



北京航空航天大学
BEIHANG UNIVERSITY

机器人导航python实践(入门)

Lecture4-同步定位与建图 part2

北航 国新院 实验实践课
智能系统与人形机器人国际研究中心



教师： 欧阳老师



邮箱： ouyangkid@buaa.edu.cn



学期： 2025年秋季

目录

Contents

01 课程内容安排

02 ICP

03 EKF SLAM

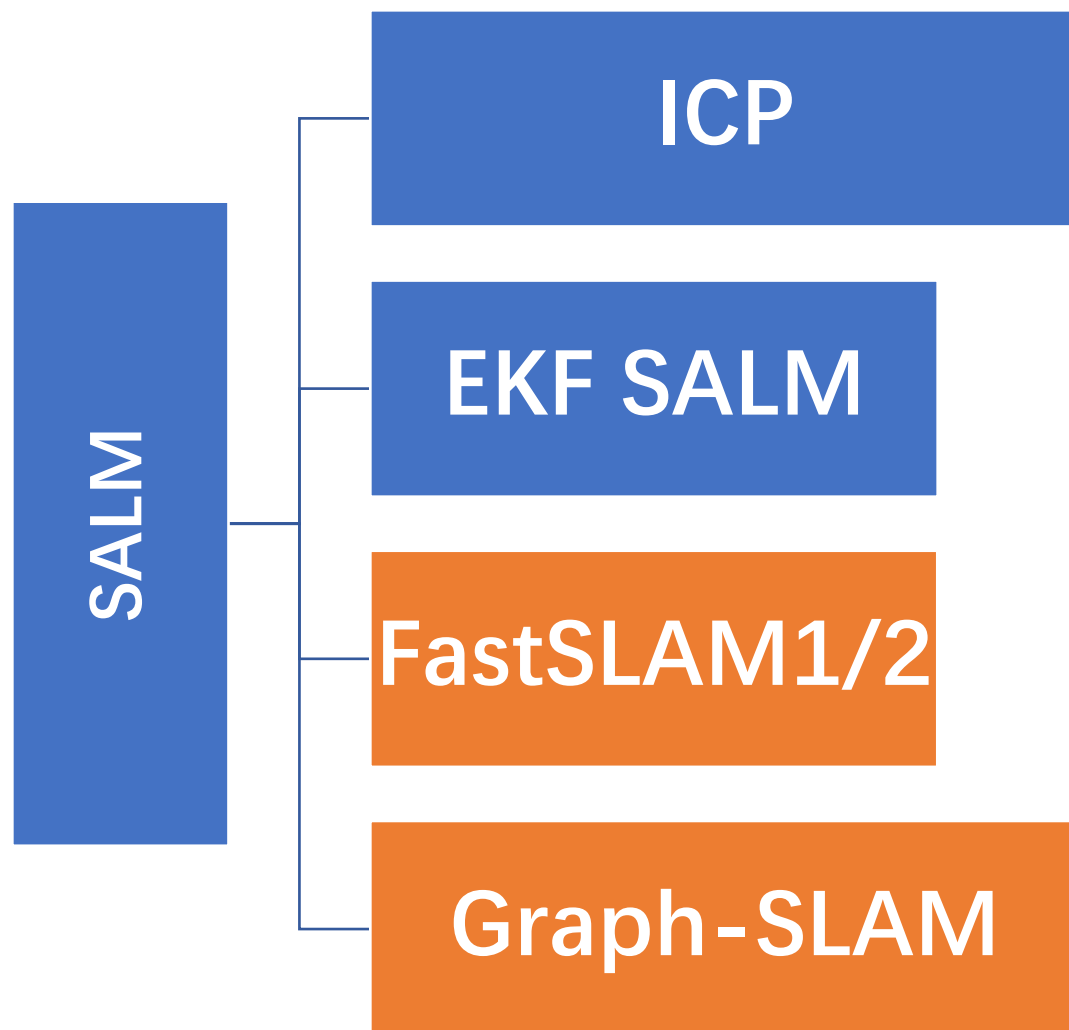
04 FastSLAM 1

05 FastSLAM 2

06 Graph-based SLAM

Part 1 | 课程内容安排

课程内容



目录

Contents

01 课程内容安排

02 ICP

03 EKF SLAM

04 FastSLAM 1

05 FastSLAM 2

06 Graph-based SLAM

Part 4 | SLAM

FastSLAM 1

FastSLAM 基于粒子滤波定位。

回顾： 粒子(分身)滤波通过大量随机点模拟可能的运动状态，再基于RF_ID的定位的信息剔除误差较大的情况，并更新匹配度较高的粒子，使得粒子通过后验概率收敛到真实状态。

包含：预测粒子运动-基于观测更新-重采样，三个过程。



增加对环境特征，也就是RF_ID的位置和不确定性建模。
通过新增信息协同优化位姿（定位）和地图。

其它配置：

- 匀速圆周运动，冷启动，初始静止若干时间；
- 基础对比运动算法：航迹推算；
- 观测状态：[x, y, yaw]

```
class Particle:
```

```
def __init__(self, n_landmark):
```

```
self.w = 1.0 / N_PARTICLE
```

```
self.x = 0.0
```

```
self.y = 0.0
```

```
self.yaw = 0.0
```

```
# landmark x-y positions
```

```
self.lm = np.zeros((n_landmark, LM_SIZE))
```

```
# landmark position covariance
```

```
self.lmP = np.zeros((n_landmark * LM_SIZE, LM_SIZE))
```

