

从推算到最优估计：卡尔曼滤波

——信息工程中的数学建模与应用

何铭源 郑一鸣 俞宸煜

何欣艺 任子桐

数学建模课程
第七组

2025 年 12 月 28 日

演讲提纲

① 提纲

② 背景引入

③ 数学模型构建

④ 核心算法流程

⑤ 公式推导与证明

⑥ 数值推演实例

⑦ 工程应用与仿真

⑧ 扩展卡尔曼滤波 (EKF)

⑨ 前沿探索

⑩ 结论

⑪ 参考文献

章节导航

1 提纲

2 背景引入

问题引入

历史中的解决办法

卡尔曼的构想

3 数学模型构建

4 核心算法流程

5 公式推导与证明

6 数值推演实例

7 工程应用与仿真

8 扩展卡尔曼滤波 (EKF)

9 前沿探索

10 结论

11 参考文献





信号估计问题

- 问题：如何从包含噪声的观测序列中实时获知系统状态？
- 例子：雷达系统、导航、语音信号增强、心电图去噪、信道状态估计等.
- 如何获知系统的状态？
 - 理论计算：
 - 用动力学等方式对系统进行建模，获知系统的状态方程，根据上一时刻的状态获知这一时刻的状态. 例如，用牛顿力学对小车进行建模，获取其坐标、速度、加速度的关系.
 - 然而，理论模型不总是理想的，存在模型无法完全涵盖的噪声，无法彻底消除.
 - 外部观测：
 - 用各种传感器进行探测，进而获知系统状态. 例如，通过小车车轮电机的传感器获知转速，通过雷达获知小车的位置.
 - 但是，传感器测量也存在一定误差，无法彻底消除.
- 如果不处理这些噪声，计算值和观测值将直接用于识别和控制，可能造成判断失误和控制失稳. 对于导航等不少对信号累积处理的系统中，误差将随时间累积，最终可能严重偏离实际情况.



直观的例子：蒙眼走直线的挑战

想象你被蒙住双眼，试图沿着直线走 100 米：

- **内部推算**: 你根据步数和步长估计自己位置.
→ **问题**: 步长有波动，步数累积的误差最终会使我们偏离直线.
- **外部观测**: 偶尔通过触碰地面标记、手杖探测前方距离，或听脚步声回响判断位置.
→ **问题**: 感知存在噪声或延迟，不完全可靠.

大脑的权衡

大脑如何判断当前位置？

- 内部推算提供连续更新，但误差会累积.
- 外部观测提供修正，但测量不精确.

结合两者，获得最可靠的位置信息.

最小二乘法

- 历史起源：1809 年高斯用于预测谷神星轨道.
- 数学模型：设观测值 $y \in \mathbb{R}^m$ 与真实值 $x \in \mathbb{R}^n$ 线性相关， $n < m$ ， x 满足 $y = Hx + v$ ，其中 H 为由建模得知的测量矩阵， v 为噪声. 求解 $\arg_x \min \|y - Hx\|^2$ ，得到
$$\hat{x} = (H^T H)^{-1} H^T y$$
- 劣势：
 - 没有充分考虑系统时变的特点，不能很好地利用已有数据；
 - 如果只观测一次，某些次观测的异常值可能对系统造成影响；如果观测多次，系统状态在这一时段已经发生了变化.

维纳滤波

- 历史贡献：1940 年代由诺伯特·维纳提出，将滤波问题引入随机过程的统计范畴.
- 核心内容：利用信号与噪声的功率谱密度，在频域内设计传递函数.
- 劣势：
 - 同样，无法适应时变系统和动态信号.

卡尔曼的构想

- **理论背景：**1960 年，Rudolf Kalman 发表论文《A New Approach to Linear Filtering and Prediction Problems》，引入了状态空间分析法.
- **核心思想：**
 - 放弃维纳滤波的频域思想，转而使用线性差分方程描述系统演进.
 - 将预测值和观察值结合起来，估计真实值.
- **早期困境：**由于其数学表达（矩阵运算与递归逻辑）过于超前，当时主流的信号处理学界对此反应冷淡，认为其缺乏工程实用性.



历史的转折

背景危机：

- 阿波罗飞船必须以极高精度切入月球轨道。NASA 正在寻求寻求地月转移轨道的实时精确定位算法。
- 机载计算机算力极低（仅 2KB RAM），存不下传统方法所需的历史数据。
- Kalman 在 NASA 艾姆斯研究中心展示其理论时，遇到了负责阿波罗导航系统的 Stanley Schmidt.
- Schmidt 意识到该算法的递归特性可以解决航天器导航中巨大的计算负载问题。并将其引入阿波罗系统。



图：左：R.E. Kalman | 右：阿波罗登月

历史里程碑

卡尔曼滤波成为了阿波罗导航计算机的灵魂，不仅把人送上了月球，也把他们带了回来。

卡尔曼滤波的 insight

- **最优状态估计器：**
 - 它不仅是传统意义上滤去噪声的滤波器；
 - 在动态系统中，还能根据噪声统计信息递推地估计系统状态.
- **对信息与控制的影响：**
 - ① **融合理论**: 将控制理论中的状态空间模型与统计信号处理（贝叶斯估计）结合.
 - ② **预测-更新范式**: 形成了递推估计的标准框架.
 - ③ **无处不在**:

- 手机定位 (GPS + 加速度计融合)
- 无人机导航与悬停 (IMU + 视觉融合)
- 导航、雷达跟踪、传感器融合等各类动态系统

章节导航

1 提纲

2 背景引入

3 数学模型构建

4 核心算法流程

5 公式推导与证明

6 数值推演实例

7 工程应用与仿真

8 扩展卡尔曼滤波 (EKF)

9 前沿探索

10 结论

11 参考文献





用向量描述系统状态

- 在同一时刻，系统通常需要多个物理量共同刻画，这些量彼此相关、共同变化.
- 一个自然的做法是：将它们组合为一个向量（如 $[位置, 速度]^T$ ）

$$\boldsymbol{x}_k = \begin{bmatrix} x_k^{(1)} & x_k^{(2)} & \cdots & x_k^{(n)} \end{bmatrix}^T$$

- 这种表示的优势：
 - 在一个对象中同时保留多个物理量
 - 物理量之间的**线性关系**可由线性映射刻画
 - 便于统一描述随时间变化的整体状态
- 与系统状态类似地，还可以用向量表示观测值、噪声、传入系统的控制量；用矩阵表示由观测值到系统状态的换算.

