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3讲 非线性高斯系统的状态估计问题

- 引言
- 离散时间的递归估计
- 离散时间的批量估计
- 连续时间的批量估计

第3讲 非线性高斯系统的状态估计问题

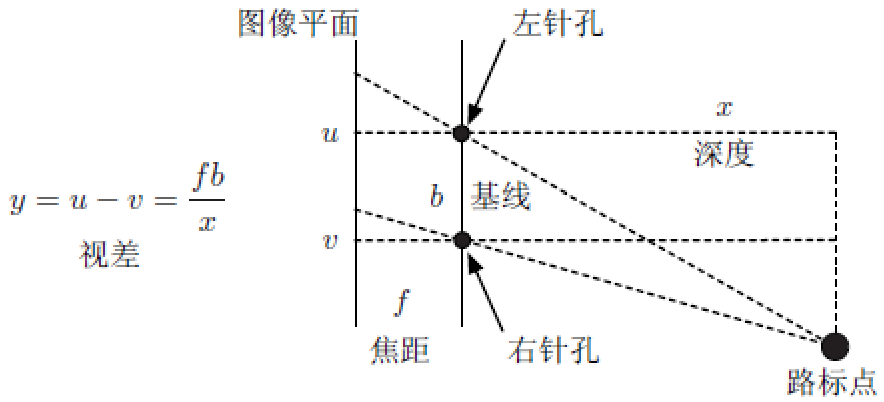
- 引言
- 离散时间的递归估计
- 离散时间的批量估计
- 连续时间的批量估计

引言

- 上一讲我们介绍了LG系统状态估计问题，得到了几个主要结论
  - LG系统的MAP估计和贝叶斯推断给出了相同的结果
  - 递归形式可由批量形式导出，我们以Cholesky分解作为例子，给出了前向和后向迭代，组成RTS Smoother
  - 前向部分等价于卡尔曼滤波器，而卡尔曼滤波器亦可从MAP或贝叶斯滤波形式导出
- 本讲我们主要介绍NL系统下的状态估计，主要方法有：
  - 通用的贝叶斯滤波理论框架；
  - NL下的MAP
  - EKF、Sigma point 卡尔曼滤波（UKF）、粒子滤波

引言

- 我们知道LG系统中，贝叶斯推断的结果与MAP结果一致
  - 这主要是因为高斯分布经过线性变换之后仍然为高斯分布，使得其均值与模相同
  - 但是在非线性模型中，这一点并不成立
- 我们从一个简单的例子入手：双目相机模型



双目相机的视差与距离成反比： $y = \frac{fb}{x} + n$

x：路标点深度；y：视差；f：焦距；b：基线；n：噪声

引言

- 考虑双目模型中，已知视差时，路标点距离的分布 $p(x|y)$
- 那么，使用贝叶斯推断：

$$p(x|y) = \frac{p(y|x)p(x)}{\int_{-\infty}^{\infty} p(y|x)p(x)\mathrm{d}x}$$

需要知道p(x)和p(y|x)

- p(y|x)是容易计算的，因为噪声服从高斯分布：

$$p(y|x) = \mathcal{N}\left(\frac{fb}{x}, R\right) = \frac{1}{\sqrt{2\pi R}}\exp\left(-\frac{1}{2R}\left(y - \frac{fb}{x}\right)^2\right)$$

- 同时也假设先验服从高斯分布： $p(x) = \mathcal{N}(\check{x}, \check{P}) = \frac{1}{\sqrt{2\pi\check{P}}}\exp\left(-\frac{1}{2\check{P}}(x-\check{x})^2\right)$

- 那么整件事情可以用蒙特卡洛仿真出来

引言

- 确定仿真的操作顺序： 给定先验 $\rightarrow$ 采样 x_{true} $\rightarrow$ 采样 y_{meas} $\rightarrow$ 计算后验

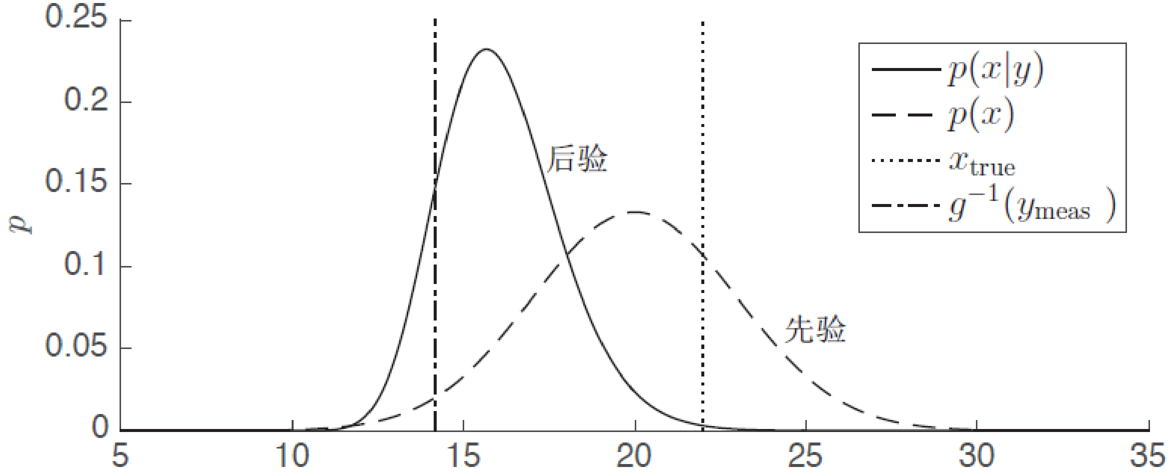
□ 注意仿真和实际是不同的

- 代入一些参数：

$$\check{x} = 20[\text{m}], \quad \check{P} = 9[\text{m}^2]$$

$$f = 400[\text{pixel}], \quad b = 0.1[\text{m}], \quad R = 0.09[\text{pixel}^2]$$

- 就可以得到后验估计的分布：



这个实验当中几个结论：

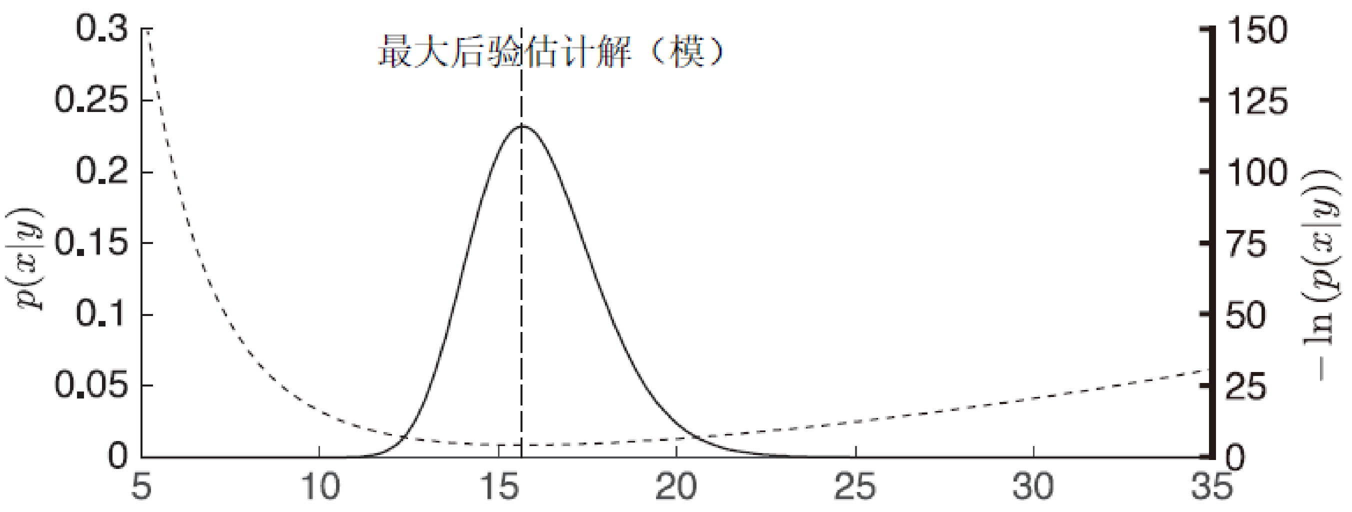
□ 后验比先验更加确定了（分布更“瘦”）
□ 后验不是对称的，左侧低右侧高
□ 后验是单峰的，容易用高斯分布近似之

它体现了状态估计的思想：

□ 将测量结果考虑进来后，得到更加确定的后验状态分布
□ 但是其实这个例子取值取得并不巧，你能看出来吗？

引言

- MAP解就是后验分布的最高值，同时也是-ln(p(x|y))的最低值
- 由于后验分布不是对称的，因此往往不等于均值



MAP目标函数：

$$\hat{x}_{\text{map}} = \arg\max_{x} p(x|y)$$

对应的负对数形式：

$$\hat{x}_{\text{map}} = \arg\min_{x}(-\ln(p(y|x)) - \ln(p(x)))$$

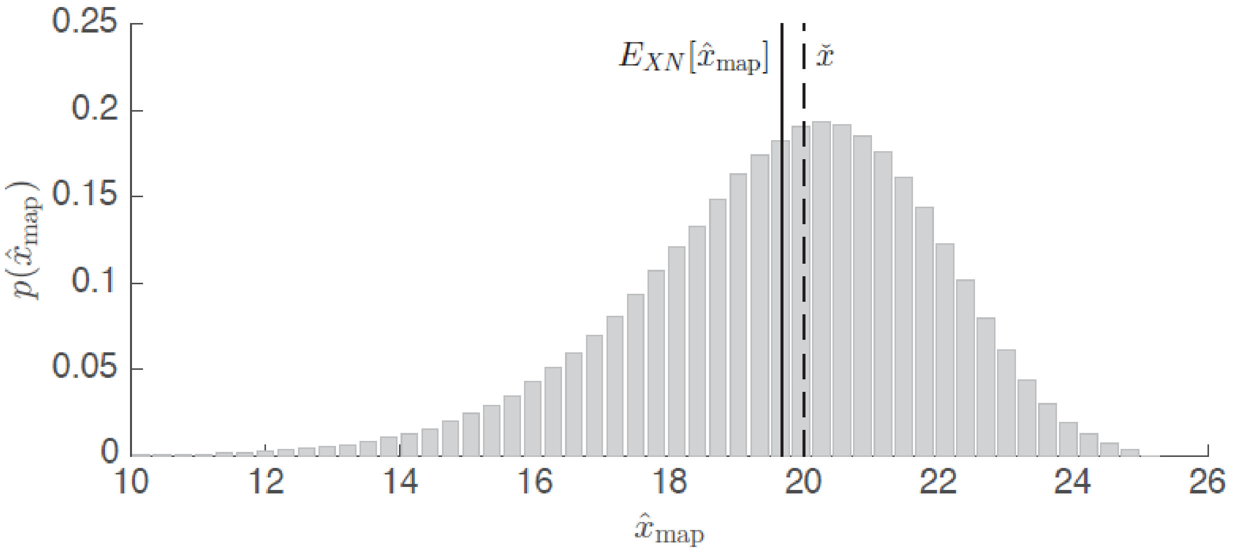
搁双目模型里：

$$J(x) = \frac{1}{2R}\left(y - \frac{fb}{x}\right)^2 + \frac{1}{2\check{P}}(\check{x} - x)^2$$