原粒子滤波定位状态

新增建图信息（地图状态）



定位任务



同步定位与建图



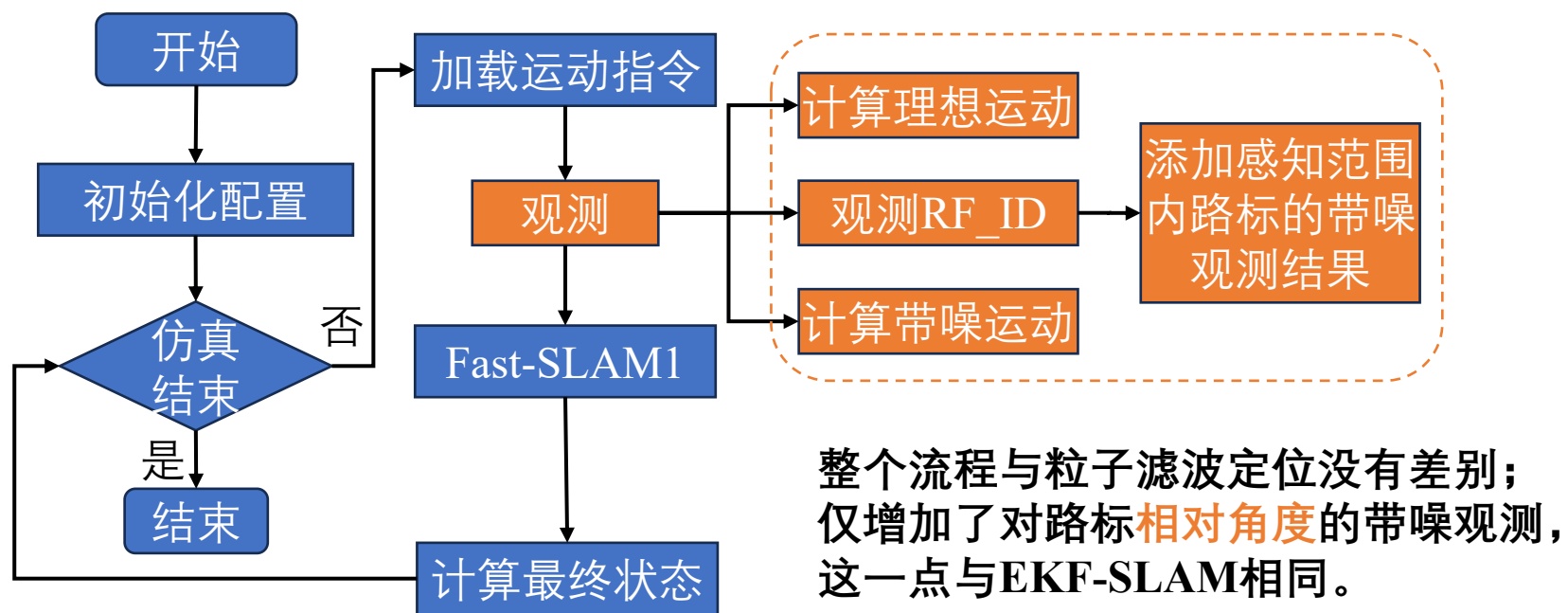
建图任务

环境修改：

- 增加RF_ID分布数量；
- 针对建图需求，粒子类，增加对路标坐标和坐标协方差信息的维护。

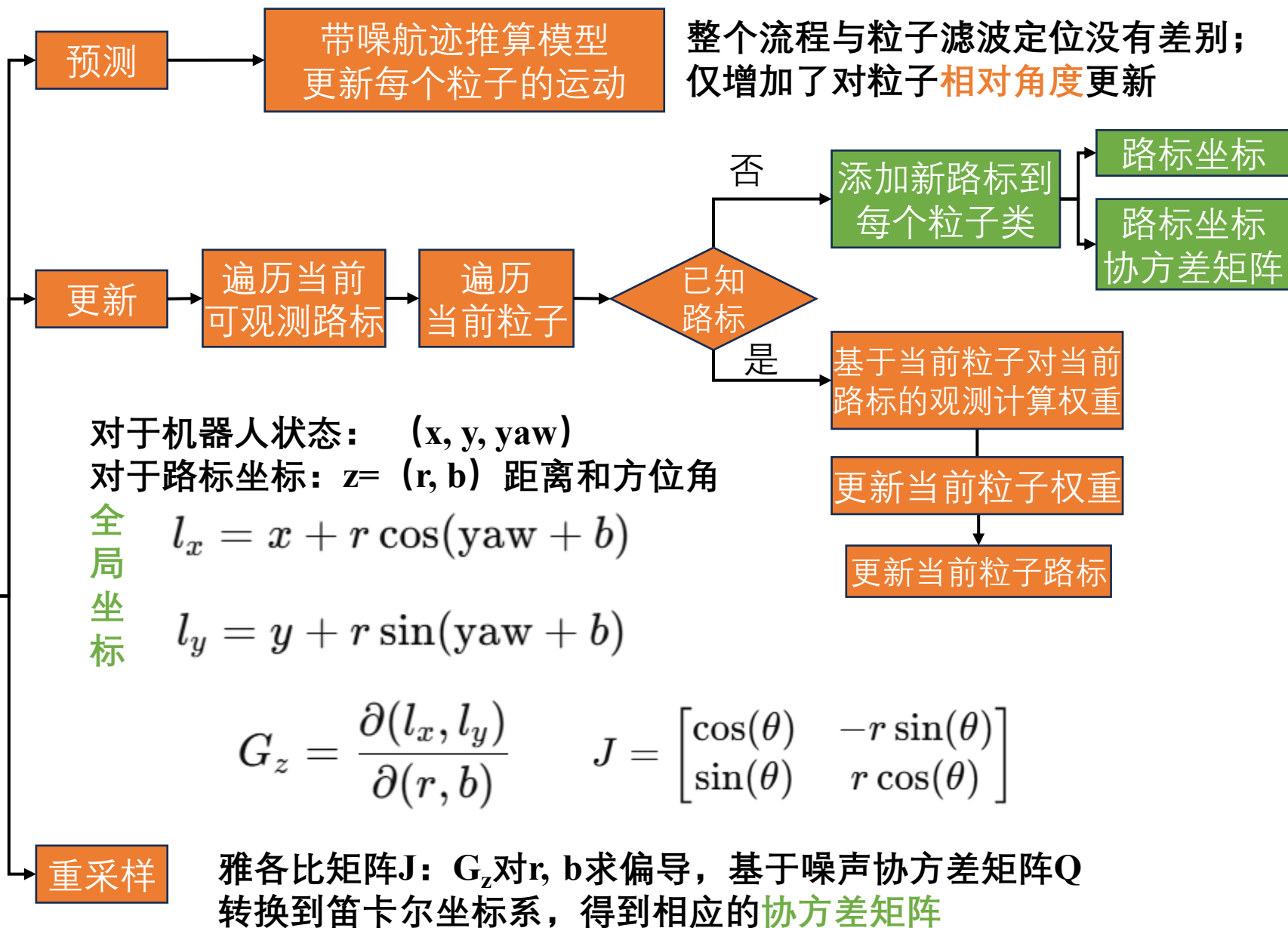
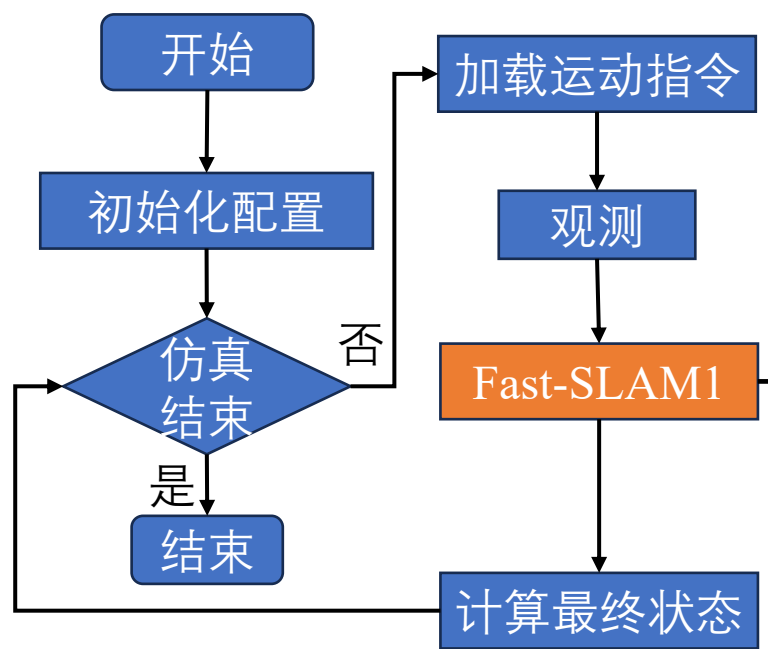
Part 4 | SLAM

FastSLAM 1



Part 4 | SLAM

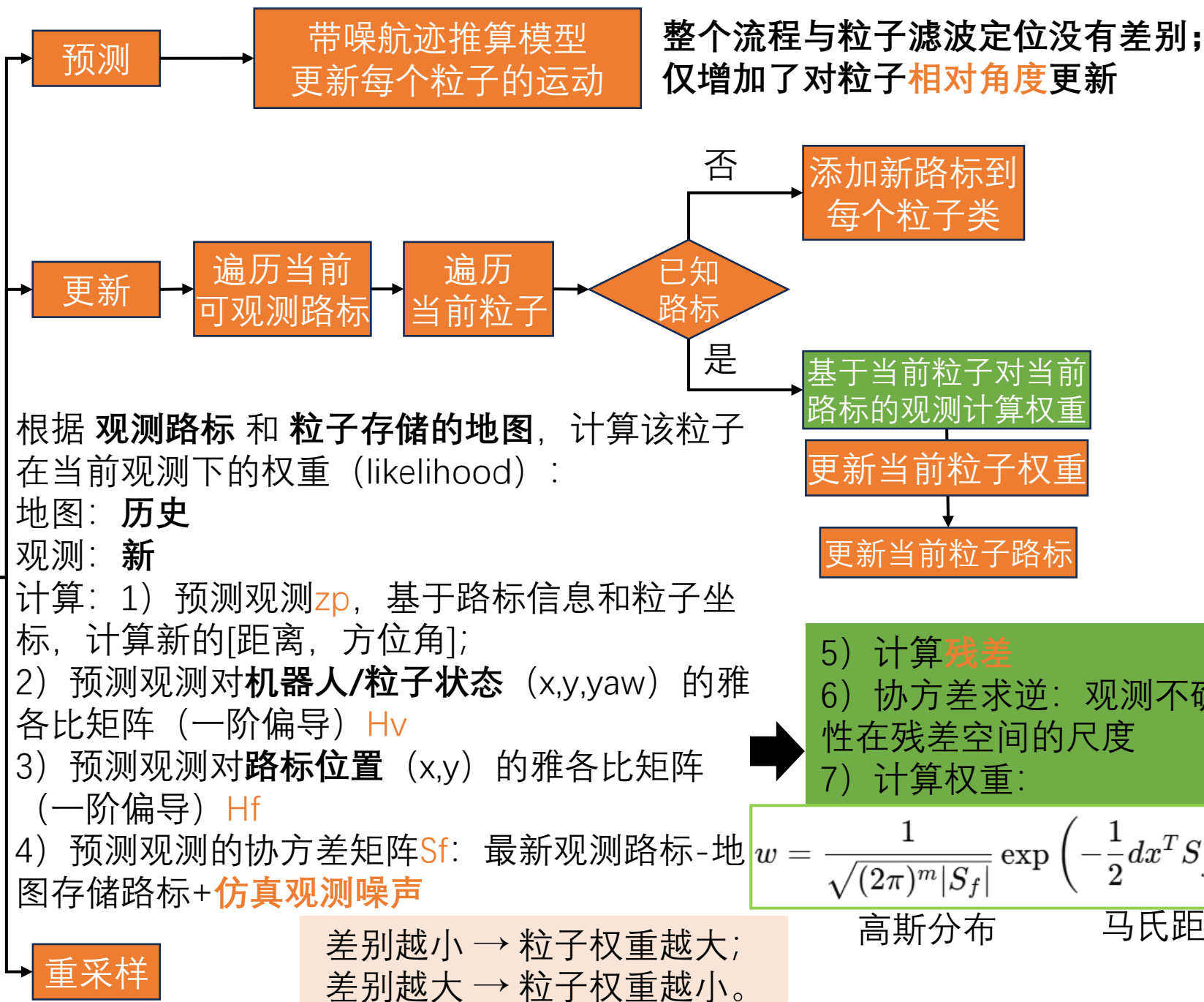
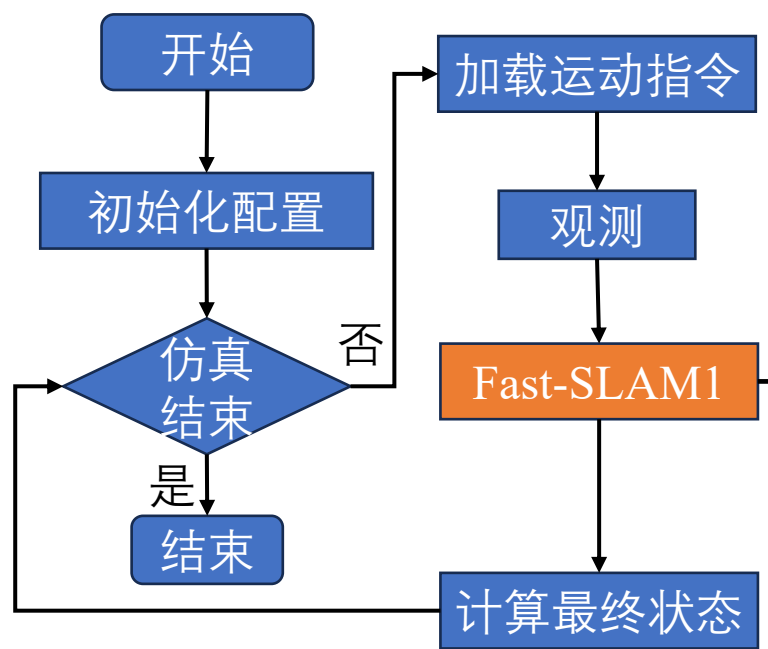
» FastSLAM 1