后续方程只涉及对向量、矩阵的变化描述



1. 状态方程：系统的物理特性

$$x_k = \underbrace{Fx_{k-1}}_{\text{状态转移}} + \underbrace{Bu_k}_{\text{控制输入}} + \underbrace{w_k}_{\text{过程噪声}}$$

(1)

- **状态转移项 Fx_{k-1}**

描述系统在无外界干预下的自然变化规律. F 为状态转移矩阵, 例如在匀速模型中, 它将上一刻的位置和速度映射到当前时刻.

- **控制输入项 Bu_k**

描述外部主动控制对系统的影响. 其中 u_k 是已知的控制量 (如油门开度、方向盘转角), B 为控制矩阵.

注: 在纯跟踪问题中, 若无主动控制, 此项为零.

- **过程噪声 w_k**

代表物理模型与真实世界之间的偏差, 包含风阻、摩擦等难以建模的随机扰动.



2. 观测方程：传感器的特性

$$z_k = \underbrace{Hx_k}_{\text{观测映射}} + \underbrace{v_k}_{\text{测量噪声}}$$

- **观测向量 z_k**
 k 时刻所有传感器读数按一定顺序排列的列向量.
- **观测映射项 Hx_k**
描述系统状态通过何种映射得到传感器读数.
例如: $H = [1, 0]$ 可将二维状态向量投影为一维观测值. 在 $x_k = [\text{位移}, \text{速度}]^T$ 的情形下, 这样的 H 对应了一个位置传感器.
- **测量噪声 v_k**
源于传感器自身的物理特性, 如电子热噪声、量化误差或环境干扰.

符号定义与噪声假设

状态量的维数: n ; 观测量的维数: m ; 控制量的维数: l .

符号	物理含义	维度
x_k	k 时刻的状态向量	$n \times 1$
F	状态转移矩阵	$n \times n$
B	控制输入矩阵	$n \times l$
z_k	k 时刻的观测向量	$m \times 1$
H	观测矩阵	$m \times n$

假设过程噪声和测量噪声都服从均值为 0 的正态分布.

$$w_k \sim N(0, Q), \quad v_k \sim N(0, R)$$

其中 Q 为过程噪声协方差矩阵, R 为测量噪声协方差矩阵.

章节导航

1 提纲

2 背景引入

3 数学模型构建

4 核心算法流程
算法总览
预测
更新

5 公式推导与证明

6 数值推演实例

7 工程应用与仿真

8 扩展卡尔曼滤波 (EKF)

9 前沿探索

10 结论

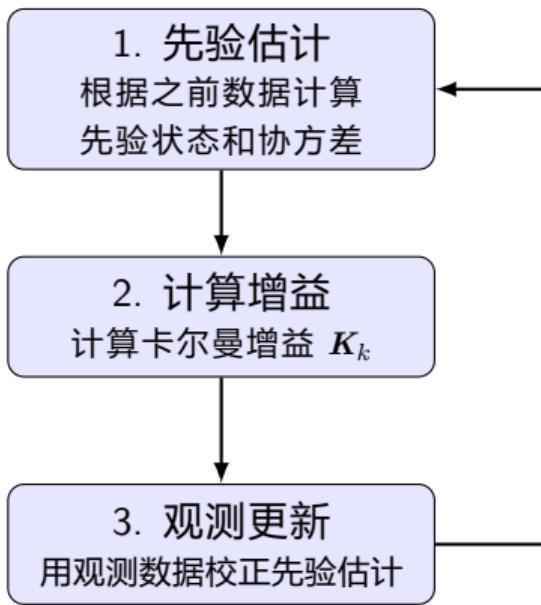
11 参考文献





算法总览

卡尔曼滤波是不断预测——更新的过程。每经过一段时间观测一次，就执行一次循环。





第一阶段：时间更新

在不涉及传感器数据的情况下，基于上一时刻的状态与物理模型进行先验估计。

1. 状态先验估计

$$\hat{x}_k^- = F\hat{x}_{k-1} + Bu_k \quad (3)$$

系统根据状态转移矩阵 F 和控制输入 Bu_k 推演当前时刻的理论状态。

2. 协方差先验估计

$$P_k^- = FP_{k-1}F^T + Q \quad (4)$$

P_k^- 是真实值和预测值的先验误差协方差矩阵， P_k 是真实值和预测值的后验误差协方差矩阵，误差随时间传递与扩散。



卡尔曼增益

卡尔曼增益 K_k 是最小均方误差准则下的最优加权因子.

一维形式的直觉: 此时公式可简化为:

$$K_k = \frac{P_k^-}{P_k^- + R}$$

物理含义:

$$\text{增益} = \frac{\text{预测不确定性}}{\text{预测不确定性} + \text{观测不确定性}}$$

- 分子代表模型预测的误差方差.
- 分母代表系统总的误差方差.
- K_k 动态衡量了我们应该多大程度上信任观测值.



矩阵形式的完整表达

在多维系统中，我们需要处理状态空间与观测空间的维度差异.

3. 计算卡尔曼增益

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}^T (\mathbf{H} \mathbf{P}_k^- \mathbf{H}^T + \mathbf{R})^{-1} \quad (5)$$

项 $(\mathbf{H} \mathbf{P}_k^- \mathbf{H}^T + \mathbf{R})^{-1}$ 充当了归一化因子的角色.



第二阶段：状态更新

利用计算出的增益，融合观测数据，得到最优的后验估计.

4. 状态后验更新

$$\hat{x}_k = \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-) \quad (6)$$

后验估计 = 先验预测 + 增益 × 新息.

其中 $(z_k - H\hat{x}_k^-)$ 被称为新息 (Innovation)，代表观测值与预测值的偏差.

5. 协方差后验更新

$$P_k = (I - K_k H) P_k^- \quad (7)$$

由于引入了观测信息，系统的不确定性得以降低 ($P_k < P_k^-$)，为下一时刻的计算做准备.



极限情形下的一些直觉

情形一：传感器极其精准 ($R \rightarrow 0$)

- $K_k \rightarrow 1$
- $\hat{x}_k = \hat{x}_k^- + 1 \cdot (z_k - \hat{x}_k^-) = z_k$
- 滤波器完全信任观测值，忽略模型预测.

情形二：模型极其精准或传感器故障 ($P_k^- \rightarrow 0$ 或 $R \rightarrow \infty$)

- $K_k \rightarrow 0$
- $\hat{x}_k = \hat{x}_k^- + 0 = \hat{x}_k^-$
- 滤波器完全信任模型预测，忽略观测数据.