引言

- MAP的估计是否准确?
  - 我们可以计算后验分布与真值之差的期望：$e_{\text{mean}}(\hat{x}) = E_{XN}[\hat{x} - x_{\text{true}}]$
  - 显然MAP是有偏的，因为MAP值和先验均值并不在一条线上
  - 我们也可以用平均平方误差来衡量精度：$e_{\text{sq}}(\hat{x}) = E_{XN}[(\hat{x} - x_{\text{true}})^2]$

- ☐ 下标XN表示对X和N取样
- ☐ 上面的是均值与均值之差
- ☐ 下面的是个例之间的均值



这个例子中，结论是：

$$e_{\text{mean}} \approx -33.0\text{cm}$$
$$e_{\text{sq}} \approx 4.41\text{m}^2$$

- ☐ 第1项指标明显小很多，因为只显示了均值与均值之差异，单次估计的正反向可以消除
- ☐ 相比来说，激光的深度误差约在2cm左右

第3讲 非线性高斯系统的状态估计问题

- 引言
- 离散时间的递归估计
- 离散时间的批量估计
- 连续时间的批量估计

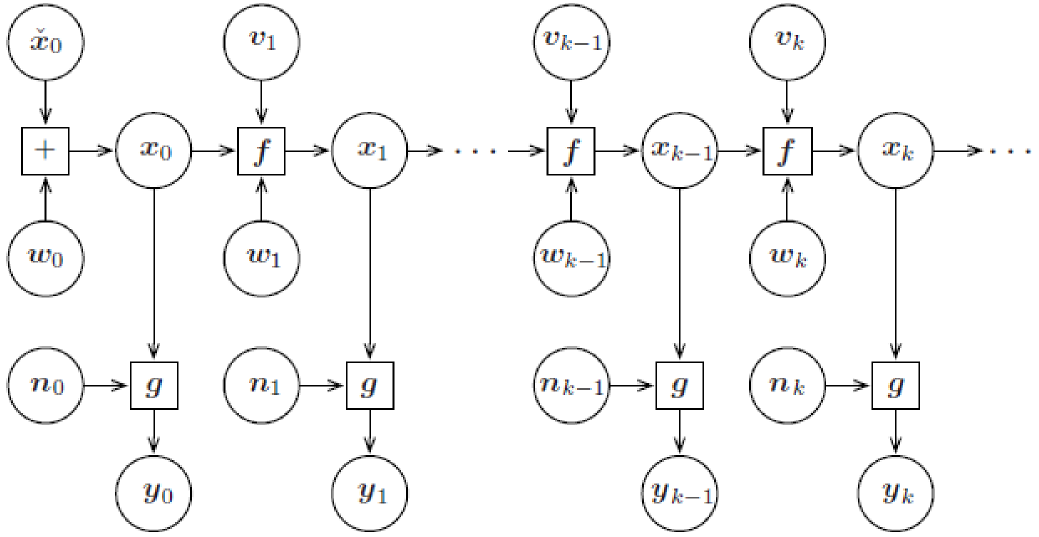
离散时间的递归估计

- 非线性系统下的运动方程与观测方程：

运动方程：$x_k = f(x_{k-1}, v_k, w_k), \quad k = 1, \cdots, K$

观测方程：$y_k = g(x_k, n_k), \quad k = 0, \cdots, K$

- 对应的图模型：



图模型的马尔可夫性：

- 给定现在时刻的状态时，未来时刻状态与过去时刻是条件独立的
- 具体地说，给定x(k-1)时，就可以递推地计算x(k)
- 在NLNG系统中是否有和LG系统一样的性质呢？

离散时间的递归估计

- 我们仍然从贝叶斯推断开始讨论，由于是递归的，也称为贝叶斯滤波
- 当前时刻的状态分布：$p(\boldsymbol{x}_k|\check{\boldsymbol{x}}_0, \boldsymbol{v}_{1:k}, \boldsymbol{y}_{0:k})$，同时假设各时刻的输入、观测都是独立的
- 用贝叶斯公式分离最新的观测y(k)：

$$p(\boldsymbol{x}_k|\check{\boldsymbol{x}}_0, \boldsymbol{v}_{1:k}, \boldsymbol{y}_{0:k}) = \eta p(\boldsymbol{y}_k|\boldsymbol{x}_k) p(\boldsymbol{x}_k|\check{\boldsymbol{x}}_0, \boldsymbol{v}_{1:k}, \boldsymbol{y}_{0:k-1})$$

- 第1项容易处理，第2项需要引入k-1时刻的状态：

$$p(\boldsymbol{x}_k|\check{\boldsymbol{x}}_0, \boldsymbol{v}_{1:k}, \boldsymbol{y}_{0:k-1}) = \int p(\boldsymbol{x}_k, \boldsymbol{x}_{k-1}|\check{\boldsymbol{x}}_0, \boldsymbol{v}_{1:k}, \boldsymbol{y}_{0:k-1}) \mathrm{d}\boldsymbol{x}_{k-1}$$

$$= \int p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}, \check{\boldsymbol{x}}_0, \boldsymbol{v}_{1:k}, \boldsymbol{y}_{0:k-1}) p(\boldsymbol{x}_{k-1}|\check{\boldsymbol{x}}_0, \boldsymbol{v}_{1:k}, \boldsymbol{y}_{0:k-1}) \mathrm{d}\boldsymbol{x}_{k-1}$$

- 又出现了联合=条件*边缘的形式
- 请读者对此保持敏感

- 而由于马尔可夫性，不必再引入k-1时刻之前的内容：

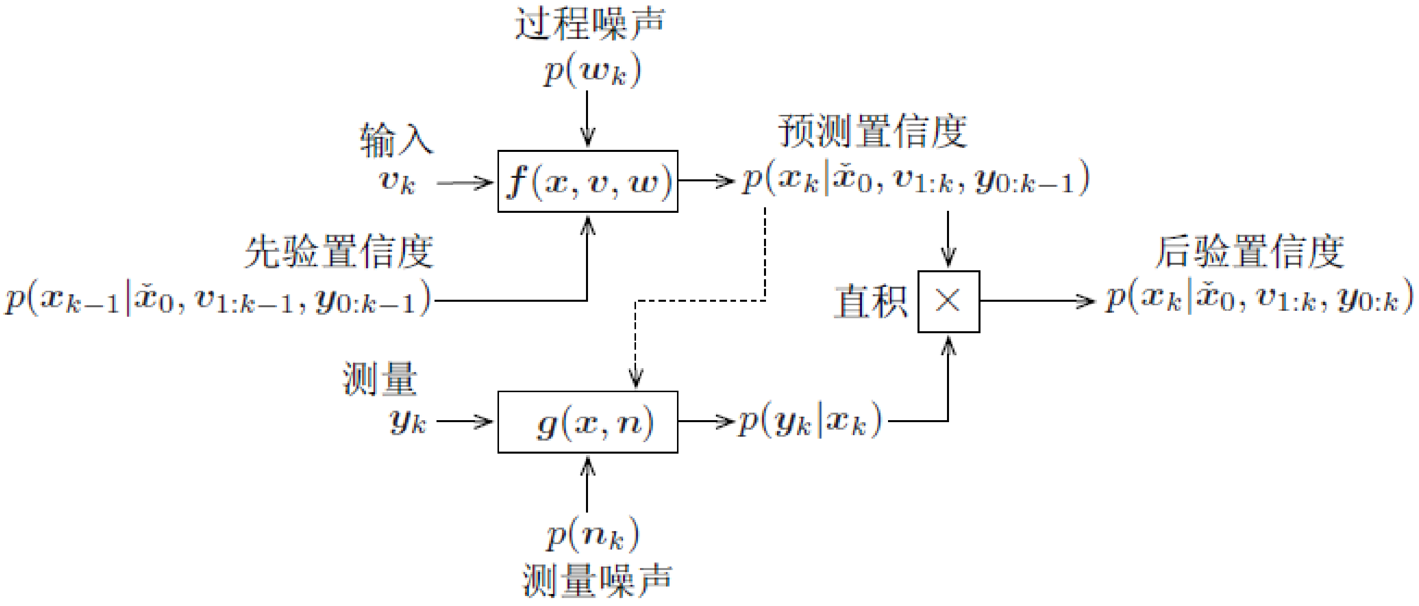
$$p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}, \check{\boldsymbol{x}}_0, \boldsymbol{v}_{1:k}, \boldsymbol{y}_{0:k-1}) = p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}, \boldsymbol{v}_k)$$

$$p(\boldsymbol{x}_{k-1}|\check{\boldsymbol{x}}_0, \boldsymbol{v}_{1:k}, \boldsymbol{y}_{0:k-1}) = p(\boldsymbol{x}_{k-1}|\check{\boldsymbol{x}}_0, \boldsymbol{v}_{1:k-1}, \boldsymbol{y}_{0:k-1})$$

离散时间的递归估计

- 综上，可整理出贝叶斯滤波器：

$$\underbrace{p(\boldsymbol{x}_k|\check{\boldsymbol{x}}_0, \boldsymbol{v}_{1:k}, \boldsymbol{y}_{0:k})}_{\text{后验置信度}} = \eta \underbrace{p(\boldsymbol{y}_k|\boldsymbol{x}_k)}_{\text{利用}\boldsymbol{g}(\cdot)\text{进行更新}} \int \underbrace{p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1}, \boldsymbol{v}_k)}_{\text{利用}\boldsymbol{f}(\cdot)\text{进行预测}} \underbrace{p(\boldsymbol{x}_{k-1}|\check{\boldsymbol{x}}_0, \boldsymbol{v}_{1:k-1}, \boldsymbol{y}_{0:k-1})}_{\text{先验置信度}} \mathrm{d}\boldsymbol{x}_{k-1}$$



好看吗？确实好看

- 怎么实现呢？
- 某种参数化形式来存储分布？（粒子、高斯）
- 怎么计算分布的非线性变换？（闭式、蒙特卡洛）
- 如果允许一定形式的近似，就会有一些经典结论
- 不要忘记这个过程中使用了马尔可夫性

离散时间的递归估计

- 扩展卡尔曼滤波（Extended Kalman Filter）
- 在贝叶斯滤波基础之上，假设状态分布为高斯的： $p(\boldsymbol{x}_k|\check{\boldsymbol{x}}_0, \boldsymbol{v}_{1:k}, \boldsymbol{y}_{0:k}) = \mathcal{N}\left(\hat{\boldsymbol{x}}_k, \hat{\boldsymbol{P}}_k\right)$
- 同时，噪声也是高斯的：

$$\boldsymbol{w}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{Q}_k)$$

$$\boldsymbol{n}_k \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{R}_k)$$

- 我们希望推导这种系统的闭式解。在第1讲中，我们谈论过线性化方法，现在对运动方程和观测方程进行线性化：

$$\boldsymbol{f}(\boldsymbol{x}_{k-1}, \boldsymbol{v}_k, \boldsymbol{w}_k) \approx \check{\boldsymbol{x}}_k + \boldsymbol{F}_{k-1}(\boldsymbol{x}_{k-1} - \hat{\boldsymbol{x}}_{k-1}) + \boldsymbol{w}'_k$$

$$\boldsymbol{g}(\boldsymbol{x}_k, \boldsymbol{n}_k) \approx \check{\boldsymbol{y}}_k + \boldsymbol{G}_k(\boldsymbol{x}_k - \check{\boldsymbol{x}}_k) + \boldsymbol{n}'_k$$