$$P_{xy} = J Q J^T \quad \text{通过} G_z \text{获得} J^{-1} \text{更容易, 因此 } P = G_z^{-1} Q (G_z^{-1})^T$$

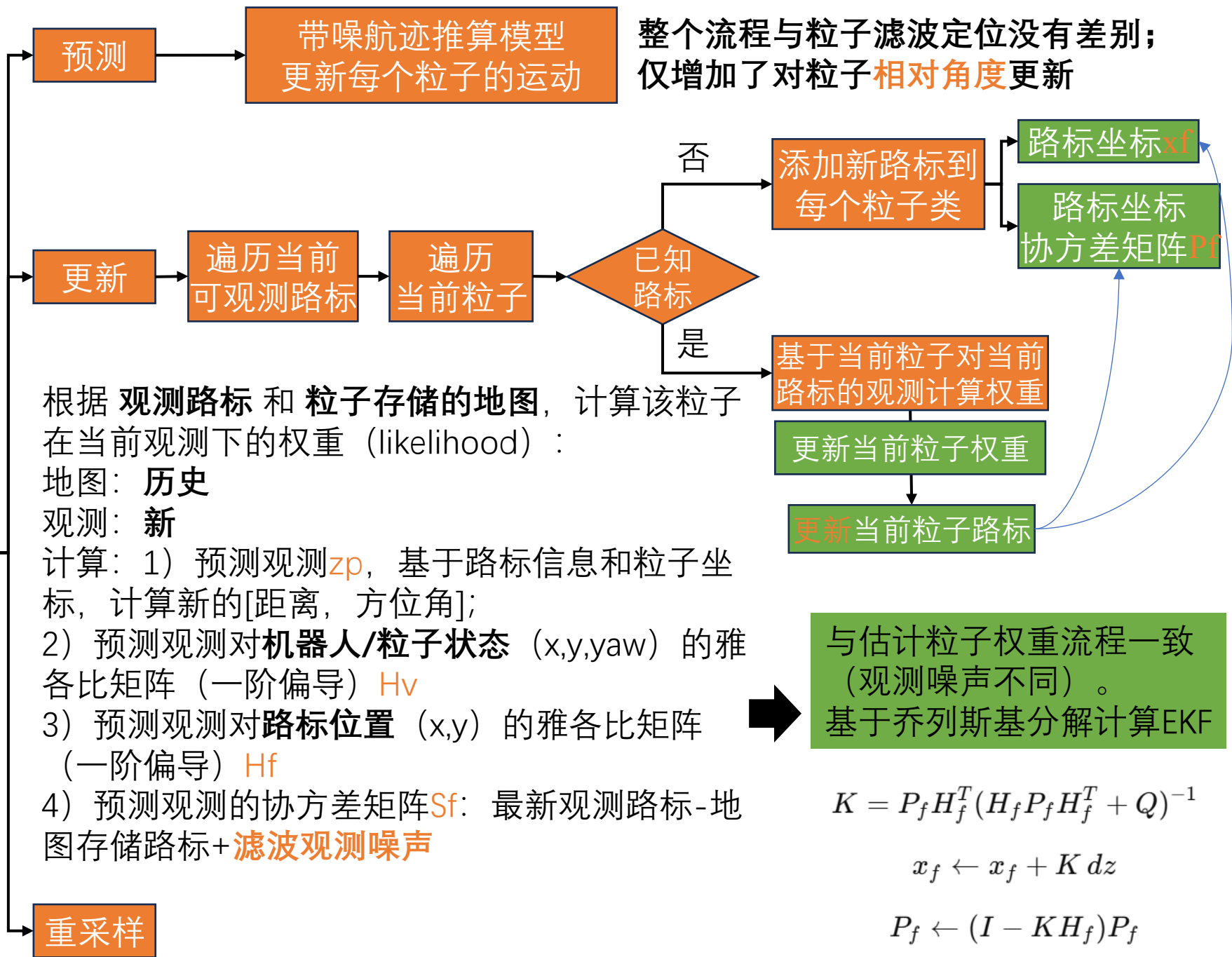
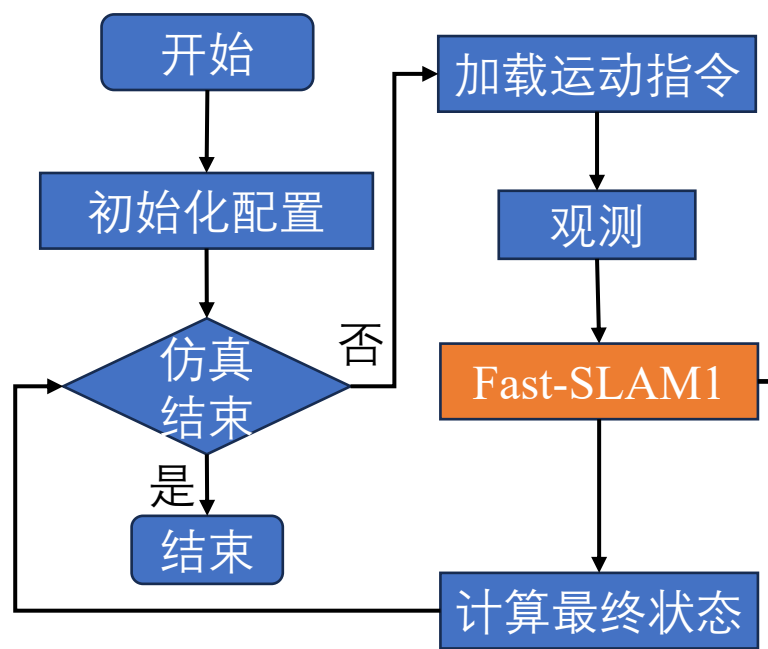
Part 4 | SLAM

