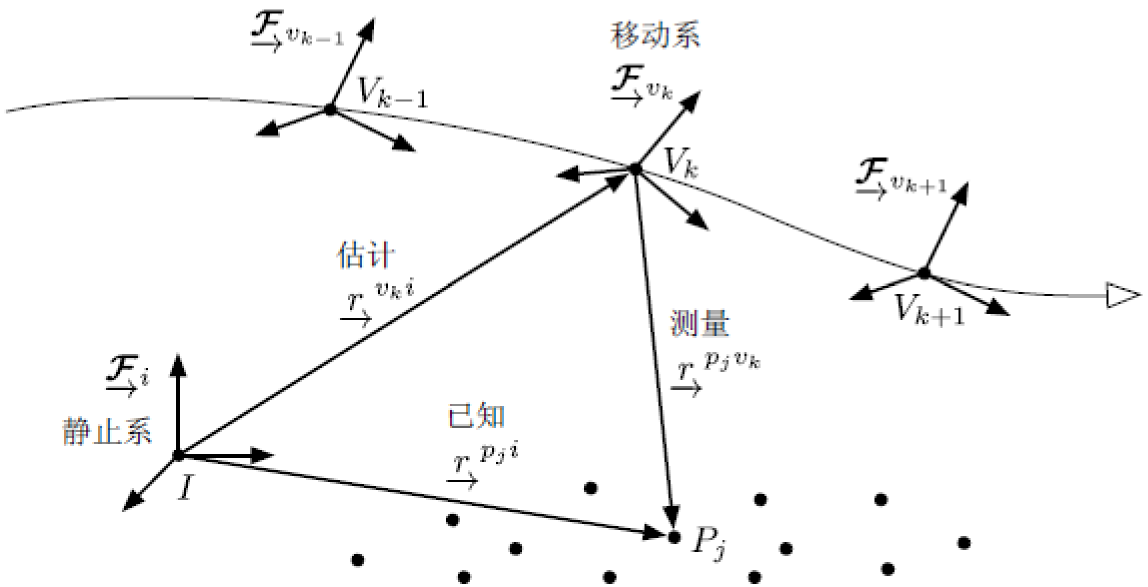
- 继续考虑上面提到的问题，但这时认为机器人在随时间运动
- 以SE(3)变量表达机器人变换矩阵：

□ 注意这时候T的平移部分不再是坐标

$$T_k = T_{v_k i} = \begin{bmatrix} C_{v_k i} & -C_{v_k i} r_i^{v_k i} \\ 0^T & 1 \end{bmatrix}$$

- 初始值： $\check{T}_0$ 以及它的不确定性

- 下面我们给出运动方程与观测方程，并给出对应的EKF滤波算法与优化算法



点云跟踪

- 运动方程（连续） 使用SE(3)的运动学

$$\dot{r} = \omega^\wedge r + \nu \qquad \dot{C} = \omega^\wedge C \qquad \varpi = \begin{bmatrix} \nu \\ \omega \end{bmatrix} \begin{matrix} \text{线速度} \\ \text{角速度} \end{matrix} \qquad \dot{T} = \varpi^\wedge T$$

- 广义速度也受噪声影响： $\varpi = \bar{\varpi} + \delta\varpi$
- 于是把这个运动分成两个部分：

标称运动：$\dot{\bar{T}} = \bar{\varpi}^\wedge \bar{T}$

扰动运动：$\delta\dot{\xi} = \bar{\varpi}^\curlywedge \delta\xi + \delta\varpi$

离散化后：

标称运动：$\bar{T}_k = \underbrace{\exp(\Delta t_k \bar{\varpi}_k^\wedge)}_{\Xi_k} \bar{T}_{k-1}$

扰动运动：$\delta\xi_k = \underbrace{\exp(\Delta t_k \bar{\varpi}_k^\curlywedge)}_{\mathrm{Ad}(\Xi_k)} \delta\xi_{k-1} + w_k$

真正的噪声：$w_k = \mathcal{N}(0, Q_k)$

点云跟踪

- 测量模型： $y_{jk} = D^{\mathrm{T}} T_k p_j + n_{jk}$
- 其中：

$$p_j = \begin{bmatrix} r_i^{p_j i} \\ 1 \end{bmatrix} \qquad D^{\mathrm{T}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