- 注意噪声也要线性化
- 线性化之后带一撇

其中：

$$\check{\boldsymbol{x}}_k = \boldsymbol{f}(\hat{\boldsymbol{x}}_{k-1}, \boldsymbol{v}_k, \boldsymbol{0}), \quad \boldsymbol{F}_{k-1} = \left.\frac{\partial \boldsymbol{f}(\boldsymbol{x}_{k-1}, \boldsymbol{v}_k, \boldsymbol{w}_k)}{\partial \boldsymbol{x}_{k-1}}\right|_{\hat{\boldsymbol{x}}_{k-1}, \boldsymbol{v}_k, \boldsymbol{0}} \qquad \check{\boldsymbol{y}}_k = \boldsymbol{g}(\check{\boldsymbol{x}}_k, \boldsymbol{0}), \quad \boldsymbol{G}_k = \left.\frac{\partial \boldsymbol{g}(\boldsymbol{x}_k, \boldsymbol{n}_k)}{\partial \boldsymbol{x}_k}\right|_{\check{\boldsymbol{x}}_k, \boldsymbol{0}}$$

$$\boldsymbol{w}'_k = \left.\frac{\partial \boldsymbol{f}(\boldsymbol{x}_{k-1}, \boldsymbol{v}_k, \boldsymbol{w}_k)}{\partial \boldsymbol{w}_k}\right|_{\hat{\boldsymbol{x}}_{k-1}, \boldsymbol{v}_k, \boldsymbol{0}} \boldsymbol{w}_k \qquad \boldsymbol{n}'_k = \left.\frac{\partial \boldsymbol{g}(\boldsymbol{x}_k, \boldsymbol{n}_k)}{\partial \boldsymbol{n}_k}\right|_{\check{\boldsymbol{x}}_k, \boldsymbol{0}} \boldsymbol{n}_k$$

离散时间的递归估计

- 线性化之后，模型就是线性的，处理方式和之前一样
- 运动方程的结果： $p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1},\boldsymbol{v}_k) \approx \mathcal{N}(\check{\boldsymbol{x}}_k + \boldsymbol{F}_{k-1}(\boldsymbol{x}_{k-1} - \hat{\boldsymbol{x}}_{k-1}), \boldsymbol{Q}'_k)$
- 观测方程的结果： $p(\boldsymbol{y}_k|\boldsymbol{x}_k) \approx \mathcal{N}(\check{\boldsymbol{y}}_k + \boldsymbol{G}_k(\boldsymbol{x}_k - \check{\boldsymbol{x}}_k), \boldsymbol{R}'_k)$
- 把这些代入贝叶斯滤波：

$$\underbrace{p(\boldsymbol{x}_k|\check{\boldsymbol{x}}_0,\boldsymbol{v}_{1:k},\boldsymbol{y}_{0:k})}_{\mathcal{N}(\hat{\boldsymbol{x}}_k,\hat{\boldsymbol{P}}_k)} = \eta \underbrace{p(\boldsymbol{y}_k|\boldsymbol{x}_k)}_{\mathcal{N}(\check{\boldsymbol{y}}_k+\boldsymbol{G}_k(\boldsymbol{x}_k-\check{\boldsymbol{x}}_k),\boldsymbol{R}'_k)} \times \int \underbrace{p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1},\boldsymbol{v}_k)}_{\mathcal{N}(\check{\boldsymbol{x}}_k+\boldsymbol{F}_{k-1}(\boldsymbol{x}_{k-1}-\hat{\boldsymbol{x}}_{k-1}),\boldsymbol{Q}'_k)} \underbrace{p(\boldsymbol{x}_{k-1}|\check{\boldsymbol{x}}_0,\boldsymbol{v}_{1:k-1},\boldsymbol{y}_{0:k-1})}_{\mathcal{N}(\hat{\boldsymbol{x}}_{k-1},\hat{\boldsymbol{P}}_{k-1})} \,\mathrm{d}\boldsymbol{x}_{k-1}$$

- 注意下方实际对应运动方程
- 这个式子的结论我们先前推导过

离散时间的递归估计

- 我们在第1讲中介绍过高斯分布的非线性变换：

$$p\left(\boldsymbol{y}|\boldsymbol{x}\right)=\mathcal{N}\left(\boldsymbol{g}\left(\boldsymbol{x}\right),\boldsymbol{R}\right)$$

$$\boldsymbol{g}\left(\boldsymbol{x}\right)\approx\boldsymbol{\mu}_y+\boldsymbol{G}\left(\boldsymbol{x}-\boldsymbol{\mu}_x\right)$$

$$\boldsymbol{G}=\left.\frac{\partial\boldsymbol{g}(\boldsymbol{x})}{\partial\boldsymbol{x}}\right|_{\boldsymbol{x}=\boldsymbol{\mu}_x}$$

$$\boldsymbol{\mu}_y=\boldsymbol{g}\left(\boldsymbol{\mu}_x\right)$$

$$\boldsymbol{y}\sim\mathcal{N}\left(\boldsymbol{\mu}_y,\boldsymbol{\Sigma}_{yy}\right)=\mathcal{N}\left(\boldsymbol{g}\left(\boldsymbol{\mu}_x\right),\boldsymbol{R}+\boldsymbol{G}\boldsymbol{\Sigma}_{xx}\boldsymbol{G}^{\mathrm{T}}\right)$$

- 所以第2项应为：

$$\underbrace{p(\boldsymbol{x}_k|\check{\boldsymbol{x}}_0,\boldsymbol{v}_{1:k},\boldsymbol{y}_{0:k})}_{\mathcal{N}(\hat{\boldsymbol{x}}_k,\hat{\boldsymbol{P}}_k)}=\eta\underbrace{p(\boldsymbol{y}_k|\boldsymbol{x}_k)}_{\mathcal{N}(\check{\boldsymbol{y}}_k+\boldsymbol{G}_k(\boldsymbol{x}_k-\check{\boldsymbol{x}}_k),\boldsymbol{R}'_k)}$$
$$\times\underbrace{\int p(\boldsymbol{x}_k|\boldsymbol{x}_{k-1},\boldsymbol{v}_k)p(\boldsymbol{x}_{k-1}|\check{\boldsymbol{x}}_0,\boldsymbol{v}_{1:k-1},\boldsymbol{y}_{0:k-1})\mathrm{d}\boldsymbol{x}_{k-1}}_{\mathcal{N}(\check{\boldsymbol{x}}_k,\boldsymbol{F}_{k-1}\hat{\boldsymbol{P}}_{k-1}\boldsymbol{F}_{k-1}^{\mathrm{T}}+\boldsymbol{Q}'_k)}$$

- 于是只须计算高斯分布的归一化积即可

离散时间的递归估计

- 经过一些乏味的代数运算（tedious algebra）可以推出EKF经典表达式（原话）
- 我们给出这里推导的思路，把具体步骤留作习题
  1. 写出归一化积指数部分，它们应该是x(k)的二次型；
  2. 比较x(k)二次项系数与一次项系数，得到协方差和均值的表达式；此时你得到信息形式的EKF；
  3. 定义卡尔曼增益，推导见第2讲37页，使用SMW得到它两种不同形式；
  4. 利用卡尔曼增益，改写第2步中的信息形式，整理成经典形式的EKF
- 实际上大部分推导过程在上讲里已经介绍过了，只需依样画葫芦即可

$$\underbrace{p(\mathbf{x}_k|\check{\mathbf{x}}_0, \mathbf{v}_{1:k}, \mathbf{y}_{0:k})}_{\mathcal{N}(\hat{\mathbf{x}}_k, \hat{\mathbf{P}}_k)} = \eta \underbrace{p(\mathbf{y}_k|\mathbf{x}_k)}_{\mathcal{N}(\check{\mathbf{y}}_k + \mathbf{G}_k(\mathbf{x}_k - \check{\mathbf{x}}_k), \mathbf{R}'_k)}$$
$$\times \underbrace{\int p(\mathbf{x}_k|\mathbf{x}_{k-1}, \mathbf{v}_k) p(\mathbf{x}_{k-1}|\check{\mathbf{x}}_0, \mathbf{v}_{1:k-1}, \mathbf{y}_{0:k-1}) d\mathbf{x}_{k-1}}_{\mathcal{N}(\check{\mathbf{x}}_k, \mathbf{F}_{k-1}\hat{\mathbf{P}}_{k-1}\mathbf{F}_{k-1}^T + \mathbf{Q}'_k)}$$

离散时间的递归估计

- EKF的经典形式：
  - 绝大部分与KF相同
  - 预测过程使用非线性模型得到先验均值；
  - 更新过程使用先验均值和非线性模型得到观测的预测值
- EKF的问题
  - 并不保证在一般非线性系统中适用或收敛；
  - 即使收敛时，也常常是有偏的或不一致的；
  - 线性化点是估计状态的均值，但离真值有可能较远；

预测：

$$\check{P}_k = F_{k-1}\hat{P}_{k-1}F_{k-1}^{\mathrm{T}} + Q_k'$$

$$\check{x}_k = f(\hat{x}_{k-1}, v_k, 0)$$

卡尔曼增益：

$$K_k = \check{P}_k G_k^{\mathrm{T}}(G_k\check{P}_k G_k^{\mathrm{T}} + R_k')^{-1}$$

更新：

$$\hat{P}_k = (1 - K_k G_k)\check{P}_k$$

$$\hat{x}_k = \check{x}_k + K_k \underbrace{(y_k - g(\check{x}_k, 0))}_{\text{更新量}}$$

离散时间的递归估计

- 广义高斯滤波
- 从k-1时刻的高斯先验开始： $p(x_{k-1}|\check{x}_0, v_{1:k-1}, y_{0:k-1}) = \mathcal{N}\left(\hat{x}_{k-1}, \hat{P}_{k-1}\right)$
- 通过运动模型，得到k时刻的先验： $p(x_k|\check{x}_0, v_{1:k}, y_{0:k-1}) = \mathcal{N}\left(\check{x}_k, \check{P}_k\right)$
- 校正部分，先写出联合分布，然后使用高斯推断：

$$p(x_k, y_k|\check{x}_0, v_{1:k}, y_{0:k-1}) = \mathcal{N}\left(\begin{bmatrix} \mu_{x,k} \\ \mu_{y,k} \end{bmatrix}, \begin{bmatrix} \Sigma_{xx,k} & \Sigma_{xy,k} \\ \Sigma_{yx,k} & \Sigma_{yy,k} \end{bmatrix}\right)$$

- 那么条件概率为：

$$p(x_k|\check{x}_0, v_{1:k}, y_{0:k}) = \mathcal{N}\Big(\underbrace{\mu_{x,k} + \Sigma_{xy,k}\Sigma_{yy,k}^{-1}(y_k - \mu_{y,k})}_{\hat{x}_k}, \underbrace{\Sigma_{xx,k} - \Sigma_{xy,k}\Sigma_{yy,k}^{-1}\Sigma_{yx,k}}_{\hat{P}_k}\Big)$$

- 注意这里并没有给出联合分布的计算方式
- 如果考虑观测的线性化，则y部分可以写出

离散时间的递归估计

- 然后定义卡尔曼增益：

$$\begin{aligned} K_k &= \Sigma_{xy,k}\Sigma_{yy,k}^{-1} \\ \hat{P}_k &= \check{P}_k - K_k\Sigma_{xy,k}^{\mathrm{T}} \\ \hat{x}_k &= \check{x}_k + K_k\left(y_k - \mu_{y,k}\right) \end{aligned}$$

- 就可以得到类EKF的过程
  - 但是这个思路中，如果不作线性化，则 $\mu_{y,k}$，$\Sigma_{yy,k}$ 和 $\Sigma_{xy,k}$无法计算，因为y是x的非线性变换，并不是高斯的
  - 不过，如果使用线性化，则可以进一步推导