结论：卡尔曼滤波总是自动偏向于“方差更小”的一方.

章节导航

1 提纲

2 背景引入

3 数学模型构建

4 核心算法流程

5 公式推导与证明

公式推导——以一维为例
一般情形证明

6 数值推演实例

7 工程应用与仿真

8 扩展卡尔曼滤波 (EKF)

9 前沿探索

10 结论

11 参考文献





0. 设定场景：两个不靠谱的信源

假设我们要测量小车的**位置**.

- **信源 1 (模型推算)：**
 - 值: x_k^- (比如 100 米)
 - 噪声: $w \sim N(0, P^-)$ (方差为 P^-)
- **信源 2 (传感器测量)：**
 - 值: z_k (比如 102 米)
 - 噪声: $v \sim N(0, R)$ (方差为 R)

目标: 找到一个最优估计 \hat{x}_k , 让它的**误差方差**最小.





1. 构建融合方程 I

根据直觉，我们采用**加权平均**的方式来融合：

$$\hat{x}_k = (1 - K)x_k^- + Kz_k$$

为什么系数是这样分配的？

这源于“**修正**”的思想： $\hat{x}_k = x_k^- + K(z_k - x_k^-)$.

- 我们以**预测** x_k^- 为基准.
- K 代表我们多大程度上相信**观测** z_k 来修正这个基准.
- 展开后即得： $(1 - K)x_k^- + Kz_k$.

代数变形：假设真实值是 x_{true} .

- **推算值** $x_k^- = x_{true} + w$
- **观测值** $z_k = x_{true} + v$



1. 构建融合方程 II

代入融合方程：

$$\begin{aligned}\hat{x}_k &= (1 - K)(x_{true} + w) + K(x_{true} + v) \\ &= (1 - K)x_{true} + (1 - K)w + Kx_{true} + Kv \\ &= \underbrace{x_{true}}_{\text{真值}} + \underbrace{(1 - K)w + Kv}_{\text{综合误差}}\end{aligned}$$





1.5 深度思考：为什么不用几何/平方平均？

有同学可能会问：为什么要用线性的 $(1 - K)a + Kb$ ，用更高级的公式不行吗？

方案 A：几何平均 $\sqrt{x \cdot z}$

致命缺陷：负数地狱

- 状态可能是负数（如倒车速度为 -5m/s ）。
- 根号下出现负数会导致系统崩溃（变成虚数）。

方案 B：平方平均 $\sqrt{\frac{x^2+z^2}{2}}$

致命缺陷：引入偏差

- 噪声平方后恒为正。
- 这会导致估计值**永远偏大**，不再是“无偏估计”。

结论：只有**线性组合**能保持高斯分布特性，并保证平均误差为 0.



2. 计算综合方差

我们的目标是让**综合误差的方差最小**.

令总误差为 $e = (1 - K)w + Kv$.

因为 w 和 v 是独立的, 于是可以利用方差性质, 得到融合后的总方差 P_k 为:

$$P_k(K) = \underbrace{(1 - K)^2 P^-}_{\text{推算部分的贡献}} + \underbrace{K^2 R}_{\text{测量部分的贡献}}$$

这就是我们要优化的目标函数, 它是关于 K 的开口向上的抛物线.

(8)



3. 求极值

为了找到方差最小的 K , 我们对 K 求导并令其为 0.

$$\frac{dP_k}{dK} = \frac{d}{dK}[(1 - K)^2 P^- + K^2 R] = 0$$

整理上式, 把含 K 的项放在左边, 常数项放在右边:

$$K(P^- + R) = P^-$$

最优卡尔曼增益 (Kalman Gain)

$$K = \frac{P^-}{P^- + R}$$



回顾

状态方程和观测方程

- **状态方程:** $x_k = Fx_{k-1} + Bu_k + w_k$
- **观测方程:** $z_k = Hx_k + v_k$

其中 $w_k \sim N(0, Q)$, $v_k \sim N(0, R)$.

一些记号

- x_k : 本时刻状态的真实值.
- \bar{x}_k : 由上一时刻状态和状态方程推出的状态先验估计值.
- \hat{x}_k : 经某些修正后的后验估计值.

我们的目标: 希望 \hat{x}_k 成为 x_k 的比较好的估计量:

- 无偏估计;
- 均方误差尽可能小.



引入观测值

残差

如果 $v_k = 0$ 而 $x_k = \bar{x}_k$, 根据观测方程应有 $z_k = H\bar{x}_k$. 可见, 残差 $z_k - Hx_k$ 在某种程度上反映了观测值与先验值的差距. 残差越大, 说明先验值越有必要修正, 基于残差的某种表达式修正先验值的想法是很自然的.

转移方程

我们可以为残差左乘系数矩阵 K , 再加上先验值作为修正, 得到如下转移方程, 也得到了修正值的表达式. 接下来的工作变为求最优的 K , 使得 \hat{x}_k 是无偏而且均方误差最小的.

$$\hat{x}_k = \bar{x}_k + K(z_k - H\bar{x}_k) \quad (9)$$



误差

先验值与真实值的误差

$$e_k^- = x_k - \bar{x}_k \quad (10)$$

修正值与真实值的误差

$$\begin{aligned} e_k &= x_k - \hat{x}_k \\ &= x_k - \bar{x}_k^- - K(Hx_k + v_k - H\bar{x}_k^-) \\ &= (I - KH)(x_k - \bar{x}_k^-) - Kv_k \\ &= (I - KH)e_k^- - Kv_k \end{aligned} \quad (11)$$



矩阵中的期望与协方差

随机向量 $x \in \mathbb{R}^n$:

$$\boldsymbol{x} = [x_1 \quad x_2 \quad \cdots \quad x_n]^T$$

期望:

$$\mathbb{E}[\boldsymbol{x}] = [\mathbb{E}[x_1] \quad \mathbb{E}[x_2] \quad \cdots \quad \mathbb{E}[x_n]]^T$$

协方差矩阵:

$$\text{Cov}(\boldsymbol{x}) = \mathbb{E}[(\boldsymbol{x} - \mathbb{E}[\boldsymbol{x}])(\boldsymbol{x} - \mathbb{E}[\boldsymbol{x}])^T]$$

其中 $P_{ij} = \text{Cov}(x_i, x_j)$

一些性质:

- 协方差矩阵是对称的: $\text{Cov}(\boldsymbol{x})^T = \text{Cov}(\boldsymbol{x})$.
- 若 $y = Ax + b$, 则 $\text{Cov}(\boldsymbol{y}) = A\text{Cov}(\boldsymbol{x})A^T$