» FastSLAM 1



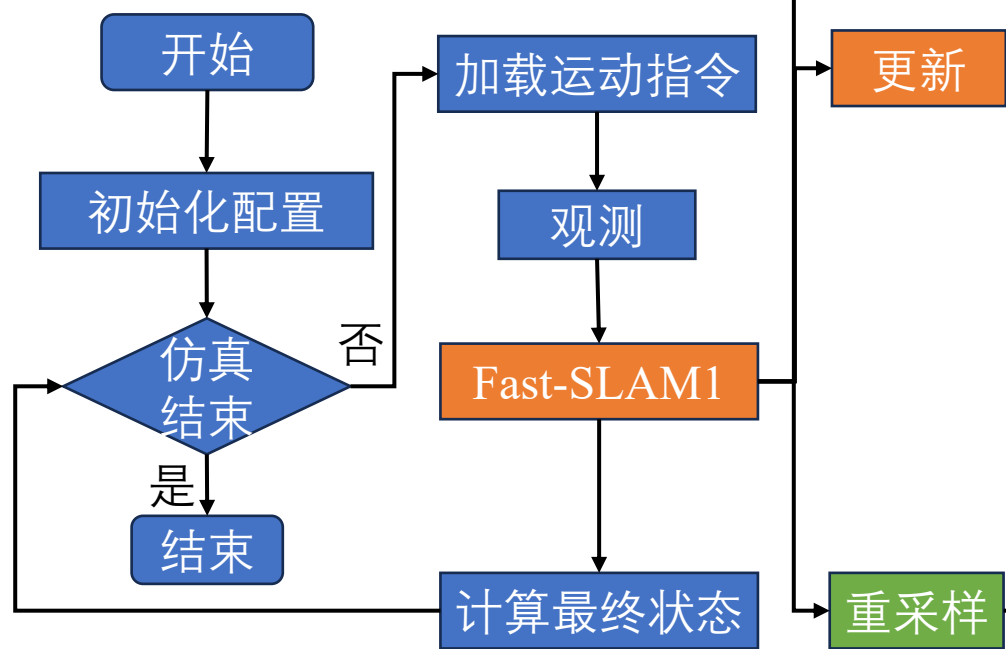
Part 4 | SLAM

FastSLAM 1

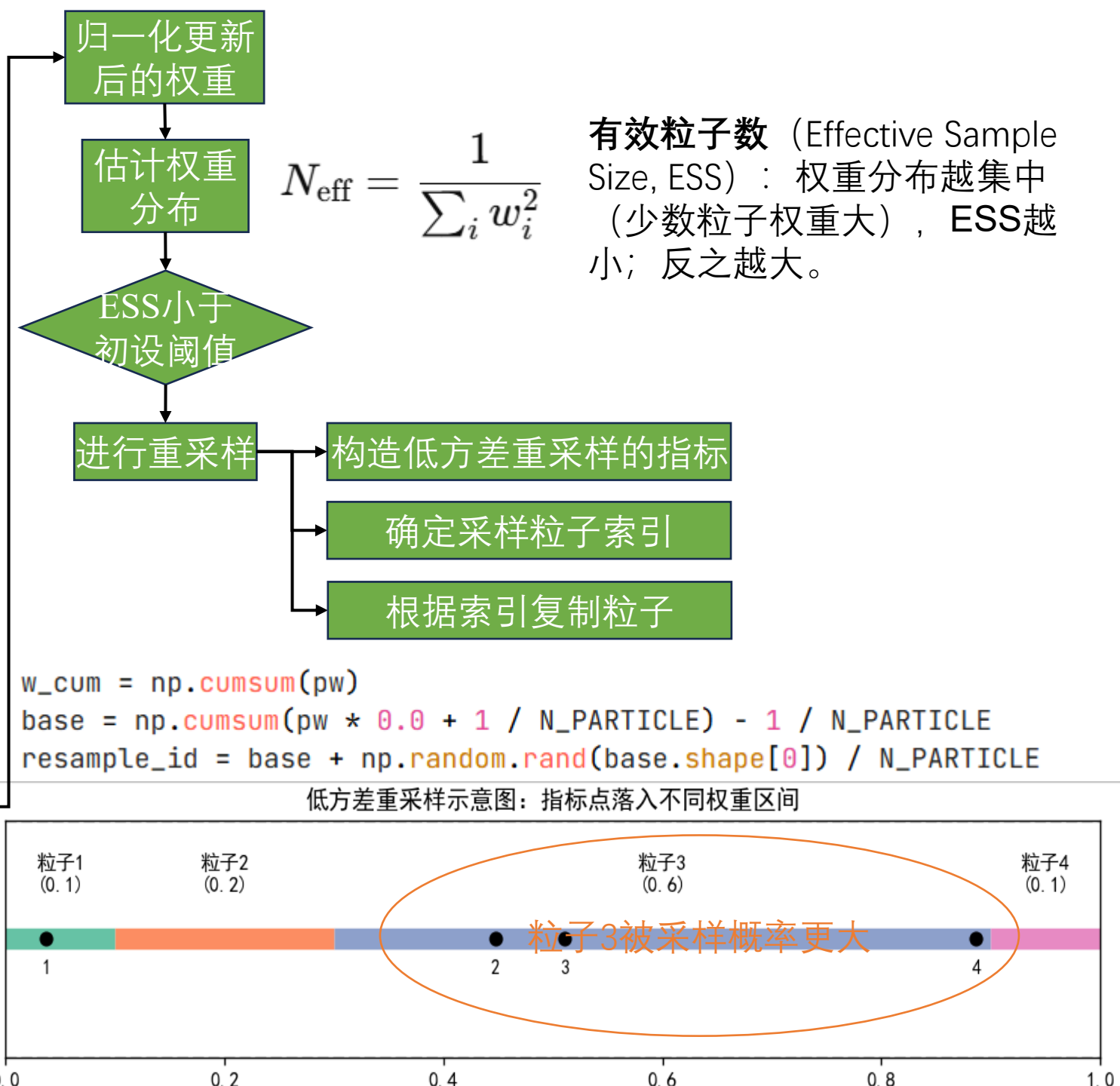


Part 4 | SLAM

» FastSLAM 1

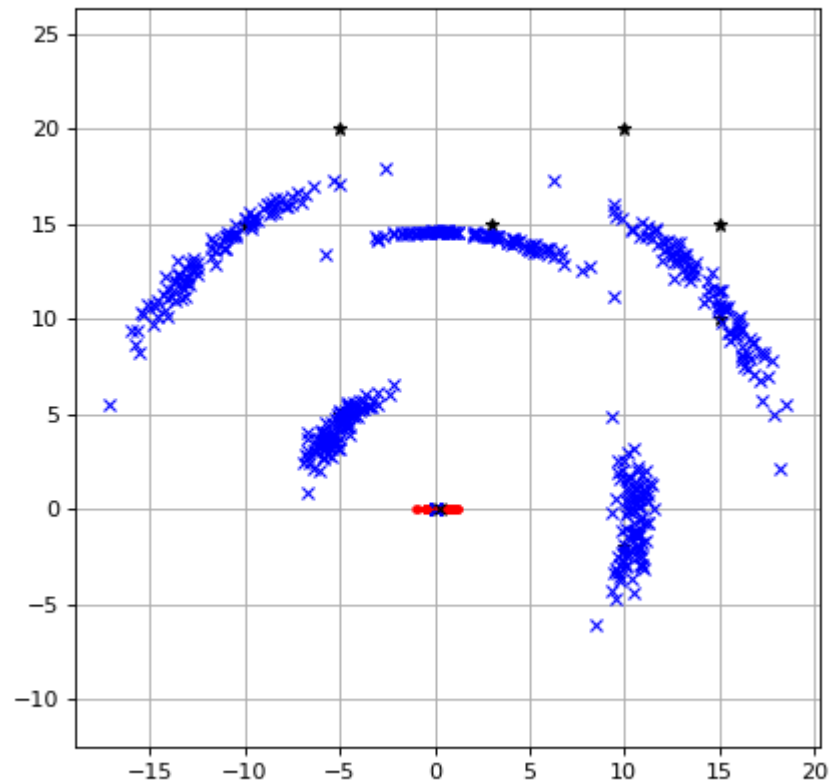
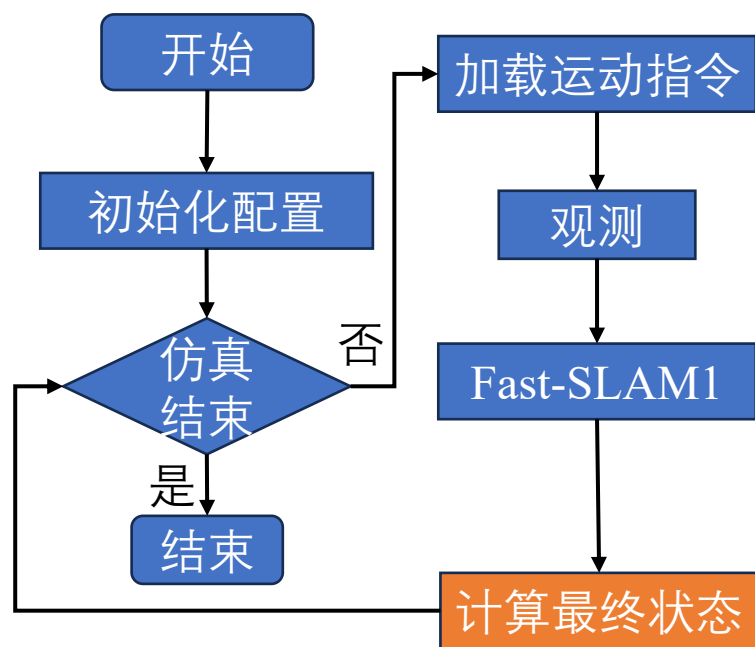


- 复制高权重粒子，淘汰低权重粒子
- 保证粒子滤波中 粒子群不退化，
- 保持多样性和有效粒子数



Part 4 | SLAM

FastSLAM 1



黑*: RF_ID的实际位置

蓝x: 为初始静止过程, 基于观测粒子滤波对观测到的RF_ID的估计位置。并会基于观测更新。

蓝线: GT圆周运动轨迹。

红线: 基于观测的地图 (x) 的定位结果。

黑线: 航迹推算结果。

基于SLAM的地图信息协助估算了权重的更新, 从而优化了粒子权重的计算。

在此之前, 粒子基于卡尔曼滤波和粒子滤波进行了淘汰和筛选。

相当于对机器人最终的位置和朝向的估计是基于所有粒子的加权平均结果。

基于归一化的权重, 更新运动状态估计并对角度进行 $[-\pi, \pi]$ 归一化

$$\hat{x} = \sum_i w_i \cdot x_i, \quad \hat{y} = \sum_i w_i \cdot y_i, \quad \hat{\theta} = \sum_i w_i \cdot \theta_i$$