离散时间的递归估计

- 迭代卡尔曼滤波
  - 首先对观测部分进行线性化：

$$g(\boldsymbol{x}_k, \boldsymbol{n}_k) \approx \boldsymbol{y}_{\mathrm{op},k} + \boldsymbol{G}_k(\boldsymbol{x}_k - \boldsymbol{x}_{\mathrm{op},k}) + \boldsymbol{n}'_k$$

其中

$$\boldsymbol{y}_{\mathrm{op},k} = \boldsymbol{g}(\boldsymbol{x}_{\mathrm{op},k}, \boldsymbol{0}), \quad \boldsymbol{G}_k = \left.\frac{\partial \boldsymbol{g}(\boldsymbol{x}_k, \boldsymbol{n}_k)}{\partial \boldsymbol{x}_k}\right|_{\boldsymbol{x}_{\mathrm{op},k}, \boldsymbol{0}}$$

$$\boldsymbol{n}'_k = \left.\frac{\partial \boldsymbol{g}(\boldsymbol{x}_k, \boldsymbol{n}_k)}{\partial \boldsymbol{n}_k}\right|_{\boldsymbol{x}_{\mathrm{op},k}, \boldsymbol{0}} \boldsymbol{n}_k$$

  - 那么k时刻状态与观测的联合分布可写为：

$$p(\boldsymbol{x}_k, \boldsymbol{y}_k | \check{\boldsymbol{x}}_0, \boldsymbol{v}_{1:k}, \boldsymbol{y}_{0:k-1}) \approx \mathcal{N}\left(\begin{bmatrix} \boldsymbol{\mu}_{x,k} \\ \boldsymbol{\mu}_{y,k} \end{bmatrix}, \begin{bmatrix} \boldsymbol{\Sigma}_{xx,k} & \boldsymbol{\Sigma}_{xy,k} \\ \boldsymbol{\Sigma}_{yx,k} & \boldsymbol{\Sigma}_{yy,k} \end{bmatrix}\right)$$

$$= \mathcal{N}\left(\begin{bmatrix} \check{\boldsymbol{x}}_k \\ \boldsymbol{y}_{\mathrm{op},k} + \boldsymbol{G}_k(\check{\boldsymbol{x}}_k - \boldsymbol{x}_{\mathrm{op},k}) \end{bmatrix}, \begin{bmatrix} \check{\boldsymbol{P}}_k & \check{\boldsymbol{P}}_k \boldsymbol{G}_k^{\mathrm{T}} \\ \boldsymbol{G}_k \check{\boldsymbol{P}}_k & \boldsymbol{G}_k \check{\boldsymbol{P}}_k \boldsymbol{G}_k^{\mathrm{T}} + \boldsymbol{R}'_k \end{bmatrix}\right)$$

  - 利用高斯推断，写出条件概率密度：

$$p(\boldsymbol{x}_k | \check{\boldsymbol{x}}_0, \boldsymbol{v}_{1:k}, \boldsymbol{y}_{0:k})$$

$$= \mathcal{N}\Big(\underbrace{\boldsymbol{\mu}_{x,k} + \boldsymbol{\Sigma}_{xy,k}\boldsymbol{\Sigma}_{yy,k}^{-1}(\boldsymbol{y}_k - \boldsymbol{\mu}_{y,k})}_{\hat{\boldsymbol{x}}_k}, \underbrace{\boldsymbol{\Sigma}_{xx,k} - \boldsymbol{\Sigma}_{xy,k}\boldsymbol{\Sigma}_{yy,k}^{-1}\boldsymbol{\Sigma}_{yx,k}}_{\hat{\boldsymbol{P}}_k}\Big)$$

离散时间的递归估计

- 定义卡尔曼增益，然后写出均值与协方差的更新式：

$$K_k = \Sigma_{xy,k}\Sigma_{yy,k}^{-1}$$
$$\hat{P}_k = \check{P}_k - K_k\Sigma_{xy,k}^{\mathrm{T}}$$
$$\hat{x}_k = \check{x}_k + K_k\left(y_k - \mu_{y,k}\right)$$

- 代入非线性模型：

$$K_k = \check{P}_k G_k^{\mathrm{T}}\left(G_k\check{P}_k G_k^{\mathrm{T}} + R_k'\right)^{-1}$$
$$\hat{P}_k = \left(1 - K_k G_k\right)\check{P}_k$$
$$\hat{x}_k = \check{x}_k + K_k\left(y_k - y_{\mathrm{op},k} - G_k(\check{x}_k - x_{\mathrm{op},k})\right)$$

- 该结论和EKF是一致的
- 但是这里显式定义了线性化的工作点 $x_{\mathrm{op},k}$ ，而在EKF中取

$$x_{\mathrm{op},k} = \check{x}_k$$

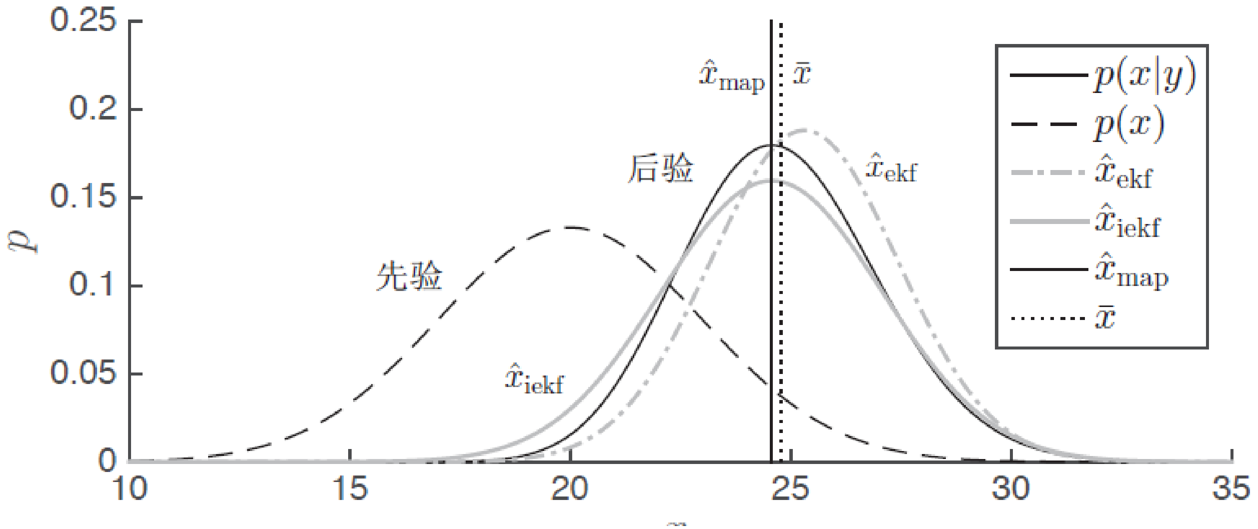
- 事实上我们可以不断迭代，然后把上一次迭代结果作为下一次的线性化点：

$$x_{\mathrm{op},k} \leftarrow \hat{x}_k$$

就可以得到更好的结果。这种滤波器称为迭代的EKF（IEKF）

离散时间的递归估计

- EKF、IEKF、MAP之间的关系
  - IEKF对应后验概率的极大值，事实上是一个估计；
  - 而EKF只处理一次线性化，相当于只迭代一步，很难说这次迭代能走到什么程度；
  - 在本课开头的双目例子中，可以得到如下仿真结果：

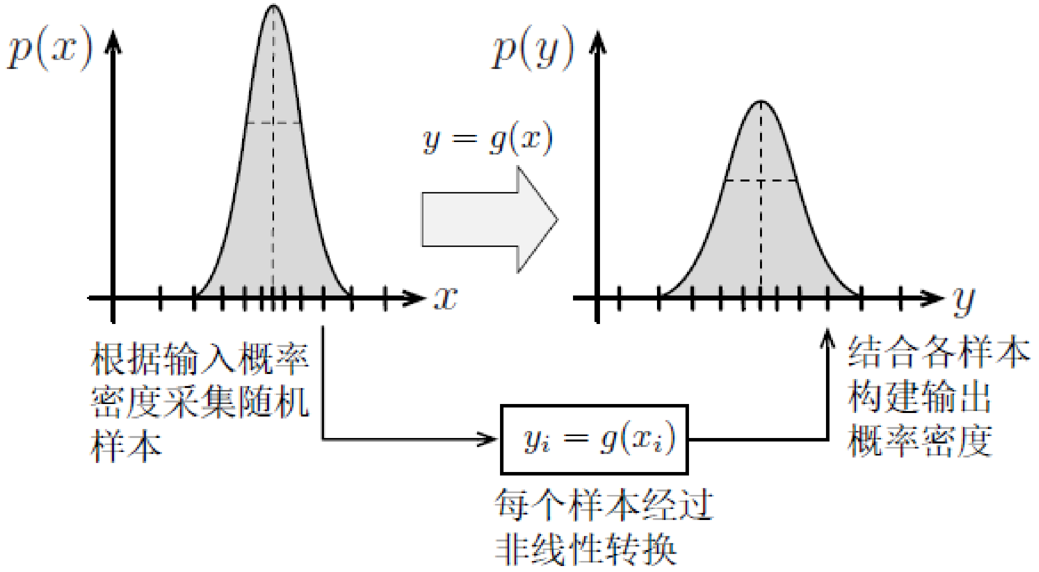


实验结果：

- ☐ IEKF的均值是一个MAP解
- ☐ 全贝叶斯估计p(x|y)、MAP解与真实x之间是有偏的
- ☐ IEKF和MAP解优于EKF

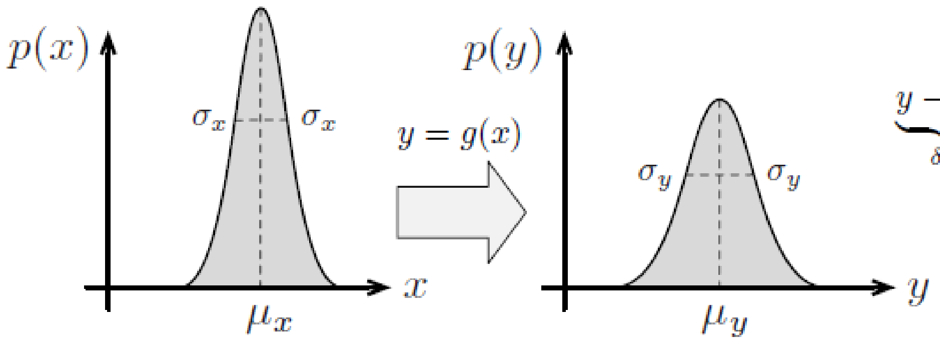
离散时间的递归估计

- 处理非线性函数的其他方法
  - 前面主要介绍了工作点处线性化方法处理非线性函数
  - 实际中还其他的手段，例如蒙特卡罗法、sigma point 变换法等等
- 蒙特卡罗方法（Monte Carlo Method）
  - 在非线性变换中，对输入分布大量采样，计算非线性变换后的值，再用变换后的值构建输出分布
  - 大数定律保证了当样本数量接近无穷大时，不管什么分布和什么变换，都能得到正确的值
  - 计算量随着维度指数增长，不过采样数量可以人为控制；采样数量很大时，是最准确的方法
  - 我们在双目例子中就使用了蒙特卡罗法来衡量不同估计器的结果