为齐次坐标　　为齐次转非齐次的矩阵

- 测量噪声： $n_{jk} \sim \mathcal{N}(0, R_{jk})$

- 该方程相对于T的线性化：

$$\bar{y}_{jk} + \delta y_{jk} = D^{\mathrm{T}} \left(\exp\left(\delta \xi_k^{\wedge}\right) \bar{T}_k \right) p_j + n_{jk}$$

$$\delta y_{jk} \approx D^{\mathrm{T}} \left(\bar{T}_k p_j \right)^{\odot} \delta \xi_k + n_{jk}$$

点云跟踪

- 推导过程中的符号：

下面先看EKF，然后看MAP估计

$\hat{\boldsymbol{T}}_k$ ： 在时刻 k 的 4×4 的校正后位姿估计

$\hat{\boldsymbol{P}}_k$ ： 在时刻 k 的 6×6 的校正后位姿估计协方差（含平移量和旋转量）

$\check{\boldsymbol{T}}_k$ ： 在时刻 k 的 4×4 的预测位姿估计

$\check{\boldsymbol{P}}_k$ ： 在时刻 k 的 6×6 的预测位姿估计协方差（含平移量和旋转量）

$\check{\boldsymbol{T}}_0$ ： 4×4 的先验输入，作为在时刻 0 的位姿

$\boldsymbol{\varpi}_k$ ： 6×1 的先验输入，作为在时刻 k 的广义速度

$\boldsymbol{Q}_k$ ： 6×6 的过程噪声协方差（含平移量和旋转量）

$\boldsymbol{y}_{jk}$ ： 机器人在时刻 k 对点 j 的 3×1 的测量

$\boldsymbol{R}_{jk}$ ： 在时刻 k 的测量 j 的 3×3 的协方差

点云跟踪

- EKF预测：　均值 $\check{T}_k = \underbrace{\exp(\Delta t_k \varpi_k^{\wedge})}_{\Xi_k} \hat{T}_{k-1}$ 　方差 $\check{P}_k = E[\delta\check{\xi}_k \delta\check{\xi}_k^{\mathrm{T}}]$

$$\delta\check{\xi}_k = \underbrace{\exp(\Delta t_k \varpi_k^{\curlywedge})}_{F_{k-1}=\mathrm{Ad}(\Xi_k)} \delta\hat{\xi}_{k-1} + w_k$$

- 线性化处系数矩阵为： $F_{k-1} = \exp(\Delta t_k \varpi_k^{\curlywedge})$

- 于是预测的协方差变为： $\check{P}_k = F_{k-1}\hat{P}_{k-1}F_{k-1}^{\mathrm{T}} + Q_k$

点云跟踪

- 校正部分
- 测量模型线性化为： $\delta y_{jk} = \underbrace{D^{\mathrm{T}}(\check{T}_k p_j)^{\odot}}_{G_{jk}} \delta\check{\xi}_k + n_{jk}$ 　雅可比矩阵 $G_{jk} = D^{\mathrm{T}}(\check{T}_k p_j)^{\odot}$
- 把M个点的观测写到一起： $$y_k = \begin{bmatrix} y_{1k} \\ \vdots \\ y_{Mk} \end{bmatrix}, \quad G_k = \begin{bmatrix} G_{1k} \\ \vdots \\ G_{Mk} \end{bmatrix}, \quad R_k = \mathrm{diag}(R_{1k}, \cdots, R_{Mk})$$
- 那么校正部分给出： $$K_k = \check{P}_k G_k^{\mathrm{T}} (G_k \check{P}_k G_k^{\mathrm{T}} + R_k)^{-1}$$ $$\hat{P}_k = (1 - K_k G_k)\check{P}_k$$
- 此式用于迭代方差部分

点云跟踪

- 校正均值部分
- 先计算更新量：$\epsilon_k = \underbrace{\ln\left(\hat{T}_k \check{T}_k^{-1}\right)^{\vee}}_{\text{更新}} = K_k \underbrace{(y_k - \check{y}_k)}_{\text{革新量}}$
- 然后再更新到先验变换矩阵上：

$$\hat{T}_k = \exp(\epsilon_k^{\wedge})\check{T}_k$$

综合起来：

$$\begin{aligned}
\text{预测：}\quad & \check{P}_k = F_{k-1}\hat{P}_{k-1}F_{k-1}^{\mathrm{T}} + Q_k \\
& \check{T}_k = \Xi_k \hat{T}_{k-1} \\
\text{卡尔曼增益：}\quad & K_k = \check{P}_k G_k^{\mathrm{T}}(G_k \check{P}_k G_k^{\mathrm{T}} + R_k)^{-1} \\
\text{校正：}\quad & \hat{P}_k = (1 - K_k G_k)\check{P}_k \\
& \hat{T}_k = \exp\left((K_k(y_k - \check{y}_k))^{\wedge}\right)\check{T}_k
\end{aligned}$$