修正值的协方差

修正值与真实值的误差 $e_k = (\mathbf{I} - \mathbf{K}\mathbf{H}) e_k^- - \mathbf{K}\mathbf{v}_k$ 其协方差为：

$$\begin{aligned} \mathbf{P}_k &= \text{Cov}(e_k) \\ &= \text{Cov}((\mathbf{I} - \mathbf{K}\mathbf{H}) e_k^- - \mathbf{K}\mathbf{v}_k) \\ &= (\mathbf{I} - \mathbf{K}\mathbf{H}) \text{Cov}(e_k^-) (\mathbf{I} - \mathbf{K}\mathbf{H})^T + \mathbf{K} \text{Cov}(\mathbf{v}_k) \mathbf{K}^T \\ &\quad + \underbrace{\text{Cov}((\mathbf{I} - \mathbf{K}\mathbf{H}) e_k^-, -\mathbf{K}\mathbf{v}_k)}_{0(\text{独立性假设})} \end{aligned}$$

最终得到：

$$\boxed{\mathbf{P}_k = (\mathbf{I} - \mathbf{K}\mathbf{H}) \mathbf{P}_k^- (\mathbf{I} - \mathbf{K}\mathbf{H})^T + \mathbf{K} \mathbf{R}_k \mathbf{K}^T}$$

说明：

- $\mathbf{P}_k^- = \text{Cov}(e_k)^-$.
- 使用了协方差线性性： $\text{Cov}(A\mathbf{x} + B\mathbf{y}) = A \text{Cov}(\mathbf{x}) A^T + B \text{Cov}(\mathbf{y}) B^T$
- 使用了独立性假设： e_k^- 与 \mathbf{v}_k 独立 \Rightarrow 交叉项为 0



协方差推导：后验与先验

后验协方差：

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k \mathbf{H}) \mathbf{P}_k^- (\mathbf{I} - \mathbf{K}_k \mathbf{H})^T + \mathbf{K}_k \mathbf{R} \mathbf{K}_k^T = (\mathbf{I} - \mathbf{K} \mathbf{H}) \mathbf{P}_k^-$$

先验协方差：

$$\begin{aligned}\mathbf{P}_k^- &= \text{Cov}(\mathbf{e}_k^-) \\ &= \text{Cov}(\mathbf{F}_k \mathbf{e}_{k-1} + \mathbf{w}_k) \\ &= \mathbf{F}_k \text{Cov}(\mathbf{e}_{k-1}) \mathbf{F}_k^T + \text{Cov}(\mathbf{w}_k) \\ &= \mathbf{F}_k \mathbf{P}_{k-1} \mathbf{F}_k^T + \mathbf{Q}_k\end{aligned}$$

运用这两个公式，即可在下一步计算时更新所需的协方差矩阵.



卡尔曼滤波伪代码

输入: 初始状态 \hat{x}_0 , 初始协方差 P_0 , 系统矩阵 F_k, B_k, H_k , 控制 u_k , 协方差 Q_k, R_k , 测量 z_k

① 循环 $k = 1$ 到 N :

① 先验状态预测:

$$\hat{x}_k^- = F_k \hat{x}_{k-1} + B_k u_k$$

② 先验协方差预测:

$$P_k^- = F_k P_{k-1} F_k^T + Q_k$$

③ 卡尔曼增益计算:

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1}$$

④ 后验状态更新:

$$\hat{x}_k = \hat{x}_k^- + K_k (z_k - H_k \hat{x}_k^-)$$

⑤ 后验协方差更新:

$$P_k = (I - K_k H_k) P_k^-$$

② 输出: \hat{x}_k, P_k 迭代序列

章节导航

1 提纲

2 背景引入

3 数学模型构建

4 核心算法流程

5 公式推导与证明

6 数值推演实例

7 工程应用与仿真

8 扩展卡尔曼滤波 (EKF)

9 前沿探索

10 结论

11 参考文献



场景设定：无人机悬停高度估计

场景：

- 我们想知道无人机的真实高度 x .
- 无人机正在悬停，理论上不动，但有风吹（过程噪声）.
- 使用气压计测量高度，但有误差（测量噪声）.

参数初始化 ($t = 0$):

- 初始估计 $\hat{x}_0 = 10m$.
- 初始不确定性 $P_0 = 5$ (不太确信).

系统参数：

- 状态转移 $F = 1$ (不动).
- 过程噪声 $Q = 1$.
- 测量噪声 $R = 4$ (传感器一般).
- 观测矩阵 $H = 1$ (直接测高度).



Step 1: 预测 @ t=1

还没看传感器，先根据物理规律猜.

1. 状态预测

$$\hat{x}_1^- = F\hat{x}_0 = 1 \times 10 = 10$$

(我猜它还在 10m 处)

2. 协方差预测

$$P_1^- = FP_0F^T + Q = 1 \times 5 \times 1 + 1 = 6$$

(如果不看传感器，不确定性从 5 增加到了 6，因为风在吹)



Step 2: 计算卡尔曼增益 @ t=1

现在传感器读数来了: $z_1 = 12m.$

3. 计算增益 K

$$K_1 = \frac{P_1^- H^T}{H P_1^- H^T + R} = \frac{6 \times 1}{1 \times 6 \times 1 + 4} = \frac{6}{10} = 0.6$$

解读:

- $K = 0.6$ 意味着我们把 **60%** 的信任给了观测值 (残差), **40%** 保留给预测值.
- 为什么信观测多一点? 因为预测误差方差 (6) 大于测量噪声方差 (4).



Step 3: 更新 @ t=1

融合预测和观测，得到最优结果.

4. 状态更新

$$\hat{x}_1 = \hat{x}_1^- + K_1(z_1 - \hat{x}_1^-)$$

$$\hat{x}_1 = 10 + 0.6 \times (12 - 10) = 10 + 1.2 = \mathbf{11.2m}$$

5. 协方差更新

$$P_1 = (1 - K_1 H) P_1^- = (1 - 0.6) \times 6 = 0.4 \times 6 = \mathbf{2.4}$$

结果：

- 估计值 11.2m 介于 10m 和 12m 之间，偏向观测值.
- 不确定性从 6 骤降至 2.4. 滤波让我们更自信了！

章节导航

- 1 提纲
- 2 背景引入
- 3 数学模型构建
- 4 核心算法流程
- 5 公式推导与证明
- 6 数值推演实例

- 7 工程应用与仿真
 - 仿真案例
 - 信息工程中的案例
 - 微电子中的案例
- 8 扩展卡尔曼滤波 (EKF)
- 9 前沿探索
- 10 结论
- 11 参考文献



仿真案例：一维小车位置追踪

场景假设：

- 小车做匀速运动，速度 $v = 1m/s$.
- 过程噪声 Q : 路面颠簸导致速度微小波动.
- 测量噪声 R : GPS 定位误差 $\pm 5m$.