离散时间的递归估计

- 线性化方法
  - 线性化方法就是我们前面介绍EKF时的方法
  - 具体来说，就是以x均值点为工作点，进行线性化后得到y的分布



$$
\begin{aligned}
\mu_y &= g(\mu_x) \\
\underbrace{y - \mu_y}_{\delta y} &\approx \underbrace{\left.\frac{\partial g(x)}{\partial x}\right|_{x=\mu_x}}_{a} \underbrace{x - \mu_x}_{\delta x} \\
\sigma_y^2 &= E[\delta y^2] \\
&= a^2 E[\delta x^2] \\
&= a^2 \sigma_x^2
\end{aligned}
$$

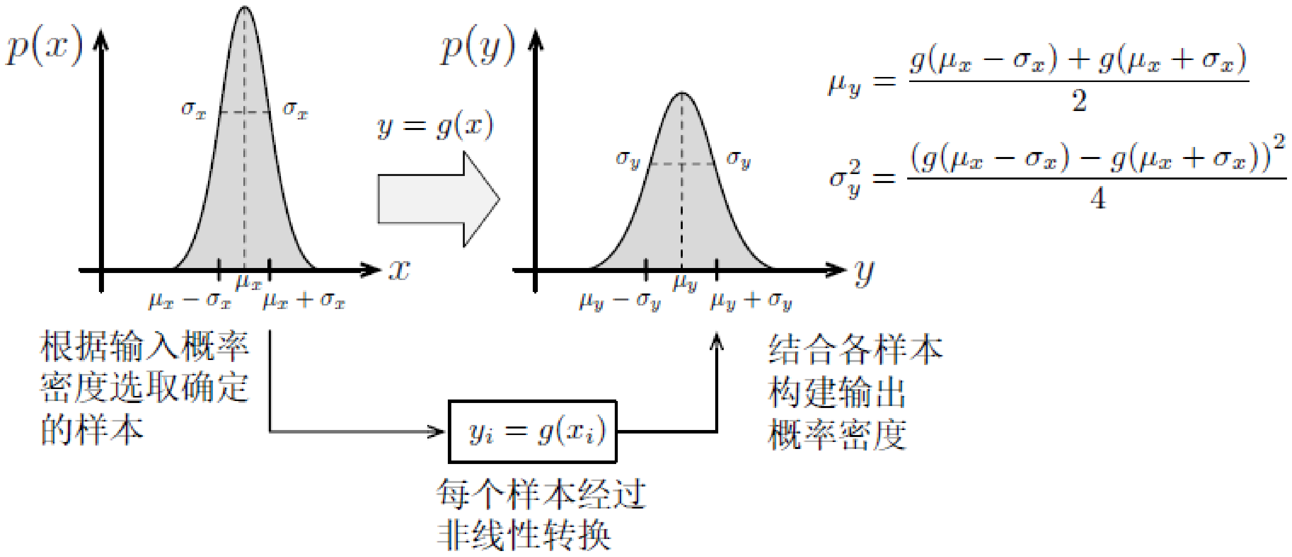
这个实际上是非常不准确的，因为：

1. 高斯分布的非线性变换本身不再是高斯分布，我们不知道它是什么分布，只保留了那个分布的一、二阶矩
2. 非线性变换的线性化本身存在误差
3. 线性化点理论上是x均值，但实际当中是x均值的估计值，它也可能是个错误的值
4. y均值=g(x均值）是成问题的

- ☐ 如果函数是近似线性的，那么线性化效果不错
- ☐ 但如果函数是高度非线性的，那么很难说单次线性化会怎么样

离散时间的递归估计

- Sigma Point变换（SP变换，或无迹变换）
  - 是线性化方法和蒙特卡尔方法的折中
  - 核心思想：选定输入分布的几个点（Sigma Point），计算这几个点的非线性变换，用它的结果构建输出分布
  - 最简单的一维SP如下所示：



离散时间的递归估计

- 高维（L维）高斯分布的sigma point（共2L+1个）：

$$\boldsymbol{L}\boldsymbol{L}^{\mathrm{T}} = \boldsymbol{\Sigma}_{xx}\text{,（Cholesky 分解，}\boldsymbol{L}\text{ 为下三角矩阵）}$$

$$\begin{aligned} \boldsymbol{x}_0 &= \boldsymbol{\mu}_x \\ \boldsymbol{x}_i &= \boldsymbol{\mu}_x + \sqrt{L+\kappa}\,\mathrm{col}_i \boldsymbol{L} \\ \boldsymbol{x}_{i+L} &= \boldsymbol{\mu}_x - \sqrt{L+\kappa}\,\mathrm{col}_i \boldsymbol{L} \end{aligned} \qquad i = 1, \cdots, L$$

- 其中κ是一个自定义常数，
- col表示取L的第i列

- 这些样本点满足：

$$\boldsymbol{\mu}_x = \sum_{i=0}^{2L} \alpha_i \boldsymbol{x}_i$$

$$\boldsymbol{\Sigma}_{xx} = \sum_{i=0}^{2L} \alpha_i (\boldsymbol{x}_i - \boldsymbol{\mu}_x)(\boldsymbol{x}_i - \boldsymbol{\mu}_x)^{\mathrm{T}}$$

其中：

$$\alpha_i = \begin{cases} \frac{\kappa}{L+\kappa} & i = 0 \\ \frac{1}{2}\frac{1}{L+\kappa} & \text{其他} \end{cases}$$

- 习题：请验证此式

离散时间的递归估计

- 对Sigma Point进行非线性变换，得到：$y_i = g(x_i), \quad i = 0, \cdots, 2L$
- 用y的结果构建输出高斯分布：

$$\mu_y = \sum_{i=0}^{2L} \alpha_i y_i \qquad \Sigma_{yy} = \sum_{i=0}^{2L} \alpha_i (y_i - \mu_y)(y_i - \mu_y)^{\mathrm{T}}$$

- Sigma Point的好处：
  - 不用计算线性化雅可比矩阵；
  - 仅使用了标准矩阵加法乘法和Cholesky分解；
  - 对非线性函数的要求很少（不要求光滑和可微）

离散时间的递归估计

- 下面通过例子来比较各方法的优劣
- 输入：$\mathcal{N}(\mu_x, \sigma_x^2)$ ，非线性变换：$f(x) = x^2$
- 蒙特卡罗方法（闭式的）：
  - 采样点： $x_i = \mu_x + \delta x_i, \quad \delta x_i \leftarrow \mathcal{N}(0, \sigma_x^2)$
  - 非线性变换： $y_i = f(x_i) = f(\mu_x + \delta x_i) = (\mu_x + \delta x_i)^2 = \mu_x^2 + 2\mu_x \delta x_i + \delta x_i^2$
  - 它的均值和方差： $\mu_y = E[y_i] = \mu_x^2 + 2\mu_x \underbrace{E[\delta x_i]}_{0} + \underbrace{E[\delta x_i^2]}_{\sigma_x^2} = \mu_x^2 + \sigma_x^2$

$$\begin{aligned}
\sigma_y^2 &= E[(y_i - \mu_y)^2] \\
&= E[(2\mu_x \delta x_i + \delta x_i^2 - \sigma_x^2)^2] \\
&= \underbrace{E[\delta x_i^4]}_{3\sigma_x^4} + 4\mu_x \underbrace{E[\delta x_i^3]}_{0} + (4\mu_x^2 - 2\sigma_x^2)\underbrace{E[\delta x_i^2]}_{\sigma_x^2} - 4\mu_x \sigma_x^2 \underbrace{E[\delta x_i]}_{0} + \sigma_x^4 \\
&= 4\mu_x^2 \sigma_x^2 + 2\sigma_x^4
\end{aligned}$$

- 众所周知，输出分布的实际学名是卡方分布
- 如果近似为高斯分布，那么仅取其一二阶矩即可（图像上来看其实并不像）
- 高斯分布的三、四阶矩可以当作结论来记，如果想自己推的话，参考教材第2章Isserlis定理（本课程略去）

离散时间的递归估计

- 线性化方法： $$y_i = f(\mu_x + \delta x_i) \approx \underbrace{f(\mu_x)}_{\mu_x^2} + \underbrace{\left.\frac{\partial f}{\partial x}\right|_{\mu_x}}_{2\mu_x} \delta x_i = \mu_x^2 + 2\mu_x \delta x_i$$
- 线性化后的期望与方差：

$$\mu_y = E[y_i] = \mu_x^2 + 2\mu_x \underbrace{E[\delta x_i]}_{0} = \mu_x^2 \qquad \sigma_y^2 = E[(y_i - \mu_y)^2] = E[(2\mu_x \delta x_i)^2] = 4\mu_x^2 \sigma_x^2$$

- 对比上一页蒙特卡罗法，可见均值是有偏的，而且方差过小了（Overconfident）

离散时间的递归估计

- Sigma point变换：
  - 选取3个sigma point： $x_0 = \mu_x, \quad x_1 = \mu_x + \sqrt{1+\kappa}\sigma_x, \quad x_2 = \mu_x - \sqrt{1+\kappa}\sigma_x$
  - 代入非线性变换：

$$y_0 = f(x_0) = \mu_x^2$$

$$y_1 = f(x_1) = \left(\mu_x + \sqrt{1+\kappa}\sigma_x\right)^2 = \mu_x^2 + 2\mu_x\sqrt{1+\kappa}\sigma_x + (1+\kappa)\sigma_x^2$$

$$y_2 = f(x_2) = \left(\mu_x - \sqrt{1+\kappa}\sigma_x\right)^2 = \mu_x^2 - 2\mu_x\sqrt{1+\kappa}\sigma_x + (1+\kappa)\sigma_x^2$$

  - 输出分布：

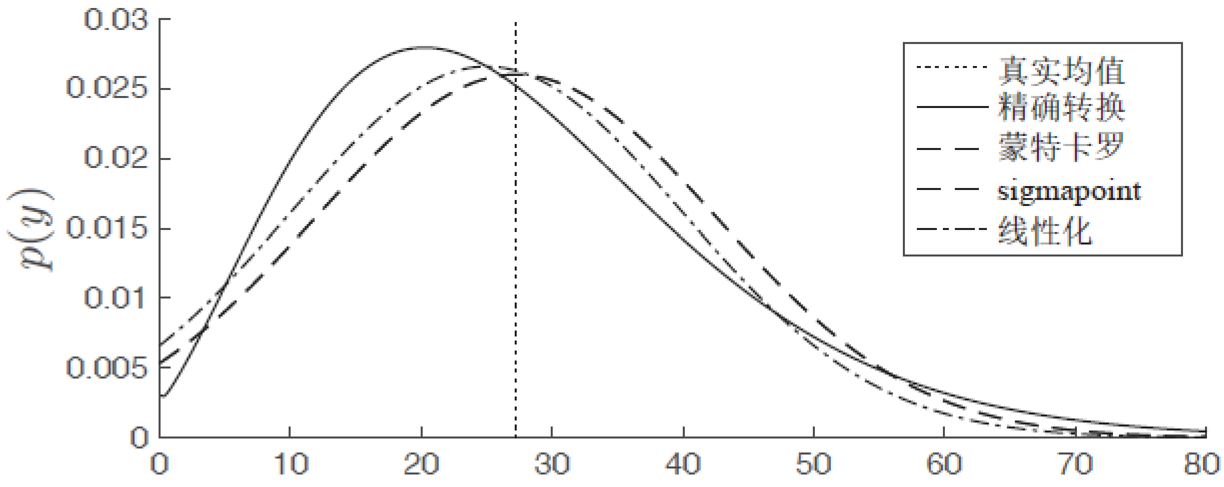
$$\begin{aligned}\mu_y &= \frac{1}{1+\kappa}\left(\kappa y_0 + \frac{1}{2}\sum_{i=1}^{2} y_i\right)\\ &= \frac{1}{1+\kappa}\left(\kappa\mu_x^2 + \frac{1}{2}\left(\mu_x^2 + 2\mu_x\sqrt{1+\kappa}\sigma_x + (1+\kappa)\sigma_x^2 + \mu_x^2\right.\right.\\ &\qquad \left.\left. -2\mu_x\sqrt{1+\kappa}\sigma_x + (1+\kappa)\sigma_x^2\right)\right)\\ &= \frac{1}{1+\kappa}\left(\kappa\mu_x^2 + \mu_x^2 + (1+\kappa)\sigma_x^2\right)\\ &= \mu_x^2 + \sigma_x^2\end{aligned}$$