实验: 分析lesson1中粒子滤波的四种噪声协方差定义, 如何调整参数, 使对路标的定位 (建图) 跟精准。

目录

Contents

01 课程内容安排

02 ICP

03 EKF SLAM

04 FastSLAM 1

05 FastSLAM 2

06 Graph-based SLAM

Part 5 | SLAM

FastSLAM 2

算法差异

特征	FastSLAM 1.0	FastSLAM 2.0
粒子状态协方差	无	自身状态不确定性P
运动模型	仅运动模型	结合运动模型和观测 (proposal_sampling)
粒子退化抵抗能力	较弱	观测信息辅助采样增强

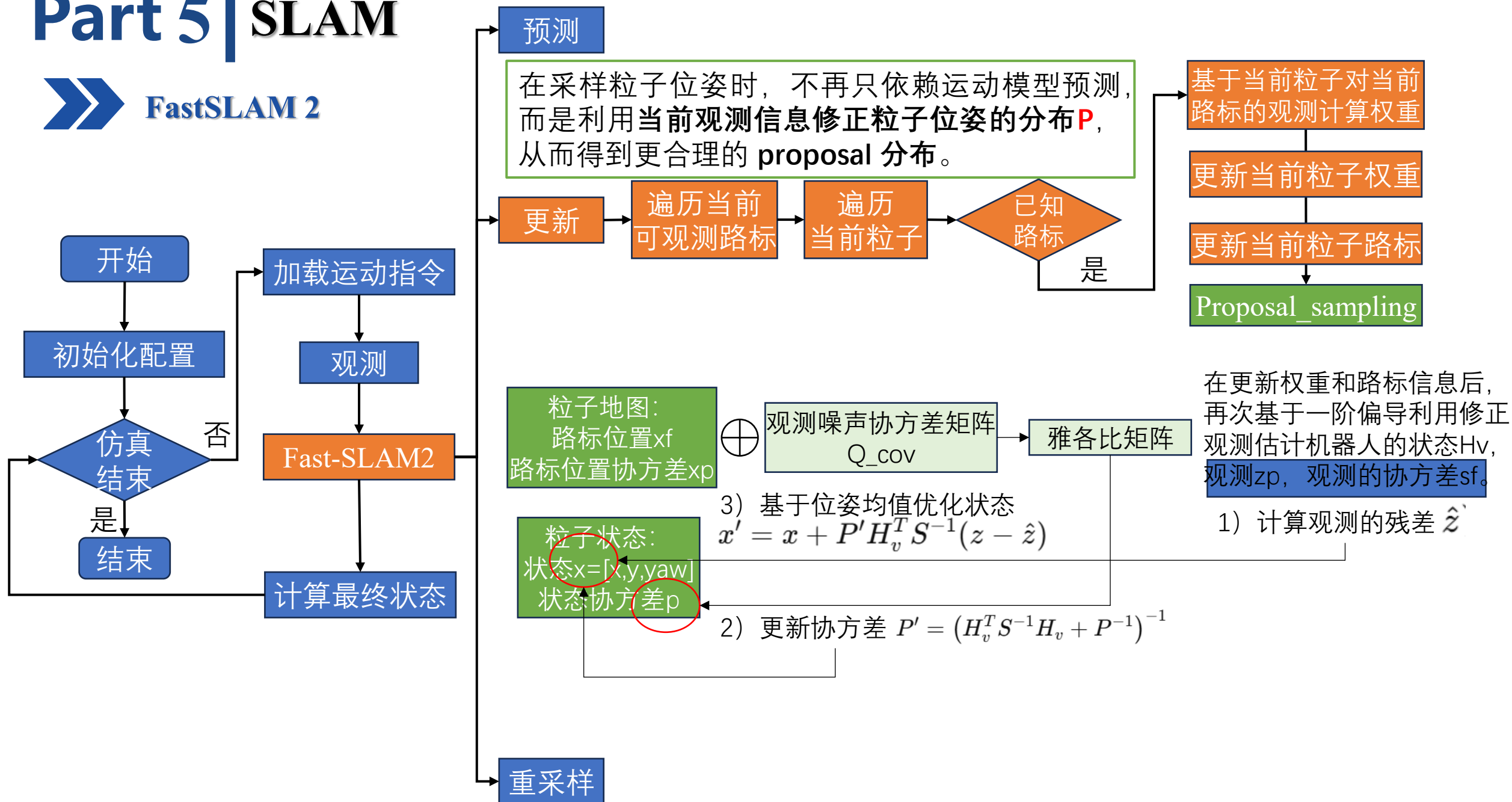
结合观测信息
更新粒子位姿的分布
(proposal)

```
def __init__(self, n_landmark):
    self.w = 1.0 / N_PARTICLE
    self.x = 0.0
    self.y = 0.0
    self.yaw = 0.0
    self.P = np.eye(3)
    # landmark x-y positions
    self.lm = np.zeros((n_landmark, LM_SIZE))
    # landmark position covariance
    self.lmP = np.zeros((n_landmark * LM_SIZE, LM_SIZE))
```