这种计算方式保证状态变量一直在SE(3)上
协方差一直定义在李代数空间上

点云跟踪

- MAP解法
- 运动方程误差:

$$e_{v,k}(x) = \begin{cases} \ln(\check{T}_0 T_0^{-1})^\vee & k=0 \\ \ln(\Xi_k T_{k-1} T_k^{-1})^\vee & k=1,\cdots,K \end{cases}$$

其中: $\Xi_k = \exp(\Delta t_k \varpi_k^\wedge)$

- 测量方程误差: $e_{y,jk}(x) = y_{jk} - D^{\mathrm{T}} T_k p_j$
- 下面考察误差的协方差矩阵（噪声特性)

点云跟踪

- 运动方程：
- 初始值 $e_{v,0}(x) = \ln\left(\check{T}_0 T_0^{-1}\right)^\vee = \ln\left(\check{T}_0 \check{T}_0^{-1} \exp\left(-\delta\xi_0^\wedge\right)\right)^\vee = -\delta\xi_0$ $\qquad e_{v,0}(x) \sim \mathcal{N}(0, \check{P}_0)$
- 后续：

$$
\begin{aligned}
e_{v,k}(x) &= \ln(\Xi_k T_{k-1} T_k^{-1})^\vee \\
&= \ln(\Xi_k \exp(\delta\xi_{k-1}^\wedge) \check{T}_{k-1} \check{T}_k^{-1} \exp(-\delta\xi_k^\wedge))^\vee \\
&= \ln\left(\underbrace{\Xi_k \check{T}_{k-1} \check{T}_k^{-1}}_{1} \exp\left((\mathrm{Ad}(\Xi_k)\delta\xi_{k-1})^\wedge\right) \exp(-\delta\xi_k^\wedge)\right)^\vee \\
&\approx \mathrm{Ad}(\Xi_k)\delta\xi_{k-1} - \delta\xi_k \\
&= -w_k
\end{aligned}
$$

$$e_{v,k}(x) \sim \mathcal{N}(0, Q_k)$$

点云跟踪

- 观测模型：

$$e_{y,jk}(x) = y_{jk} - D^{\mathrm{T}} T_k p_j = n_{jk} \qquad e_{y,jk}(x) \sim \mathcal{N}(0, R_{jk})$$

- 于是目标函数为：

$$J_{v,k}(x) = \begin{cases} \frac{1}{2} e_{v,0}(x)^{\mathrm{T}} \check{P}_0^{-1} e_{v,0}(x) & k = 0 \\ \frac{1}{2} e_{v,k}(x)^{\mathrm{T}} Q_k^{-1} e_{v,k}(x) & k = 1, \cdots, K \end{cases}$$

$$J_{y,k}(x) = \frac{1}{2} e_{y,k}(x)^{\mathrm{T}} R_k^{-1} e_{y,k}(x)$$

- 把M个点合写：

$$e_{y,k}(x) = \begin{bmatrix} e_{y,1k}(x) \\ \vdots \\ e_{y,Mk}(x) \end{bmatrix}, \qquad R_k = \mathrm{diag}(R_{1k}, \cdots, R_{Mk})$$

- 目标函数写为：

$$J(x) = \sum_{k=0}^{K} (J_{v,k}(x) + J_{y,k}(x))$$

点云跟踪

- 下面对各误差项进给线性化，并给出迭代求解方案
- 工作点设为：$T_{\mathrm{op},k}$ ，我们每次求工作点上的一个扰动： $T_k = \exp(\epsilon_k^\wedge) T_{\mathrm{op},k}$
- 每次迭代的工作点统一为： $x_{\mathrm{op}} = \{T_{\mathrm{op},1}, T_{\mathrm{op},2}, \cdots, T_{\mathrm{op},K}\}$
- 初始误差项： $e_{v,0}(x) = \ln(\check{T}_0 T_0^{-1})^\vee = \ln\Big(\underbrace{\check{T}_0 T_{\mathrm{op},0}^{-1}}_{\exp(e_{v,0}(x_{\mathrm{op}})^\wedge)} \exp(-\epsilon_0^\wedge)\Big)^\vee \approx e_{v,0}(x_{\mathrm{op}}) - \epsilon_0$