$$\begin{aligned}\sigma_y^2 &= \frac{1}{1+\kappa}\left(\kappa(y_0-\mu_y)^2 + \frac{1}{2}\sum_{i=1}^{2}(y_i-\mu_y)^2\right)\\ &= \frac{1}{1+\kappa}\left(\kappa\sigma_x^4 + \frac{1}{2}\left(\left(2\mu_x\sqrt{1+\kappa}\sigma_x + \kappa\sigma_x^2\right)^2\right.\right.\\ &\qquad \left.\left. + \left(-2\mu_x\sqrt{1+\kappa}\sigma_x + \kappa\sigma_x^2\right)^2\right)\right)\\ &= \frac{1}{1+\kappa}\left(\kappa\sigma_x^4 + 4(1+\kappa)\mu_x^2\sigma_x^2 + \kappa^2\sigma_x^4\right)\\ &= 4\mu_x^2\sigma_x^2 + \kappa\sigma_x^4\end{aligned}$$

□ 取κ=2时，得到真实的均值和方差

离散时间的递归估计

- 三种方法的结果对比：



- ☐ 精确变换并不是高斯的
- ☐ 蒙特卡罗（闭式）可看成最优近似
- ☐ Sigma-point在κ=2时与蒙特卡罗法相同
- ☐ 线性化的均值和方差都有少许差异，均值有偏，方差较小（更瘦）

离散时间的递归估计

- 粒子滤波
  - 粒子滤波是唯一一种可以处理NLNG系统的实用技术
  - 而且特别简单
- 粒子滤波算法流程

1. 采样：从先验与运动噪声中采样M个样本：

$$\begin{bmatrix} \hat{x}_{k-1,m} \\ w_{k,m} \end{bmatrix} \leftarrow p(x_{k-1}|\check{x}_0, v_{1:k-1}, y_{1:k-1})p(w_k)$$

2. 使用运动方程得到预测分布：

$$\check{x}_{k,m} = f(\hat{x}_{k-1,m}, v_k, w_{k,m})$$

3. 使用观测方程进行校正

(a) 为每个粒子计算权重：

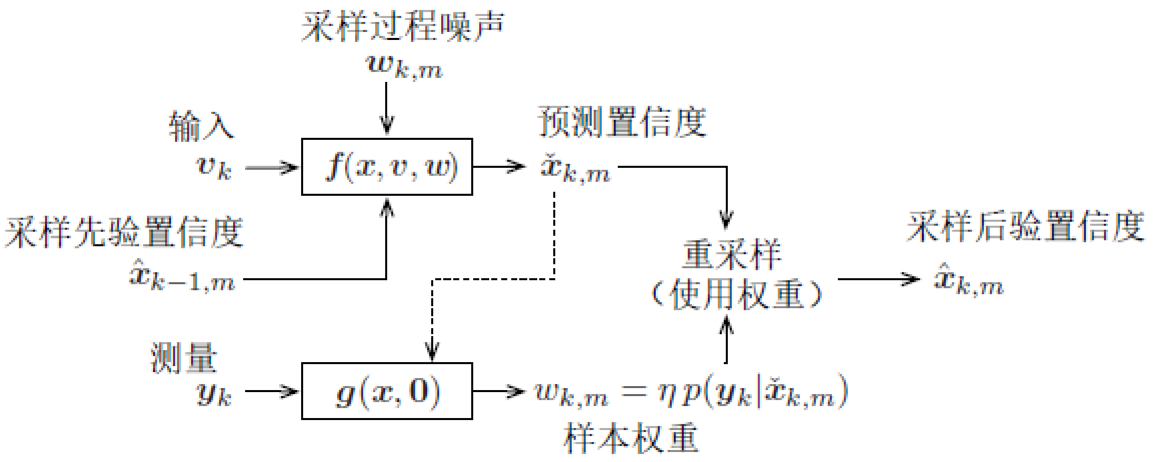
$$w_{k,m} = \frac{p(\check{x}_{k,m}|\check{x}_0, v_{1:k}, y_{1:k})}{p(\check{x}_{k,m}|\check{x}_0, v_{1:k}, y_{1:k-1})} = \eta p(y_k|\check{x}_{k,m}) \qquad \check{y}_{k,m} = g(\check{x}_{k,m}, 0)$$

(b) 对粒子进行重要性重采样（Sample importance resampling），具体方法见下一页

$$\hat{x}_{k,m} \xleftarrow{\text{重要性重采样}} \{\check{x}_{k,m}, w_{k,m}\}$$

离散时间的递归估计

- 粒子滤波的算法流程图示
- 注解：
  - 需要粒子数量和问题维度有关（三自由度定位使用几百个粒子即可），随维度指数增长
  - 粒子数量也可动态设置
  - 重采样可以每隔一段时间做一次



Madow的重采样算法[36]（轮盘赌）：

- 设粒子具有归一化的权重： $w_m \in \mathbb{R} > 0$，那么按照权重建立[0,1]上的小区间： $\beta_m = \frac{\sum_{n=1}^{m} w_n}{\sum_{\ell=1}^{M} w_\ell}$
- 在[0,1)上取均匀分布随机数，把区间内的样本作为重采样样本；

[36] Madow W G. On the Theory of Systematic Sampling, II. The Annals of Mathematical Statistics, 1949, 30:333–354.

离散时间的递归估计

- SPKF(UKF) 无迹卡尔曼滤波
  - 将sigma point变换应用到卡尔曼的运动和观测模型中
  - 预测部分：先把状态和运动噪声放在一起写成联合形式，设其维度为L：

$$\boldsymbol{\mu}_z = \begin{bmatrix} \hat{\boldsymbol{x}}_{k-1} \\ \mathbf{0} \end{bmatrix}, \quad \boldsymbol{\Sigma}_{zz} = \begin{bmatrix} \hat{\boldsymbol{P}}_{k-1} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{Q}_k \end{bmatrix}$$

  - 取Sigma point，然后代入运动模型：

$$\boldsymbol{L}\boldsymbol{L}^{\mathrm{T}} = \boldsymbol{\Sigma}_{zz}, \text{（Cholesky 分解，} \boldsymbol{L} \text{ 为下三角矩阵）}$$

$$\begin{aligned} \boldsymbol{z}_0 &= \boldsymbol{\mu}_z \\ \boldsymbol{z}_i &= \boldsymbol{\mu}_z + \sqrt{L+\kappa}\operatorname{col}_i \boldsymbol{L} \\ \boldsymbol{z}_{i+L} &= \boldsymbol{\mu}_z - \sqrt{L+\kappa}\operatorname{col}_i \boldsymbol{L} \end{aligned} \qquad i = 1, \cdots, L$$

其中：$\boldsymbol{z}_i = \begin{bmatrix} \hat{\boldsymbol{x}}_{k-1,i} \\ \boldsymbol{w}_{k,i} \end{bmatrix}$

代入运动模型后：$\check{\boldsymbol{x}}_{k,i} = \boldsymbol{f}(\hat{\boldsymbol{x}}_{k-1,i}, \boldsymbol{v}_k, \boldsymbol{w}_{k,i}), \quad i = 0, \cdots, 2L$

得到预测的先验：

$$\check{\boldsymbol{x}}_k = \sum_{i=0}^{2L} \alpha_i \check{\boldsymbol{x}}_{k,i}$$

$$\check{\boldsymbol{P}}_k = \sum_{i=0}^{2L} \alpha_i (\check{\boldsymbol{x}}_{k,i} - \check{\boldsymbol{x}}_k)(\check{\boldsymbol{x}}_{k,i} - \check{\boldsymbol{x}}_k)^{\mathrm{T}}$$

离散时间的递归估计

- SPKF校正步骤：
  1. 把预测和观测噪声写在一起：$\mu_z = \begin{bmatrix} \check{x}_k \\ 0 \end{bmatrix}, \quad \Sigma_{zz} = \begin{bmatrix} \check{P}_k & 0 \\ 0 & R_k \end{bmatrix}$
  2. Sigma Point变换：

     $LL^{\mathrm{T}} = \Sigma_{zz}$，（Cholesky 分解，$L$ 为下三角矩阵）

     $$\begin{aligned} z_0 &= \mu_z \\ z_i &= \mu_z + \sqrt{L+\kappa}\operatorname{col}_i L \\ z_{i+L} &= \mu_z - \sqrt{L+\kappa}\operatorname{col}_i L \end{aligned} \quad i = 1, \cdots, L \qquad z_i = \begin{bmatrix} \check{x}_{k,i} \\ n_{k,i} \end{bmatrix}$$

  3. 构建后验分布：

     $$\mu_{y,k} = \sum_{i=0}^{2L} \alpha_i \check{y}_{k,i}$$

     $$\Sigma_{yy,k} = \sum_{i=0}^{2L} \alpha_i (\check{y}_{k,i} - \mu_{y,k})(\check{y}_{k,i} - \mu_{y,k})^{\mathrm{T}}$$

     $$\Sigma_{xy,k} = \sum_{i=0}^{2L} \alpha_i (\check{x}_{k,i} - \check{x}_k)(\check{y}_{k,i} - \mu_{y,k})^{\mathrm{T}}$$

代入观测方程：

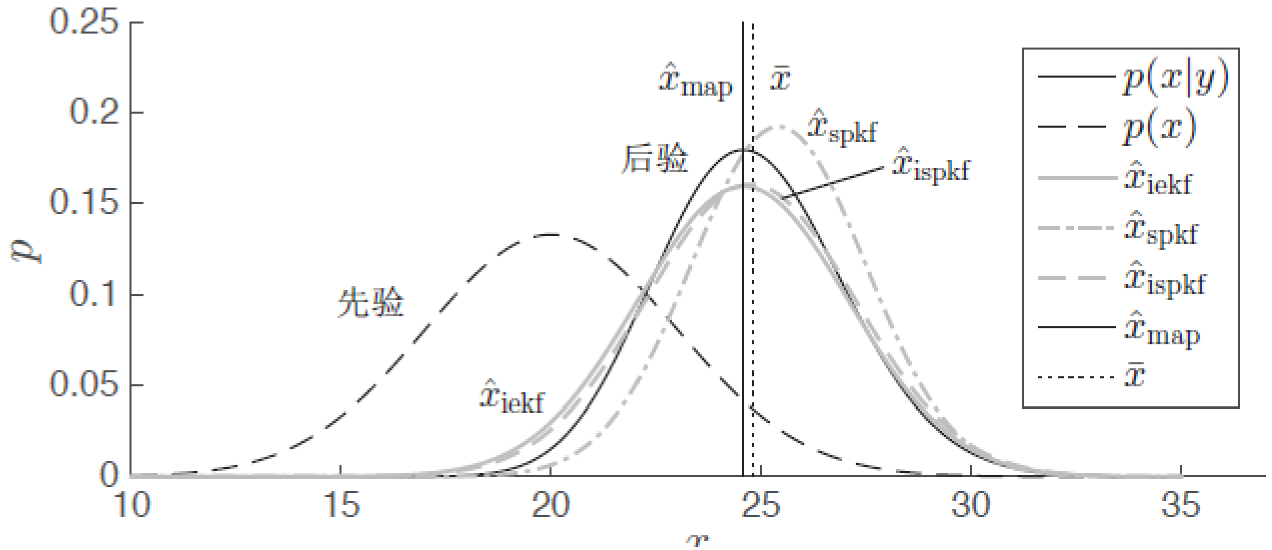
$$\check{y}_{k,i} = g(\check{x}_{k,i}, n_{k,i}), \quad i = 0, \cdots, 2L$$