粒子新增了自身状态的协方差矩阵 P（维度 3x3，对应 x, y, yaw 的不确定性），用于后续的提议分布采样。

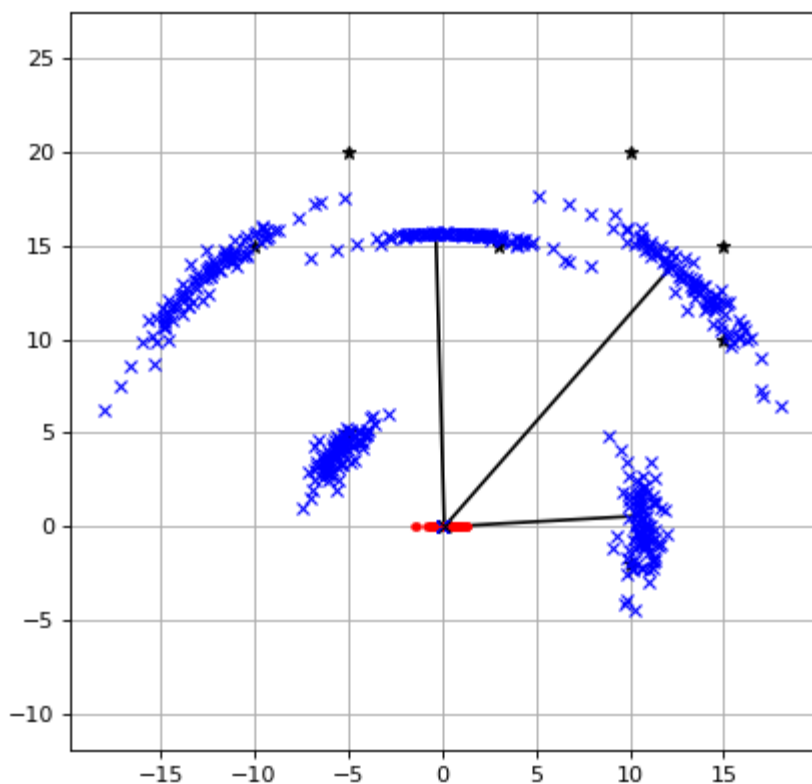
Part 5 | SLAM

FastSLAM 2



Part 5 | SLAM

FastSLAM 2



黑*: RF_ID的实际位置

蓝x: 为初始静止过程，基于观测粒子滤波对观测到的RF_ID的估计位置。并会基于观测更新。

蓝线: GT圆周运动轨迹。

红线: 基于观测的地图 (x) 的定位结果。

黑曲线: 航迹推算结果。

黑直线: 当前可被观测到的路标 (最优情况)。

实验：将两个算法在一个窗口进行可视化对比。可以尝试调整噪声参数。

使用相同的仿真配置，对比FastSLAM1和FastSLAM2的时间复杂度（理论+实际），定位平移误差，方位角误差。

1.使用绝对位姿误差，即与坐标原点的相对值进行度量。2.时间复杂度实际值给出计算硬件参数。3.关闭可视化。4.可视化

目录

Contents

01 课程内容安排

02 ICP

03 EKF SLAM

04 FastSLAM 1

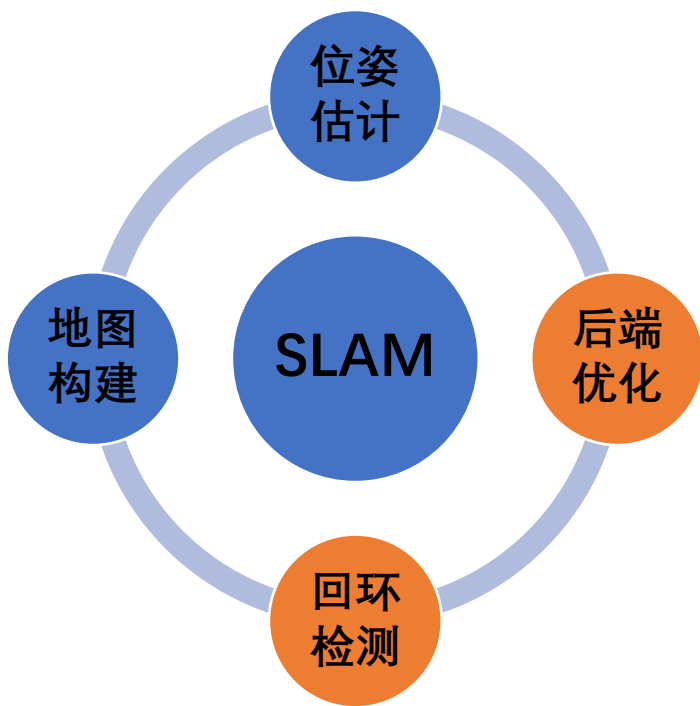
05 FastSLAM 2

06 Graph-based SLAM (选修)

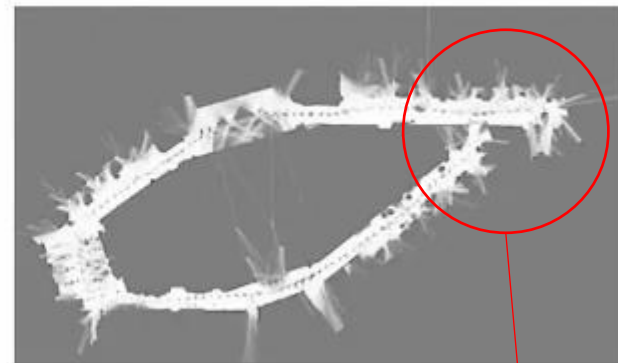
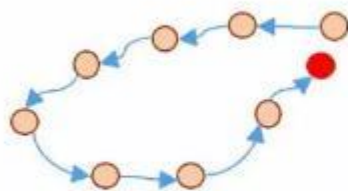
Part 5 | SLAM

Graph SLAM

图优化一般在SLAM的后端 (back-end)



回环 (Closed-loop) 检测是指系统识别出当前观测到的场景或环境区域与过去已经访问过的区域是同一位置的过程。

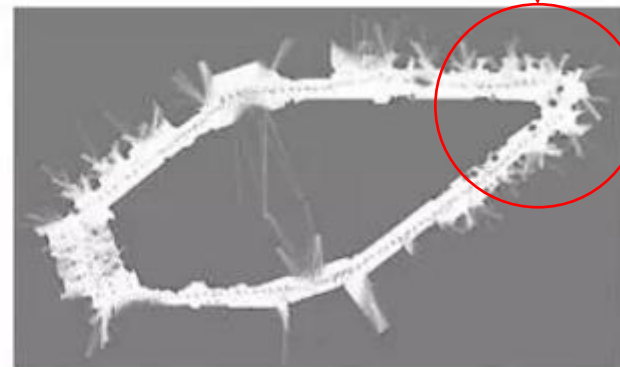
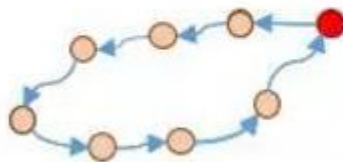


闭环

两个距离，空间距离 和 累积距离：

在（里程计）估计中，起始点之间的累积距离较远，也就是从A经过x米到达B；

在坐标系下，起始点之间的空间距离很近，也就是A到B之间的直线距离很近。

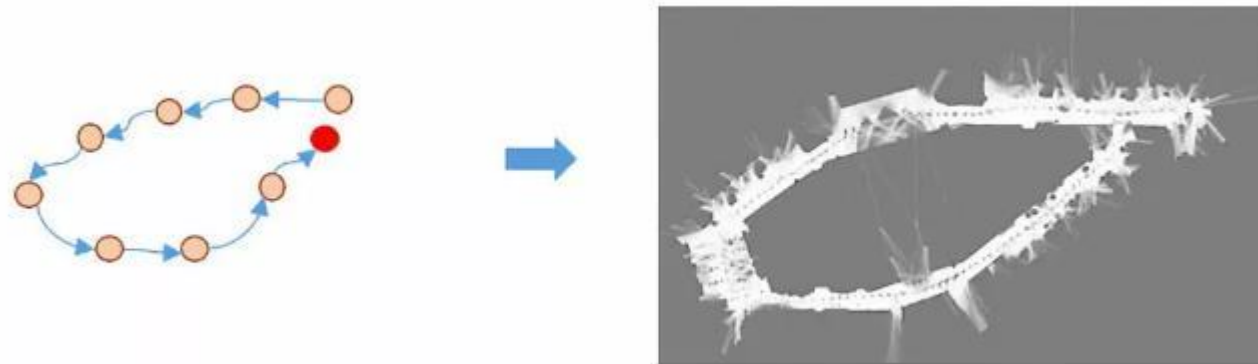
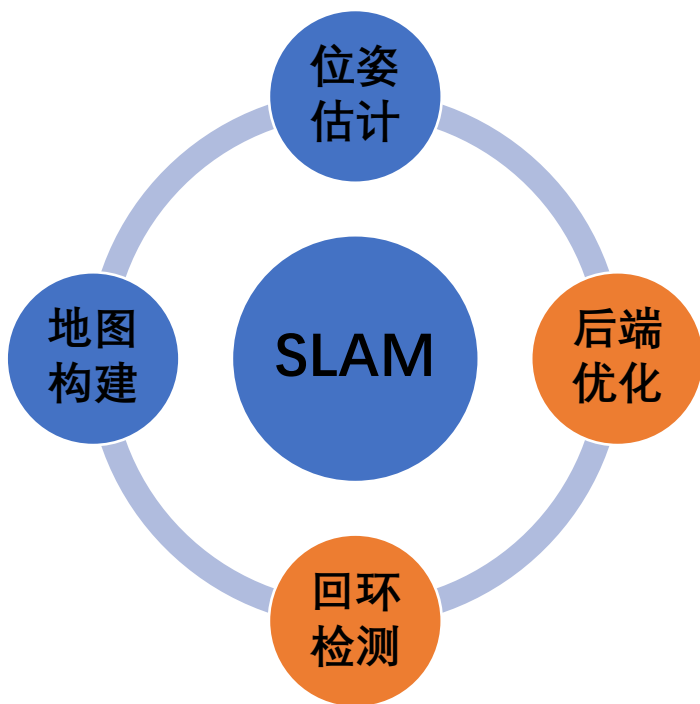


实际上，这一条件也不是在任何情况下都通用，当累积误差过大时，仅依赖空间距离判断就会失效。

Part 5 | SLAM

Graph SLAM

图优化一般在SLAM的后端 (back-end)



基于图对SLAM过程进行建模，每个采样的位姿构成节点 (**node**)，节点之间的边 (**edge**) 代表传感器位姿的约束关系。



找到节点的一种空间配置 (即确定各节点的位置与姿态)，使其与所有测量数据的一致~~性~~达到最高。

数学描述: 1) 移动轨迹: $x_{1:T} = \{x_1, \dots, x_T\}$

2) 里程估计: $u_{1:T} = \{u_1, \dots, u_T\}$

3) 环境观测: $z_{1:T} = \{z_1, \dots, z_T\}$

条件先验-机器人轨迹: $P(x_{1:T} | z_{1:T}, u_{1:T})$

基于初始位置 x_0 的地图 m : $P(x_{1:T}, m | z_{1:T}, u_{1:T}, x_0)$

后验

Part 5 | SLAM

Graph SLAM

对后验概率的估计依赖高维状态空间，并需要SLAM问题建模具有结构化表征。

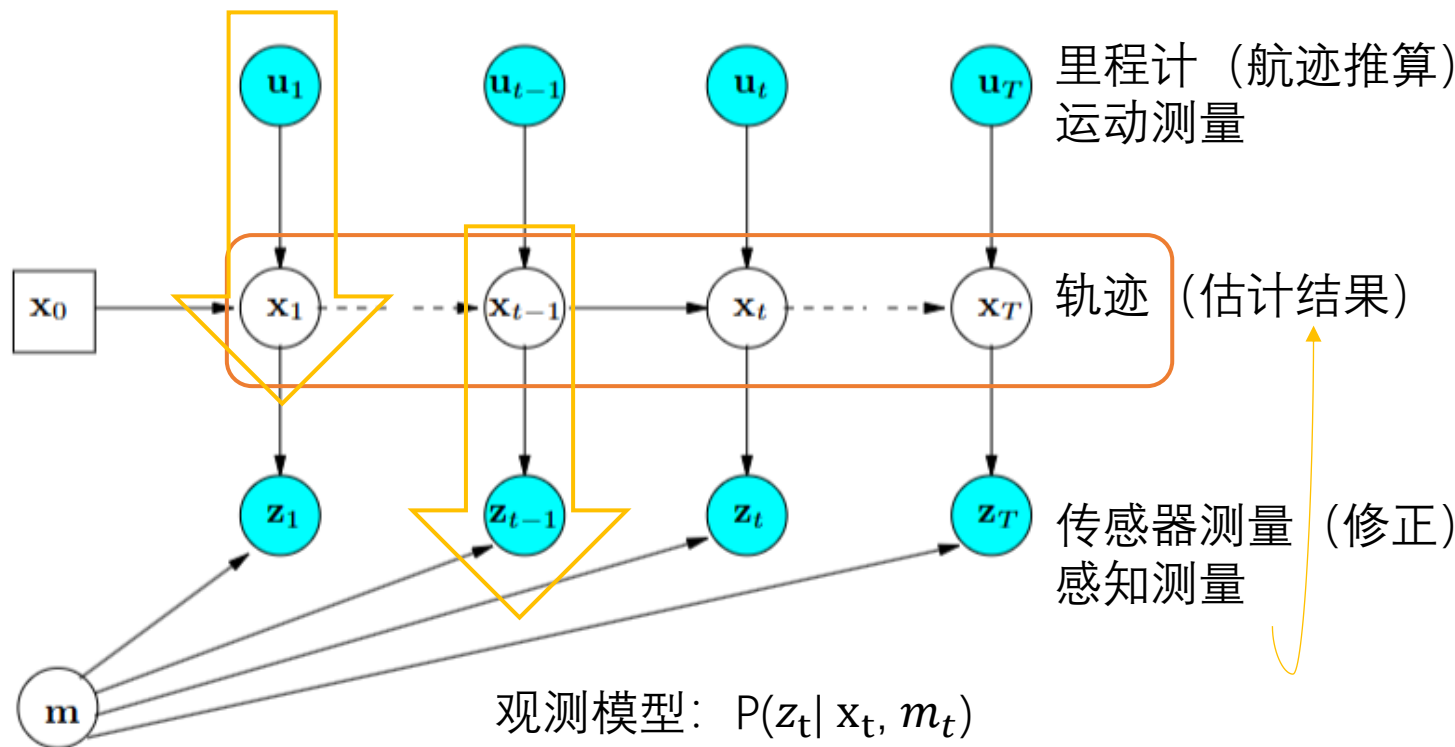
$$P(x_{1:T}, m | z_{1:T}, u_{1:T}, x_0)$$