仿真案例：一维小车位置追踪

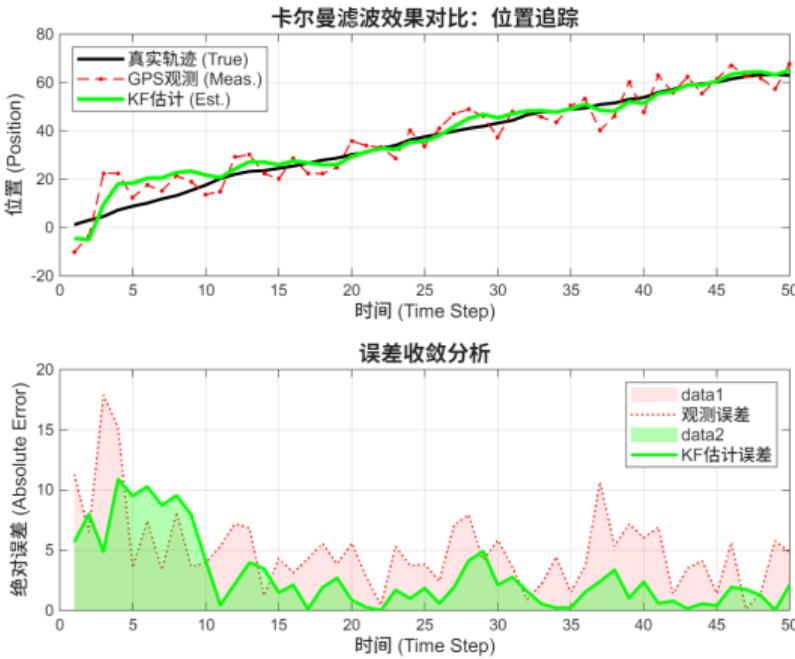


图: 利用 Matlab 模拟卡尔曼滤波



参数 Q 与 R 的调节策略

在信息工程实践中, F 和 H 通常由物理系统决定, 但 Q 和 R 是调试的关键.

信任模型 (Q 小, R 大)

- 滤波结果非常平滑
- 对突变反应滞后

信任观测 (Q 大, R 小)

- 紧跟观测值变化
- 引入较多噪声

结论: 调参本质上是在**响应速度与平滑度**之间做权衡.



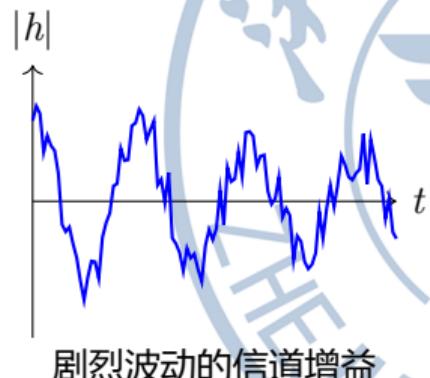
信息工程案例 (1/3): 高速移动通信的挑战

背景: 在高铁或车载通信场景 (5G/V2X) 中, 终端的高速移动会产生严重的**多普勒频移** (Doppler Shift).

问题: 快时变信道 (Fast Fading)

信道冲激响应 h_k 不再是常数, 而是随时间快速变化的随机过程.

- 传统方法 (如最小二乘 LS): 假设信道在一段 block 内不变 → **失效**.
- 需求: 必须利用**时间相关性**进行实时追踪.





信息工程案例 (2/3): AR(1) 动力学建模

为了使用 Kalman 滤波，我们需要建立状态方程。文献 komninakis2002multi 提出利用一阶自回归模型 (AR1) 近似 Jakes 衰落模型。

$$\text{状态方程: } h_k = \alpha h_{k-1} + w_k$$

(12)

- h_k : 复数信道系数 (状态量).
- α : **自相关系数**, 由物理环境决定:

$$\alpha = J_0(2\pi f_d T_s)$$

(J_0 : 零阶贝塞尔函数, f_d : 多普勒频率, T_s : 符号周期)

- w_k : 过程噪声, 代表建模误差, 方差 $\mathbf{Q} = 1 - |\alpha|^2$.

建模的核心思想

不仅仅是“估计”当前值，而是利用 f_d 预测信道的“惯性”变化趋势。



信息工程案例 (3/3): KF vs LS 性能对比

我们将卡尔曼滤波 (KF) 与传统的最小二乘 (LS) 估计算法进行对比.

观测方程: $y_k = s_k h_k + v_k$ (s_k 为已知导频)

特性	最小二乘 (LS)	卡尔曼滤波 (KF)
原理	仅基于当前观测 y_k	融合当前观测 y_k + 历史预测
计算量	低 ($O(1)$)	中等 ($O(N^3)$ 矩阵求逆)
抗噪性	差 (随 SNR 线性下降)	强 (有效滤除高斯白噪)
适用场景	静态/慢速环境	高速移动/低信噪比环境

结论: KF 能够提供比 LS 高约 3dB 的信噪比增益, 显著降低误码率 (BER).



微电子案例 (1/3): MEMS 传感器的误差源

在微电子制造中，低成本 MEMS 陀螺仪（如手机中的 IMU）受限于工艺，存在两类主要误差 **woodman2007introduction**:

1. 角度随机游走 (White Noise)

- 来源: 电子热噪声、机械抖动.
- 特性: 高频噪声, 均值为 0.
- KF 对应: 测量噪声矩阵 R .

2. 零偏不稳定性 (Bias Instability)

- 来源: 温度漂移、应力释放.
- 特性: 低频缓慢变化, 类似 $1/f$ 噪声.
- 危害: 直接积分会导致角度误差随时间 t 二次发散.



微电子案例 (2/3): 状态扩维 (Augmented State)

为了消除零偏 (Bias), 我们将未知的 Bias 也视为一个“状态”放入滤波器中进行估计.