SPKF的优势：

- ☐ 完全不需要求导
- ☐ 甚至不需要运动和观测方程的解析形式，视为黑盒模型
- ☐ 如同IEKF之于EKF一样，SPKF也可以有迭代版本ISPKF，本课程略去

离散时间的递归估计

- SPKF与MAP
  - 我们在之前的双目视差问题里验证SPKF的解与其他解的问题
  - SPKF中取kappa=2



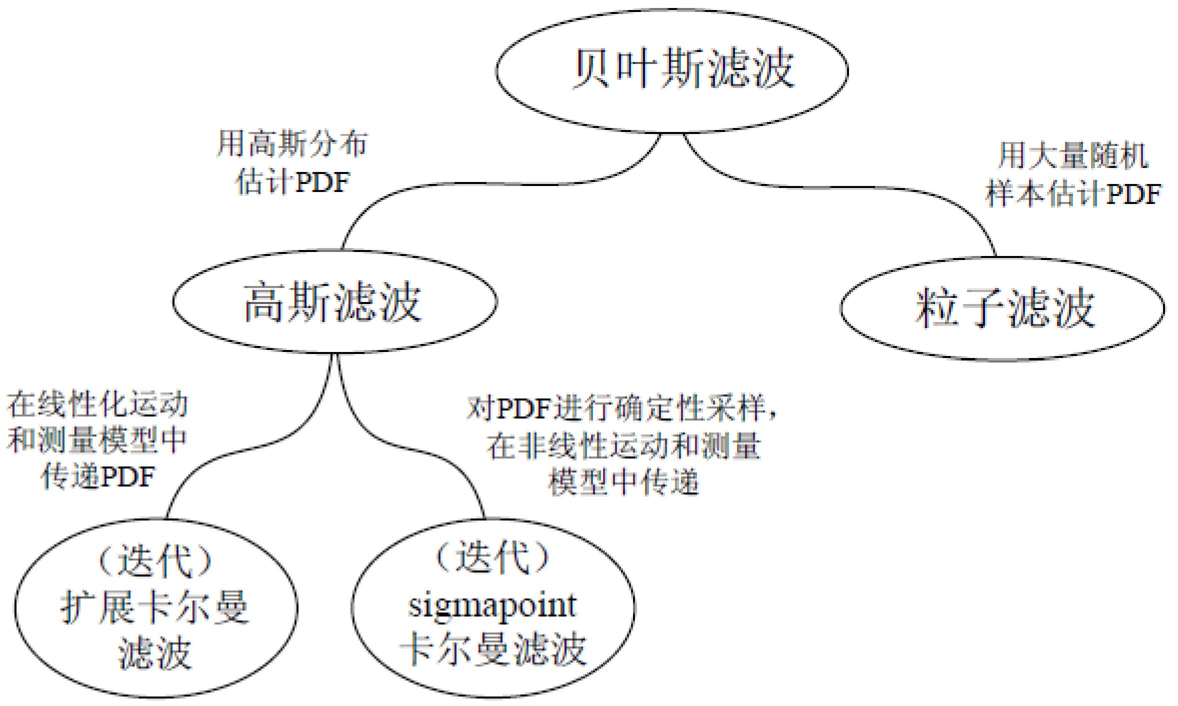
结论：

- ☐ 单次SPKF和EKF差不多，与MAP无明显关系
- ☐ 迭代的ISPKF和真实均值更接近；

本实验中可以合理地认为ISPKF收敛于真实后验均值；而MAP解收敛于模

离散时间的递归估计

- 本小节几种滤波器之间的关系



第3讲 非线性高斯系统的状态估计问题

- 引言
- 离散时间的递归估计
- 离散时间的批量估计
- 连续时间的批量估计

离散时间的批量估计

- 我们已经介绍了许多递归形式的滤波器，但是它们都蕴含了马尔可夫假设
- 现在我们来考察批量形式下的情况
- 从上一讲内容中，我们已经知道批量解可以等价于最小二乘问题
- 定义优化变量、优化目标函数为：

$$x = \begin{bmatrix} x_0 \\ x_1 \\ \vdots \\ x_K \end{bmatrix}$$

优化变量

$$e_{v,k}(x) = \begin{cases} \check{x}_0 - x_0 & k = 0 \\ f(x_{k-1}, v_k, 0) - x_k & k = 1, \cdots, K \end{cases}$$

$$e_{y,k}(x) = y_k - g(x_k, 0), \quad k = 0, \cdots, K$$

运动与观测的误差

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

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

误差的马氏范数

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

整体优化目标

离散时间的批量估计

- 整理其矩阵形式：

$$
\boldsymbol{e}(\boldsymbol{x})=\begin{bmatrix}\boldsymbol{e}_v(\boldsymbol{x})\\ \boldsymbol{e}_y(\boldsymbol{x})\end{bmatrix},\quad \boldsymbol{e}_v(\boldsymbol{x})=\begin{bmatrix}\boldsymbol{e}_{v,0}(\boldsymbol{x})\\ \vdots\\ \boldsymbol{e}_{v,K}(\boldsymbol{x})\end{bmatrix},\quad \boldsymbol{e}_y(\boldsymbol{x})=\begin{bmatrix}\boldsymbol{e}_{y,0}(\boldsymbol{x})\\ \vdots\\ \boldsymbol{e}_{y,K}(\boldsymbol{x})\end{bmatrix}
$$

$$
\boldsymbol{W}=\operatorname{diag}(\boldsymbol{W}_v,\boldsymbol{W}_y)
$$

$$
\boldsymbol{W}_v=\operatorname{diag}(\boldsymbol{W}_{v,0},\cdots,\boldsymbol{W}_{v,K}),\quad \boldsymbol{W}_y=\operatorname{diag}(\boldsymbol{W}_{y,0},\cdots,\boldsymbol{W}_{y,K})
$$

- 那么目标函数简单写为： $J(\boldsymbol{x})=\frac{1}{2}\boldsymbol{e}(\boldsymbol{x})^{\mathrm{T}}\boldsymbol{W}^{-1}\boldsymbol{e}(\boldsymbol{x})$
- 这是一个无约束最小二乘问题，可以通过通用的牛顿法、高斯牛顿法来求解

离散时间的批量估计

- 牛顿法
  - 不断迭代并线性化，向着最优解前进；线性化：

$$J(\boldsymbol{x}_{\mathrm{op}}+\delta\boldsymbol{x}) \approx J(\boldsymbol{x}_{\mathrm{op}}) + \underbrace{\left(\left.\frac{\partial J(\boldsymbol{x})}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}_{\mathrm{op}}}\right)}_{\text{雅可比}}\delta\boldsymbol{x} + \frac{1}{2}\delta\boldsymbol{x}^{\mathrm{T}}\underbrace{\left(\left.\frac{\partial^2 J(\boldsymbol{x})}{\partial \boldsymbol{x}\,\partial \boldsymbol{x}^{\mathrm{T}}}\right|_{\boldsymbol{x}_{\mathrm{op}}}\right)}_{\text{海塞}}\delta\boldsymbol{x}$$

  - 令线性化函数对delta x导数为零，来求得最优delta x：

$$\frac{\partial J(\boldsymbol{x}_{\mathrm{op}}+\delta\boldsymbol{x})}{\partial\,\delta\boldsymbol{x}} = \left(\left.\frac{\partial J(\boldsymbol{x})}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}_{\mathrm{op}}}\right) + \delta\boldsymbol{x}^{*\mathrm{T}}\left(\left.\frac{\partial^2 J(\boldsymbol{x})}{\partial \boldsymbol{x}\,\partial \boldsymbol{x}^{\mathrm{T}}}\right|_{\boldsymbol{x}_{\mathrm{op}}}\right) = \boldsymbol{0}$$

$$\Rightarrow \left(\left.\frac{\partial^2 J(\boldsymbol{x})}{\partial \boldsymbol{x}\,\partial \boldsymbol{x}^{\mathrm{T}}}\right|_{\boldsymbol{x}_{\mathrm{op}}}\right)\delta\boldsymbol{x}^* = -\left(\left.\frac{\partial J(\boldsymbol{x})}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}_{\mathrm{op}}}\right)^{\mathrm{T}}$$

  - 然后不断迭代： $\boldsymbol{x}_{\mathrm{op}} \leftarrow \boldsymbol{x}_{\mathrm{op}} + \delta\boldsymbol{x}^*$

离散时间的批量估计

- Gauss-Newton法与Levernberg-Marquardt方法
  - 基于牛顿法的改进，更为实用
  - 由于在VSLAM课程和VIO课程里均已详细介绍过，且有编程实现，此处省略；
  - 不懂的同学可以参考这两个课程的课件或者状态估计教材

离散时间的批量估计

- 回到批量估计的问题定义处：
  - 批量问题的误差定义与线性化：

$$e_{v,k}(\boldsymbol{x}_{\text{op}} + \delta\boldsymbol{x}) \approx \begin{cases} e_{v,0}(\boldsymbol{x}_{\text{op}}) - \delta\boldsymbol{x}_0, & k = 0 \\ e_{v,k}(\boldsymbol{x}_{\text{op}}) + \boldsymbol{F}_{k-1}\delta\boldsymbol{x}_{k-1} - \delta\boldsymbol{x}_k, & k = 1, \cdots, K \end{cases}$$

$$e_{y,k}(\boldsymbol{x}_{\text{op}} + \delta\boldsymbol{x}) \approx e_{y,k}(\boldsymbol{x}_{\text{op}}) - \boldsymbol{G}_k\delta\boldsymbol{x}_k, \quad k = 0, \cdots, K$$

雅可比矩阵：

$$\boldsymbol{F}_{k-1} = \left.\frac{\partial \boldsymbol{f}(\boldsymbol{x}_{k-1}, \boldsymbol{v}_k, \boldsymbol{w}_k)}{\partial \boldsymbol{x}_{k-1}}\right|_{\boldsymbol{x}_{\text{op},k-1}, \boldsymbol{v}_k, \boldsymbol{0}}, \quad \boldsymbol{G}_k = \left.\frac{\partial \boldsymbol{g}(\boldsymbol{x}_k, \boldsymbol{n}_k)}{\partial \boldsymbol{x}_k}\right|_{\boldsymbol{x}_{\text{op},k}, \boldsymbol{0}}$$