实际主要采用动态贝叶斯网络=贝叶斯网络 (BN) +动态 (Dynamic) 调整的节点和边。其中，BN是有向图，描述了节点之间的**条件依赖**。DBN=状态转移模型+观测模型。

$$P(x_t | x_{t-1}, u_t)$$

状态转移模型，当前的位姿 x_t ，从上一时刻位姿 x_{t-1} 和当前时刻的里程估计 u_t 的转移过程。

描述DBN的动态增长过程的空间结构，相关算法也被称为graph-based算法。边描述了位姿相对变换的概率分布：机器人连续位置间的里程测量 ($u \rightarrow x$) / 对齐两个位置观测配准 $z(t) \rightarrow z(t-1)$



构建上述图结构的过程，被称为**建图**。前面的课程中所描述的算法，主要在进行此过程。

图构建完成之后，需要找到能够满足所有约束的机器人的位姿配置。此过程，也被称为**图优化/后端优化**。通常基于误差最小化，如最小二乘等进行求解。

后端构建的边并不像轨迹，是顺序和成对的。

Part 5 | SLAM

Graph SLAM

节点 (nodes / vertices)

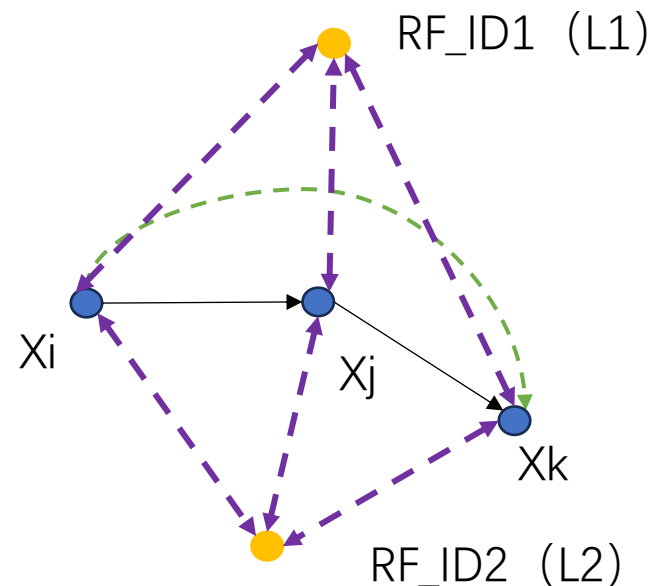
每个节点 x_i 表示机器人在某一时刻的 **位姿** (位置+朝向)。

如果有RFID标签, 也可以把标签的位置当作一个固定的节点 l_j 。

边 (edges / constraints)

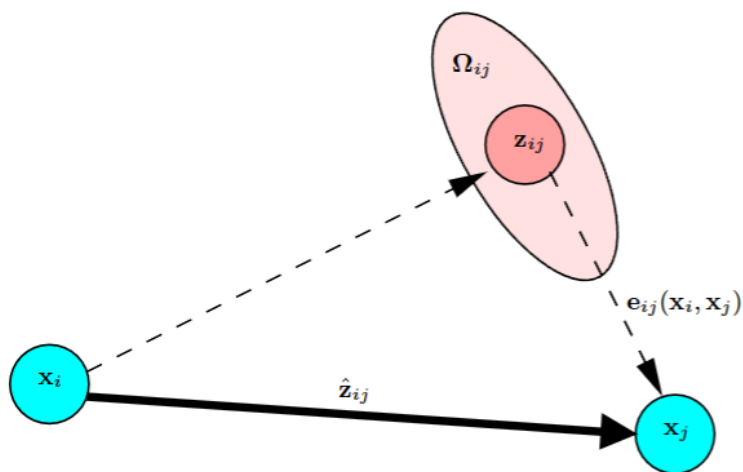
边代表“约束条件”, 也就是机器人运动或传感器测量带来的“相对位置信息”。

- **里程计 (odometry) 约束**: 连接 x_{t-1} 和 x_t , 表示机器人移动的估计。
- **观测约束**: 连接机器人位姿 x_t 和标签位置 l_j , 表示“机器人在 x_t 时刻观测到标签 j , 测量得到一个距离/方位”。



Part 5 | SLAM

Graph SLAM



z_{ij} : 在 x_i 位置观测到标签 l_j 的测量值, 比如“距离 2.3m, 角度 45° ”。

$\hat{z}_{ij}(x_i, l_j)$: 如果已知 x_i, l_j , 理论上应该观测到的测量值 (预测值)。

$e_{ij}(x_i, l_j) = z_{ij} - \hat{z}_{ij}(x_i, l_j)$: **误差函数**, 表示真实观测和预测值的差距。

Ω_{ij} : 信息矩阵, 表示该观测的可靠程度 (噪声越小, 权重越大)。

$$x^*, l^* = \arg \min_{x, l} \sum_{(i, j)} e_{ij}^T \Omega_{ij} e_{ij}$$

找到一组机器人轨迹 x 和标签位置 l , 使得所有观测误差加权平方和最小。

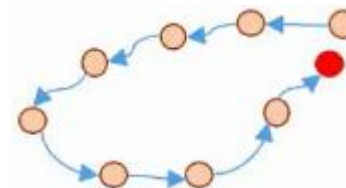
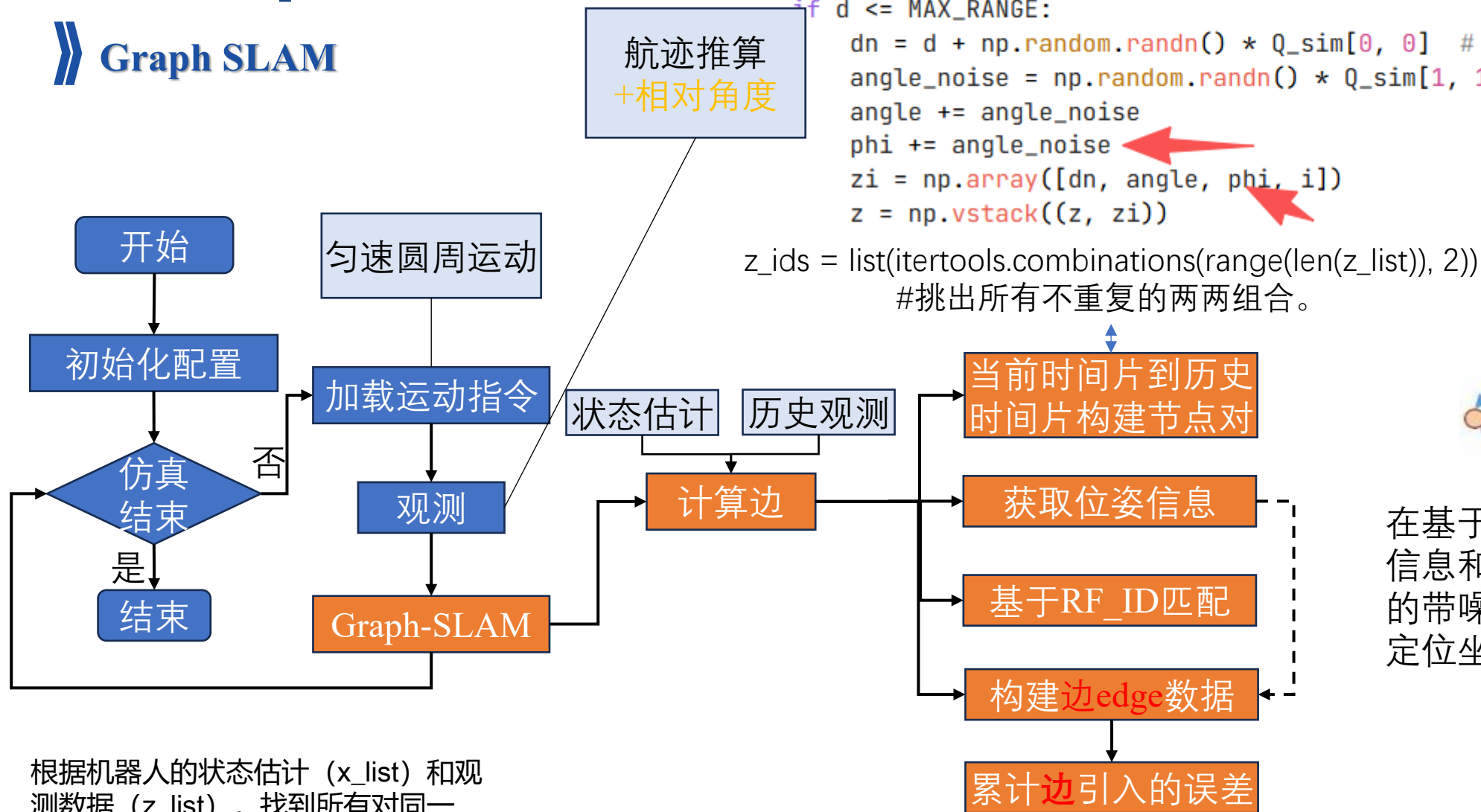
把所有边累加成稀疏矩阵 H, b 。