系统状态向量: $x = [\theta, \text{bias}]^T$

离散化状态空间模型:

$$\begin{bmatrix} \theta_k \\ b_k \end{bmatrix} = \begin{bmatrix} 1 & -\Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \theta_{k-1} \\ b_{k-1} \end{bmatrix} + \begin{bmatrix} \omega_{mea}\Delta t \\ 0 \end{bmatrix} + w_k$$

- 第一行: $\theta_k = \theta_{k-1} + (\omega_{mea} - b_{k-1})\Delta t$
(解释: 真实角速度 = 测量值 - 漂移值)
- 第二行: $b_k = b_{k-1} + w_b$
(解释: 假设漂移本身是一个随机游走过程, 即 Bias 也是在变的)



微电子案例 (3/3): 软件定义硬件精度

通过卡尔曼滤波，我们实现了一种“虚拟传感器”的效果。

算法流程闭环：

- ① **预测**: 利用上一时刻估算的 Bias 修正当前的陀螺仪读数.
- ② **观测**: 引入加速度计或磁力计数据作为“观测值” z_k .
- ③ **校正**: 计算卡尔曼增益 K , 同时更新 **角度** θ 和 **零偏** b .

注：这是自动驾驶、无人机飞控中最底层的核心算法之一。

工程价值

精度提升 10×

利用 KF 算法，可以将售价 \$5 的消费级 MEMS 芯片，提升至接近工业级 (\$50) 的性能表现。

章节导航

1 提纲

2 背景引入

3 数学模型构建

4 核心算法流程

5 公式推导与证明

6 数值推演实例

7 工程应用与仿真

8 扩展卡尔曼滤波 (EKF)

卡尔曼滤波的局限性
核心数学工具
算法对比
历史脉络与学术发展

9 前沿探索

10 结论

11 参考文献

1. 线性系统的局限性

标准 KF 的致命弱点

标准卡尔曼滤波假设系统是**线性的**:

$$\mathbf{x}_k = \mathbf{F}\mathbf{x}_{k-1} \quad \text{和} \quad \mathbf{z}_k = \mathbf{H}\mathbf{x}_k$$

然而，现实世界充满了**非线性**:

- **机器人运动**: $x_{new} = x_{old} + v \cdot \cos(\theta) \cdot \Delta t$
- **雷达观测**: $r = \sqrt{x^2 + y^2}$

问题: 高斯分布经过非线性变换后，**不再是高斯分布**，导致标准 KF 的公式失效。



2. 解决思路：线性化

既然非线性太难处理，我们能不能在**局部**把它看作线性的？

泰勒级数展开

对于非线性函数 $f(x)$ ，我们在估计点 \hat{x} 附近做一阶展开：

$$f(x) \approx f(\hat{x}) + \underbrace{\frac{\partial f}{\partial x} \Big|_{x=\hat{x}}}_{\text{切线斜率}} \cdot (x - \hat{x}) \quad (14)$$

- 我们抛弃高阶项，只保留一阶导数。
- 用**切线**来近似**曲线**。



3. 核心工具：雅可比矩阵

在多维系统中，导数变成了**雅可比矩阵**. 它是 EKF 的核心.
假设状态转移函数为 $x_k = f(x_{k-1}, u_k)$, 则雅可比矩阵 F_k 为:

$$F_k = \frac{\partial f}{\partial x} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \dots \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \dots \\ \vdots & \vdots & \ddots \end{bmatrix}$$

物理意义：描述了输入微小变化如何影响输出的每一个维度.



4. EKF 算法流程的变化 (对比 KF)

EKF 的步骤与 KF 几乎一样，区别在于**如何传递均值和如何传递协方差**.

步骤	标准 KF (线性)	扩展 KF (非线性)
状态预测	$x^- = Fx$	$x^- = f(x, u)$ (直接代入函数)
协方差预测	$P^- = FPF^T + Q$	$P^- = F_k P F_k^T + Q$ (用雅可比)
卡尔曼增益	$K = P^- H^T (\dots)^{-1}$	$K = P^- H_k^T (H_k P^- H_k^T + R)^{-1}$
状态更新	$x = x^- + K(z - Hx^-)$	$x = x^- + K(z - h(x^-))$



5. 经典案例：雷达追踪 (Radar Tracking)

状态向量 (直角坐标系):

$$x = [p_x, p_y, v_x, v_y]^T$$

(飞机的位置和速度)

观测向量 (极坐标系):

$$z = [\rho, \phi, \dot{\rho}]^T$$

(雷达测量的距离、角度、径向速度)

非线性观测函数 $h(x)$:

- 距离: $\rho = \sqrt{p_x^2 + p_y^2}$
- 角度: $\phi = \arctan(p_y / p_x)$

问题

p_x 和 p_y 是状态，但在观测方程里被平方和求根了。这就是典型的非线性！必须求雅可比矩阵 H_j .



6. 雷达雅可比矩阵的计算

我们需要对 $h(x)$ 求偏导数来得到 H_j :

$$H_j = \frac{\partial h}{\partial x} = \begin{bmatrix} \frac{\partial \rho}{\partial p_x} & \frac{\partial \rho}{\partial p_y} & 0 & 0 \\ \frac{\partial \phi}{\partial p_x} & \frac{\partial \phi}{\partial p_y} & 0 & 0 \end{bmatrix} = \begin{bmatrix} \frac{p_x}{\sqrt{p_x^2+p_y^2}} & \frac{p_y}{\sqrt{p_x^2+p_y^2}} & 0 & 0 \\ \frac{-p_y}{p_x^2+p_y^2} & \frac{p_x}{p_x^2+p_y^2} & 0 & 0 \end{bmatrix}$$
(16)

意义: 这个矩阵把**位置的不确定性** (直角坐标误差), 投影到了**观测的不确定性** (距离和角度误差) 上.



EKF 的诞生与发展：从航天到芯片

扩展卡尔曼滤波的发展史，就是一部将理论数学转化为工程实践的历史。

- **1960: 线性起源**

Rudolf E. Kalman 发表了经典论文，但仅适用于线性系统。

- **1961: 阿波罗的困境**

NASA 的 *Stanley F. Schmidt* 发现飞船轨道方程是非线性的。

- **突破点：**

Schmidt 创造性地提出了在参考轨迹附近进行**线性化**的想法，发明了 EKF.^a

^aSchmidt, "The Kalman Filter: Its Recognition and Development for Aerospace Applications."





1970s-1990s: 信息工程领域的深化

随着计算机算力的提升，EKF 开始从航空航天下沉到一般的信号处理和控制领域。

目标追踪

Yaakov Bar-Shalom 等人完善了 EKF 在雷达追踪中的应用理论。^a

- 关键贡献：提出了混合估计和交互式多模型。

^aBar-Shalom and Fortmann, *Tracking and Data Association*.