点云跟踪

- 运动误差项：

$$
\begin{aligned}
e_{v,k}(x) &= \ln(\Xi_k T_{k-1} T_k^{-1})^{\vee} \\
&= \ln\left(\Xi_k \exp(\epsilon_{k-1}^{\wedge}) T_{\text{op},k-1} T_{\text{op},k}^{-1} \exp(-\epsilon_k^{\wedge})\right)^{\vee} \\
&= \ln\left(\underbrace{\Xi_k T_{\text{op},k-1} T_{\text{op},k}^{-1}}_{\exp(e_{v,k}(x_{\text{op}})^{\wedge})} \exp\left(\left(\text{Ad}(T_{\text{op},k} T_{\text{op},k-1}^{-1}) \epsilon_{k-1}\right)^{\wedge}\right) \exp(-\epsilon_k^{\wedge})\right)^{\vee} \\
&\approx e_{v,k}(x_{\text{op}}) + \underbrace{\text{Ad}(T_{\text{op},k} T_{\text{op},k-1}^{-1})}_{F_{k-1}} \epsilon_{k-1} - \epsilon_k
\end{aligned}
$$

- 测量误差项：

$$
\begin{aligned}
e_{y,jk}(x) &= y_{jk} - D^{\mathrm{T}} T_k p_j \\
&= y_{jk} - D^{\mathrm{T}} \exp(\epsilon_k^{\wedge}) T_{\text{op},k} p_j \\
&\approx y_{jk} - D^{\mathrm{T}} (1 + \epsilon_k^{\wedge}) T_{\text{op},k} p_j \\
&= \underbrace{y_{jk} - D^{\mathrm{T}} T_{\text{op},k} p_j}_{e_{y,jk}(x_{\text{op}})} - \underbrace{\left(D^{\mathrm{T}} (T_{\text{op},k} p_j)^{\odot}\right)}_{G_{jk}} \epsilon_k
\end{aligned}
$$

点云跟踪

- 把k时刻所有测量点写在一起，有： $e_{y,k}(x) \approx e_{y,k}(x_{\text{op}}) - G_k \epsilon_k$

$$e_{y,k}(x) = \begin{bmatrix} e_{y,1k}(x) \\ \vdots \\ e_{y,Mk}(x) \end{bmatrix}, \quad e_{y,k}(x_{\text{op}}) = \begin{bmatrix} e_{y,1k}(x_{\text{op}}) \\ \vdots \\ e_{y,Mk}(x_{\text{op}}) \end{bmatrix}, \quad G_k = \begin{bmatrix} G_{1k} \\ \vdots \\ G_{Mk} \end{bmatrix}$$

- 于是得到了各个雅可比

点云跟踪

- Putting them together
- 在每次高斯-牛顿迭代中，使用各矩阵：

$$\delta x = \begin{bmatrix} \epsilon_0 \\ \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_K \end{bmatrix}, \quad H = \left[\begin{array}{cccccc} 1 & & & & & \\ -F_0 & 1 & & & & \\ & -F_1 & \ddots & & & \\ & & \ddots & 1 & & \\ & & & -F_{K-1} & 1 & \\ \hline G_0 & & & & & \\ & G_1 & & & & \\ & & G_2 & & & \\ & & & & \ddots & \\ & & & & & G_K \end{array}\right], \quad e(x_{\text{op}}) = \left[\begin{array}{c} e_{v,0}(x_{\text{op}}) \\ e_{v,1}(x_{\text{op}}) \\ \vdots \\ e_{v,K-1}(x_{\text{op}}) \\ e_{v,K}(x_{\text{op}}) \\ \hline e_{y,0}(x_{\text{op}}) \\ e_{y,1}(x_{\text{op}}) \\ \vdots \\ e_{y,K-1}(x_{\text{op}}) \\ e_{y,K}(x_{\text{op}}) \end{array}\right] \tag{8.160}$$

$$W = \text{diag}(\check{P}_0, Q_1, \cdots, Q_K, R_0, R_1, \cdots, R_K)$$

点云跟踪

- 得到高斯-牛顿法的线性方程： $A\delta x^{\star}=b$
- 其中：

$$A=\underbrace{H^{\mathrm{T}}W^{-1}H}_{\text{三对角块}},\quad b=H^{\mathrm{T}}W^{-1}e(x_{\mathrm{op}})$$

- 把扰动左乘到当前估计值即可： $T_{\mathrm{op},k}\leftarrow\exp\left(\epsilon_k^{\star\wedge}\right)T_{\mathrm{op},k}$

第7讲 位姿估计问题

- 点云对准
- 点云跟踪
- 位姿图松驰化（略）

习题

理论部分：

1. 推导第9页r的表达式；
2. 推导第23页第1个式子；
3. 推导第24页第1个式子；

编程部分

1. 写一段ICP程序，完成两个PCD文件的Pose计算，不允许使用PCL或第三方点云库