  - 定义噪声为： $\boldsymbol{W}_{v,k} = \boldsymbol{Q}'_k, \quad \boldsymbol{W}_{y,k} = \boldsymbol{R}'_k$
  - 那么整个问题可写成矩阵形式：

$$\delta\boldsymbol{x} = \begin{bmatrix} \delta\boldsymbol{x}_0 \\ \delta\boldsymbol{x}_1 \\ \delta\boldsymbol{x}_2 \\ \vdots \\ \delta\boldsymbol{x}_K \end{bmatrix}, \quad \boldsymbol{H} = \left[\begin{array}{ccccc} 1 & & & & \\ -\boldsymbol{F}_0 & 1 & & & \\ & -\boldsymbol{F}_1 & \ddots & & \\ & & \ddots & 1 & \\ & & & -\boldsymbol{F}_{K-1} & 1 \\ \hline \boldsymbol{G}_0 & & & & \\ & \boldsymbol{G}_1 & & & \\ & & \boldsymbol{G}_2 & & \\ & & & \ddots & \\ & & & & \boldsymbol{G}_K \end{array}\right]$$

$$\boldsymbol{e}(\boldsymbol{x}_{\text{op}}) = \left[\begin{array}{c} e_{v,0}(\boldsymbol{x}_{\text{op}}) \\ e_{v,1}(\boldsymbol{x}_{\text{op}}) \\ \vdots \\ e_{v,K}(\boldsymbol{x}_{\text{op}}) \\ \hline e_{y,0}(\boldsymbol{x}_{\text{op}}) \\ e_{y,1}(\boldsymbol{x}_{\text{op}}) \\ \vdots \\ e_{y,K}(\boldsymbol{x}_{\text{op}}) \end{array}\right]$$

$$\boldsymbol{W} = \operatorname{diag}(\check{\boldsymbol{P}}_0, \boldsymbol{Q}'_1, \cdots, \boldsymbol{Q}'_K, \boldsymbol{R}'_0, \boldsymbol{R}'_1, \cdots, \boldsymbol{R}'_K)$$

Gauss-Newton的线性方程：

$$\underbrace{(\boldsymbol{H}^{\mathrm{T}}\boldsymbol{W}^{-1}\boldsymbol{H})}_{\text{三对角块}}\delta\boldsymbol{x}^* = \boldsymbol{H}^{\mathrm{T}}\boldsymbol{W}^{-1}\boldsymbol{e}(\boldsymbol{x}_{\text{op}})$$

于是得到批量MAP的解法

离散时间的批量估计

- 贝叶斯推断
  - 假设整条轨迹有初值 x_{op} 我们在这个点上线性化运动模型：

$$x_k \approx f(x_{\text{op},k-1}, v_k, 0) + F_{k-1}(x_{k-1} - x_{\text{op},k-1}) + w'_k$$

  - 通过一些数学手段，写成提升形式：

$$F = \begin{bmatrix} 1 & & & & & \\ F_0 & 1 & & & & \\ F_1 F_0 & F_1 & 1 & & & \\ \vdots & \vdots & \vdots & \ddots & & \\ F_{K-2} \cdots F_0 & F_{K-2} \cdots F_1 & F_{K-2} \cdots F_2 & \cdots & 1 & \\ F_{K-1} \cdots F_0 & F_{K-1} \cdots F_1 & F_{K-1} \cdots F_2 & \cdots & F_{K-1} & 1 \end{bmatrix} \quad \nu = \begin{bmatrix} \check{x}_0 \\ f(x_{\text{op},0}, v_1, 0) - F_0 x_{\text{op},0} \\ f(x_{\text{op},1}, v_2, 0) - F_1 x_{\text{op},1} \\ \vdots \\ f(x_{\text{op},K-1}, v_K, 0) - F_{K-1} x_{\text{op},K-1} \end{bmatrix}$$

  - 运动方程的提升形式为： $x = F(\nu + w')$

离散时间的批量估计

- 那么，在初值已知的情况下，先验的均值和协方差为：

$$\check{x} = E[x] = E[F(\nu + w')] = F\nu \qquad \check{P} = E\left[(x - E[x])(x - E[x])^{\mathrm{T}}\right] = FE\left[w'w'^{\mathrm{T}}\right]F^{\mathrm{T}} = FQ'F^{\mathrm{T}}$$

- 于是先验记为： $x \sim \mathcal{N}(F\nu, FQ'F^{\mathrm{T}})$ 其中 $Q' = \mathrm{diag}(\check{P}_0, Q'_1, Q'_2, \cdots, Q'_K)$
- 观测模型： $y_k \approx g(x_{\mathrm{op},k}, 0) + G_k(x_{k-1} - x_{\mathrm{op},k-1}) + n'_k$ 提升形式： $y = y_{\mathrm{op}} + G(x - x_{\mathrm{op}}) + n'$
- 那么联合分布：

$$p(x, y|v) = \mathcal{N}\left(\begin{bmatrix} \check{x} \\ y_{\mathrm{op}} + G(\check{x} - x_{\mathrm{op}}) \end{bmatrix}, \begin{bmatrix} \check{P} & \check{P}G^{\mathrm{T}} \\ G\check{P} & G\check{P}G^{\mathrm{T}} + R' \end{bmatrix}\right)$$

$$y_{\mathrm{op}} = \begin{bmatrix} g(x_{\mathrm{op},0}, 0) \\ g(x_{\mathrm{op},1}, 0) \\ \vdots \\ g(x_{\mathrm{op},K}, 0) \end{bmatrix}$$

$$G = \mathrm{diag}(G_0, G_1, G_2, \cdots, G_K)$$

$$R = \mathrm{diag}(R'_0, R'_1, R'_2, \cdots, R'_K)$$

离散时间的批量估计

- 利用高斯推断可得：

$$\begin{aligned} K &= \check{P}G^{\mathrm{T}}\left(G\check{P}G^{\mathrm{T}}+R'\right)^{-1} \\ \hat{P} &= (1-KG)\check{P} \\ \hat{x} &= \check{x}+K(y-y_{\mathrm{op}}-G(\check{x}-x_{\mathrm{op}})) \end{aligned}$$

- 使用SMW重新组织方程：$\left(\check{P}^{-1}+G^{\mathrm{T}}R'^{-1}G\right)\delta x^{*}=\check{P}^{-1}\left(\check{x}-x_{\mathrm{op}}\right)+G^{\mathrm{T}}R'^{-1}\left(y-y_{\mathrm{op}}\right)$
- 代入先验部分：

$$\underbrace{\left(F^{-\mathrm{T}}Q'^{-1}F^{-1}+G^{\mathrm{T}}R'^{-1}G\right)}_{\text{三对角块}}\delta x^{*}$$

$$=F^{-\mathrm{T}}Q'^{-1}\left(\nu-F^{-1}x_{\mathrm{op}}\right)+G^{\mathrm{T}}R'^{-1}\left(y-y_{\mathrm{op}}\right)$$

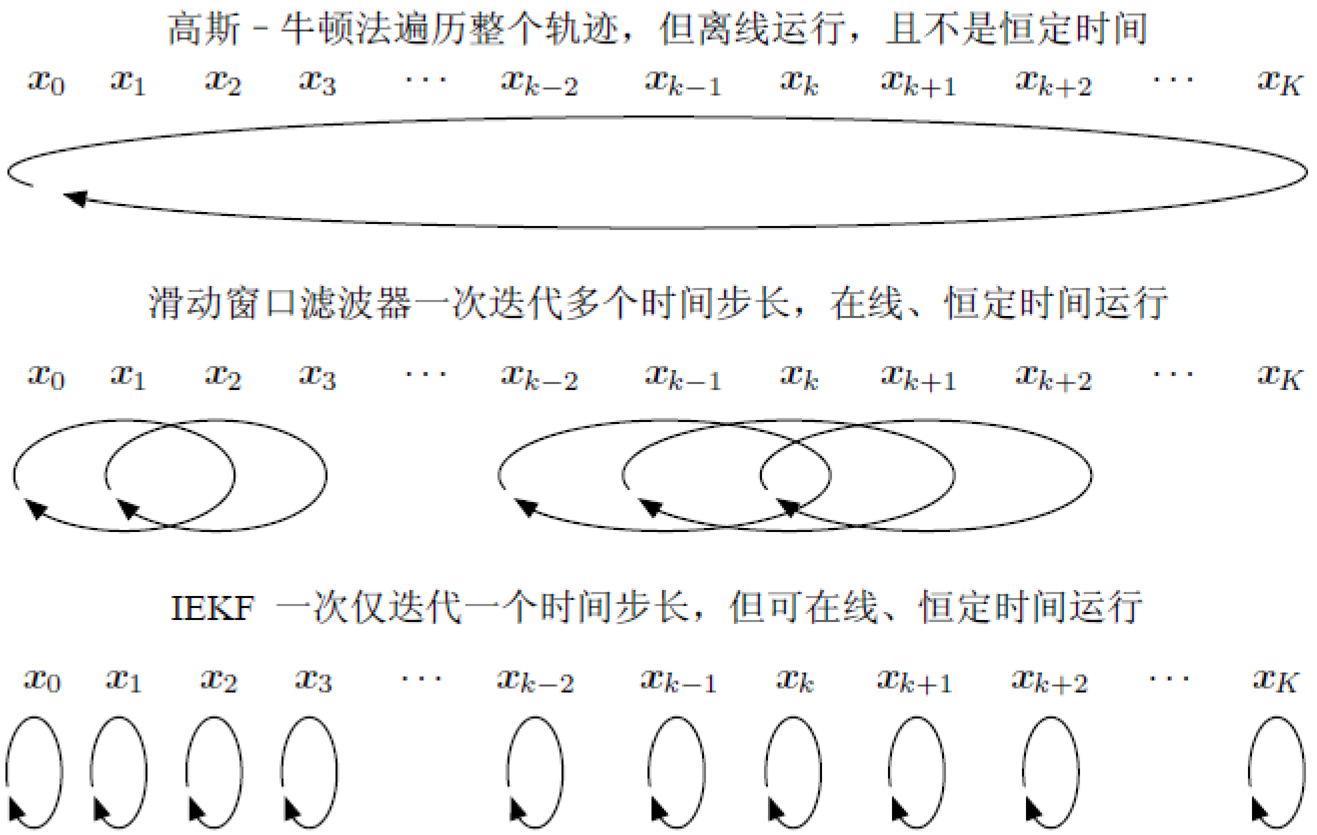
其中：$\delta x^{*}=\hat{x}-x_{\mathrm{op}}$

- 定义统一的矩阵：$H=\begin{bmatrix} F^{-1} \\ G \end{bmatrix}, \quad W=\operatorname{diag}(Q', R'), \quad e(x_{\mathrm{op}})=\begin{bmatrix} \nu-F^{-1}x_{\mathrm{op}} \\ y-y_{\mathrm{op}} \end{bmatrix}$
- 那么批量解：$\underbrace{\left(H^{\mathrm{T}}W^{-1}H\right)}_{\text{三对角块}}\delta x^{*}=H^{\mathrm{T}}W^{-1}e(x_{\mathrm{op}})$

- ☐ 贝叶斯推断和MAP基本是一样的，不过显式给出了协方差的估计

离散时间的批量估计

- 讨论
  - 相比于MAP，EKF的问题主要有两个：没有迭代过程，依赖马尔可夫假设
  - EKF基本上和单次MAP迭代一样（实际上稍微优于单次迭代）
  - 相比于各种针对EKF的改进，实际当中马尔可夫性是一个更本质的约束
  - 也有Sliding Window Filter这种介于批量和递归之间的滤波方法



离散时间的批量估计

- 本节课省略的内容
  - 高斯-牛顿法和Leverberg-Marquardt法的推导
  - 最大似然估计（ML）
  - 连续时间的估计问题

习题

1. 完成PPT第17页的推导
2. 验证27页下方式子
3. 课后习题1