机器人定位 (SLAM) 的开端

Smith, Self, & Cheeseman (1990) 发表了开创性论文，提出用 EKF 同时估计机器人的位置和地图路标点。^a

^aSmith et al., "Estimating uncertain spatial relationships in robotics."

2000s: 算法的收敛性证明与微电子爆发

1. SLAM 的理论基石

- 悉尼科技大学的 **Gamini Dissanayake** 等人证明了 EKF-SLAM 的解是收敛的¹, 确立了其统治地位.

2. 能源电子的革命 (微电子/电力电子)

- 科罗拉多大学的 **Gregory L. Plett** 将 EKF 引入锂电池管理 (BMS), 解决了 SOC 估算难题.²

¹Dissanayake et al., "A solution to the simultaneous localization and map building (SLAM) problem."

²Plett, "Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs. Part 2. Modeling and identification."

7. EKF 的优缺点分析

优点

- **事实标准**: 目前导航系统 (GPS/IMU 融合)、机器人定位 (SLAM) 的首选基础算法.
- **计算量适中**: 比粒子滤波 (Particle Filter) 快得多.

缺点

- **发散风险**: 如果线性化点选得不好 (初始估计太差), 泰勒展开误差会很大, 导致滤波器发散.
- **繁琐的雅可比**: 对于复杂的系统, 计算雅可比矩阵非常痛苦且容易出错.



8. 为什么我们需要 UKF?

当系统具有**强非线性**或雅可比矩阵**难以计算**时，EKF 会失效.

EKF 的痛点

- ① **精度丢失**: 泰勒展开只保留了一阶项, 对于剧烈变化的函数, 高阶误差巨大.
- ② **计算难度大**: 很多复杂物理模型 (如流体、化学反应) 根本写不出解析的雅可比矩阵 $F = \frac{\partial f}{\partial x}$.

UKF 的哲学: 无迹变换

- **不线性化函数**, 而是去近似概率分布.
- **Sigma 点采样**: 选取几个确定性的点, 把它们扔进非线性函数里跑一遍.
- **加权重组**: 根据变换后的点, 统计出新的均值和协方差.

“近似概率分布比近似任意非线性函数要容易得多.”

——Julier & Uhlmann³

³Julier and Uhlmann, “New extension of the Kalman filter to nonlinear systems.”



总结：从 EKF 到 UKF 的演进逻辑

特性	EKF (扩展卡尔曼)	UKF (无迹卡尔曼)
核心思想	线性化：用切线逼近曲线	采样：用点集逼近分布
数学工具	雅可比矩阵 (Jacobian)	无迹变换 (Unscented Transform)
适用场景	轻度非线性 (如 GPS 定位)	强非线性、导数不存在 (如复杂化学反应、机动目标)
工程代价	推导公式极难，计算极快	推导简单 (黑盒)，计算量略大 ($2L + 1$ 次运算)

章节导航

1 提纲

2 背景引入

3 数学模型构建

4 核心算法流程

5 公式推导与证明

6 数值推演实例

7 工程应用与仿真

8 扩展卡尔曼滤波 (EKF)

9 前沿探索
算法演进全景
卡尔曼滤波遇上神经网络
前沿案例

10 结论

11 参考文献



1. 算法演进全景

卡尔曼滤波已发展为庞大的家族。针对不同场景，主流算法的特性对比如下⁴：

算法	核心机制	优势 (Pros)	适用场景
Basic KF	线性递归最小二乘	计算极快，线性最优	卫星导航，稳态控制
EKF	泰勒展开线性化 (J)	工业标准，适用性广	机器人定位 (SLAM)
UKF	无迹变换 (Sigma 点)	无需导数，精度更高	复杂非线性系统
IMM	多模型概率交互	搞定机动目标 ，平滑切换	导弹/无人机追踪

⁴Khodarahmi and Maihami, "A Review on Kalman Filter Models."

2. 突破瓶颈：混合模型的诞生

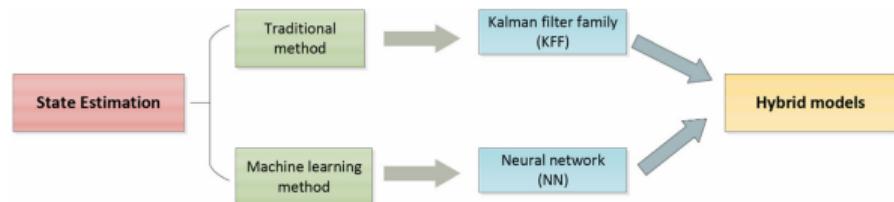


图: 状态估计技术融合路线图^a

^aFeng et al., "A review."

为何要融合?

- **传统 KF:** 依赖精确物理模型，难以处理未知环境.
- **神经网络 (NN):** 强大的数据拟合能力，但缺乏物理约束.

融合的三种模式:

- ① **串联:** 互为预处理/后处理.
- ② **训练:** KF 优化 NN 权重.
- ③ **辅助 (Aided):** NN 实时估计 KF 参数 (Q, R).



深度思考：卡尔曼滤波 vs 机器学习

卡尔曼滤波 (KF)

“理性的物理学家”

- **驱动核心：**物理模型 (F, H) + 概率统计.
- **透明度：**白盒. 每一步都有明确物理含义，可解释性强.
- **优势：**小样本即可工作，不仅给结果，还给置信度 (P).
- **劣势：**模型必须已知且准确.

机器学习 (ML/NN)

“经验丰富的工匠”

- **驱动核心：**海量数据 + 拟合映射.
- **透明度：**黑盒. 内部权重难以解释.
- **优势：**能拟合极其复杂的非线性关系，无需懂物理机理.
- **劣势：**数据饥渴，对未见过的场景泛化能力弱.

融合的哲学：

用 KF 的逻辑框架约束 ML 的发散，用 ML 的拟合能力弥补 KF 的模型缺陷.



雷达追踪中的“奇异点”

在雷达追踪中，如果飞机做**协同转弯运动 (CTRV)**，状态向量包含转弯率 ω ：

$$\boldsymbol{x} = [p_x, p_y, v, \psi, \omega]^T$$

状态转移方程：

$$p_{x,k+1} = p_{x,k} + \frac{v_k}{\omega_k} (\sin(\psi_k + \omega_k \Delta t) - \sin(\psi_k))$$
(17)

EKF 在此处的致命缺陷

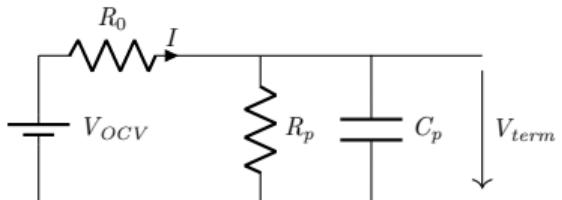
当我们尝试求雅可比矩阵 $\frac{\partial f}{\partial \omega}$ 时，分母含有 ω .

- 当飞机**直线飞行时**， $\omega \rightarrow 0$.
- 此时雅可比矩阵中会出现 $\frac{0}{0}$ 型的**除零奇点**，导致 EKF 数值爆炸.

UKF 优势：不~~需要~~求导，直接代入数值计算，完美避开数学奇点.⁵

⁵Li and Jilkov, “Survey of maneuvering target tracking. Part I: Dynamic models.”

基础阶段：基于二阶 RC 模型的 SOC 估算



1. 物理建模：利用二阶 RC 电路模拟电池极化效应.

2. 状态定义：

$$\mathbf{x} = [SOC, V_p]^T$$

3. 核心挑战： V_{OCV} 与 SOC 呈非线性关系 (查表), 导致无法直接使用线性卡尔曼滤波 (KF).

工业界标准解法：EKF (扩展卡尔曼滤波)

由于 V_{OCV} 曲线相对平滑, 我们通过**一阶泰勒展开** (求导) 即可获得良好的线性化近似.

$$\mathbf{H}_k = \left[\frac{\partial V_{OCV}}{\partial SOC}, -1 \right]$$

此时, EKF 能够有效修正电流积分带来的累积误差.

进阶挑战：电池老化带来的“模型失配”

现实困境：EKF 准确的前提是**模型参数** (R_0, R_p, C_p) 是已知且准确的.

- 然而，随着电池老化和温度变化，内阻 R_0 可能增加 100%，电容 C_p 也会漂移.
- **后果：**如果坚持使用固定参数模型，EKF 会强行把模型误差“归咎”于 SOC，导致 SOC 估算严重偏离.

解决方案：联合估算将参数也视为状态的一部分，构建**增强状态向量**:

$$\boldsymbol{X}_{aug} = \underbrace{[SOC, V_p]}_{\text{原状态}}, \underbrace{[R_0, R_p, C_p]}_{\text{漂移参数}}^T$$

我们要同时估算这 5 个变量.



方法论对比：为何参数辨识需要 UKF？

面对增强后的高维非线性系统，EKF 显得力不从心，而 UKF 展现出降维打击的优势：

EKF 的死穴：求导困难

EKF 需要计算雅可比矩阵 $\frac{\partial h}{\partial X}$.

- 对 R_0 求导很简单.
- 但对 C_p 求导极其痛苦： C_p 位于离散化方程的指数项 $e^{-\Delta t/(R_p C_p)}$ 中，且涉及递归.



推导繁琐，极易出错，数值不稳定.

UKF 的优势：黑盒思维

UKF 不需要计算雅可比矩阵.

- **直接仿真**：根据当前参数 R, C 生成一组 Sigma 点，直接代入电路公式算出预测电压.
- **自动拟合**：算法自动调整参数分布以逼近观测电压.



无需推导公式，仅需物理模型代码.

章节导航

1 提纲

2 背景引入

3 数学模型构建

4 核心算法流程

5 公式推导与证明

6 数值推演实例

7 工程应用与仿真

8 扩展卡尔曼滤波 (EKF)

9 前沿探索

10 结论

11 参考文献

总结

- ① **有效性**: 卡尔曼滤波利用了系统的**冗余信息** (模型预测 + 传感器观测), 从而降低了整体的不确定性.
- ② **高效性**: 递归算法, 只需存储上一刻状态, 非常适合**嵌入式系统及实时控制**.
- ③ **局限性**:
 - 假设系统是线性的 (Linear).
 - 假设噪声是高斯的 (Gaussian).
- ④ **拓展**: 面对非线性系统, 可以使用**扩展卡尔曼滤波 (EKF)**.

章节导航

1 提纲

2 背景引入

3 数学模型构建

4 核心算法流程

5 公式推导与证明

6 数值推演实例

7 工程应用与仿真

8 扩展卡尔曼滤波 (EKF)

9 前沿探索

10 结论

11 参考文献



参考文献 |

- [1] S. F. Schmidt, "The kalman filter: Its recognition and development for aerospace applications," *Journal of Guidance, Control, and Dynamics*, vol. 4, no. 1, pp. 4–7, 1981.
- [2] Y. Bar-Shalom and T. E. Fortmann, *Tracking and Data Association*. Academic Press, 1988.
- [3] R. Smith, M. Self, and P. Cheeseman, "Estimating uncertain spatial relationships in robotics," in *Autonomous robot vehicles*, Springer, 1990, pp. 167–193.
- [4] G. Dissanayake, P. Newman, S. Clark, H. F. Durrant-Whyte, and M. Csorba, "A solution to the simultaneous localization and map building (slam) problem," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 3, pp. 229–241, 2001.
- [5] G. L. Plett, "Extended kalman filtering for battery management systems of lipb-based hev battery packs. part 2. modeling and identification," *Journal of Power Sources*, vol. 134, no. 2, pp. 262–276, 2004.

参考文献 II

- [6] S. J. Julier and J. K. Uhlmann, "New extension of the kalman filter to nonlinear systems," in *Signal processing, sensor fusion, and target recognition VI*, Spie, vol. 3068, 1997, pp. 182–193.
- [7] M. Khodarahmi and V. Maihami, "A review on kalman filter models," *Archives of Computational Methods in Engineering*, vol. 30, no. 1, pp. 727–747, Jan. 2023, Num Pages: 21 Place: Dordrecht Publisher: Springer Web of Science ID: WOS:000862515100001, ISSN: 1134-3060, 1886-1784. DOI: 10.1007/s11831-022-09815-7.
- [8] S. Feng, X. Li, S. Zhang, Z. Jian, H. Duan, and Z. Wang, "A review: State estimation based on hybrid models of kalman filter and neural network," *Systems Science & Control Engineering*, vol. 11, no. 1, p. 2173682, Dec. 31, 2023, Publisher: Taylor & Francis eprint: <https://doi.org/10.1080/21642583.2023.2173682>, ISSN: null. DOI: 10.1080/21642583.2023.2173682. [Online]. Available: <https://doi.org/10.1080/21642583.2023.2173682> (visited on 12/14/2025).

参考文献 III

- [9] X. Li and V. Jilkov, "Survey of maneuvering target tracking. part i: Dynamic models," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1333–1364, 2003.

谢谢大家!
Q & A