解线性方程 $H \Delta x = -b$, 更新位姿。 $dx = -H^{-1} \cdot b$

迭代至收敛。

Part 5 | SLAM

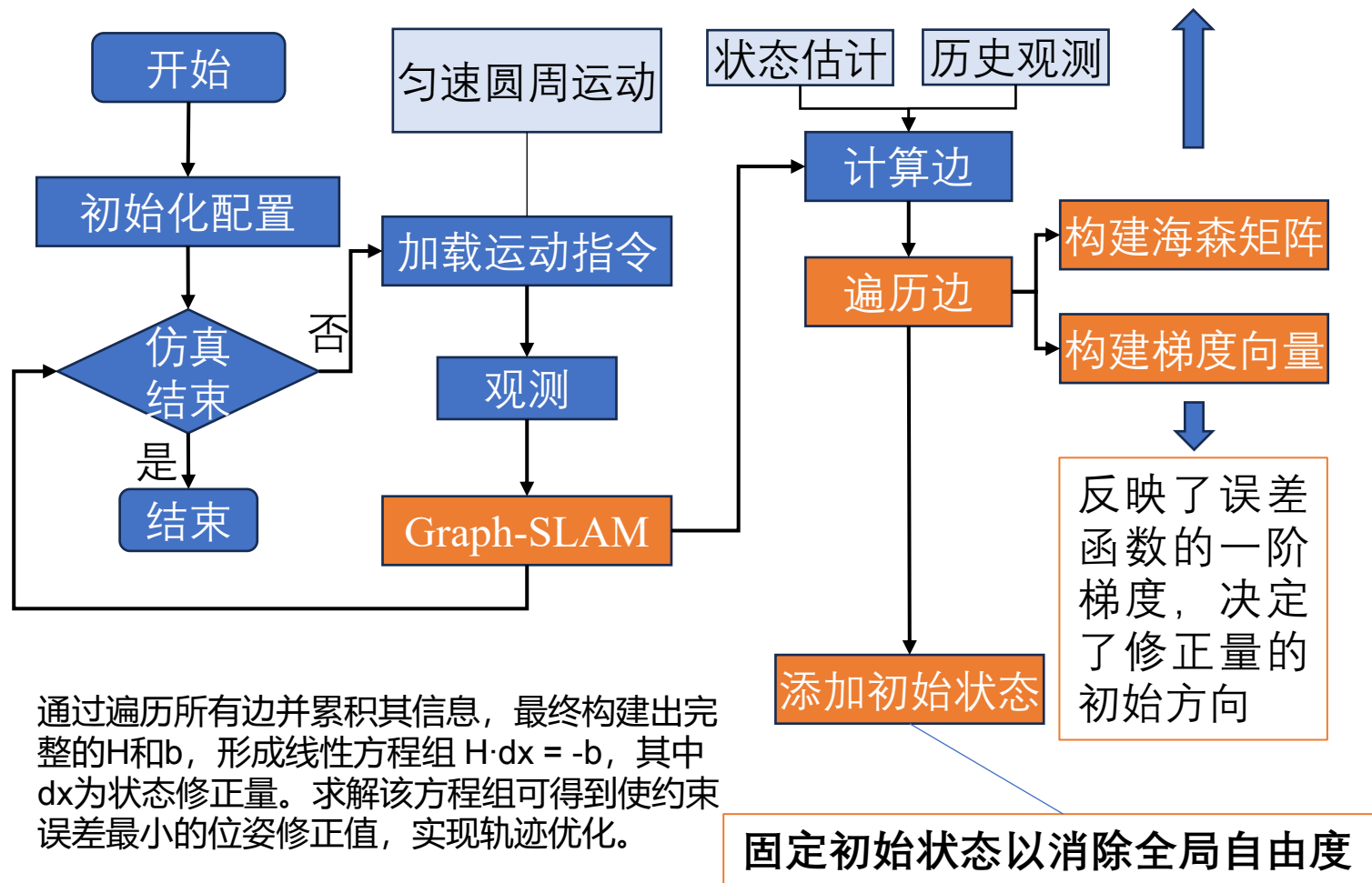
》 Graph SLAM



在基于RF_ID的定位中，观测信息和地图就是Agent获取到的带噪定位坐标，以及所有定位坐标组合。

Part 5 | SLAM

》 Graph SLAM



将边的约束关系，通过雅各比矩阵（两个节点的）转化为对海森矩阵和梯度向量的增量更新。

节点1的二阶导；
节点1对节点2的影响；
节点2对节点1的影响；
节点2的二阶导；

// compute the nonzero Hessian blocks

$$\begin{aligned}\tilde{H}_{[ii]} &+= \tilde{A}_{ij}^T \Omega_{ij} \tilde{A}_{ij} & \tilde{H}_{[ij]} &+= \tilde{A}_{ij}^T \Omega_{ij} \tilde{B}_{ij} \\ \tilde{H}_{[ji]} &+= \tilde{B}_{ij}^T \Omega_{ij} \tilde{A}_{ij} & \tilde{H}_{[jj]} &+= \tilde{B}_{ij}^T \Omega_{ij} \tilde{B}_{ij}\end{aligned}$$

通过雅可比矩阵的转置、信息矩阵和误差向量（edge.e）的乘积，累积约束对状态变量的一阶误差影响。

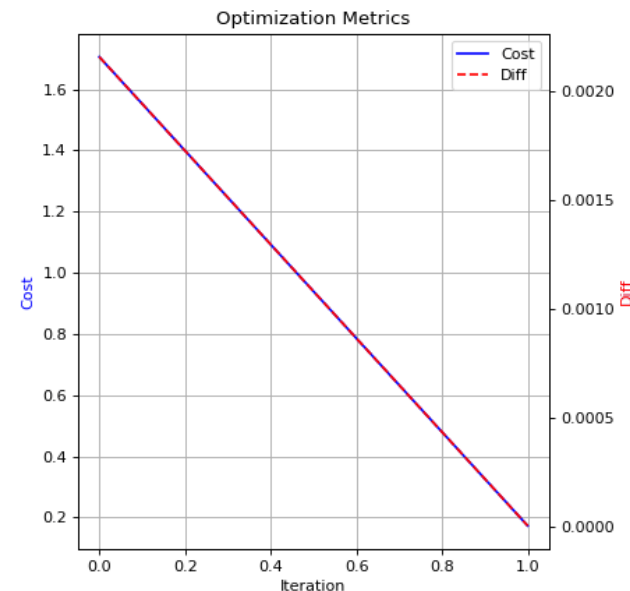
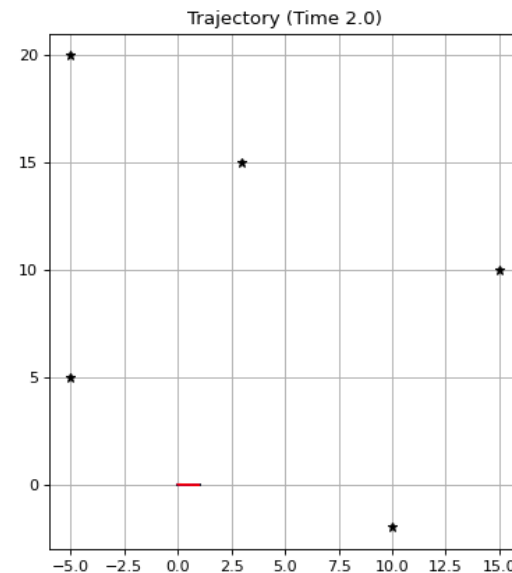
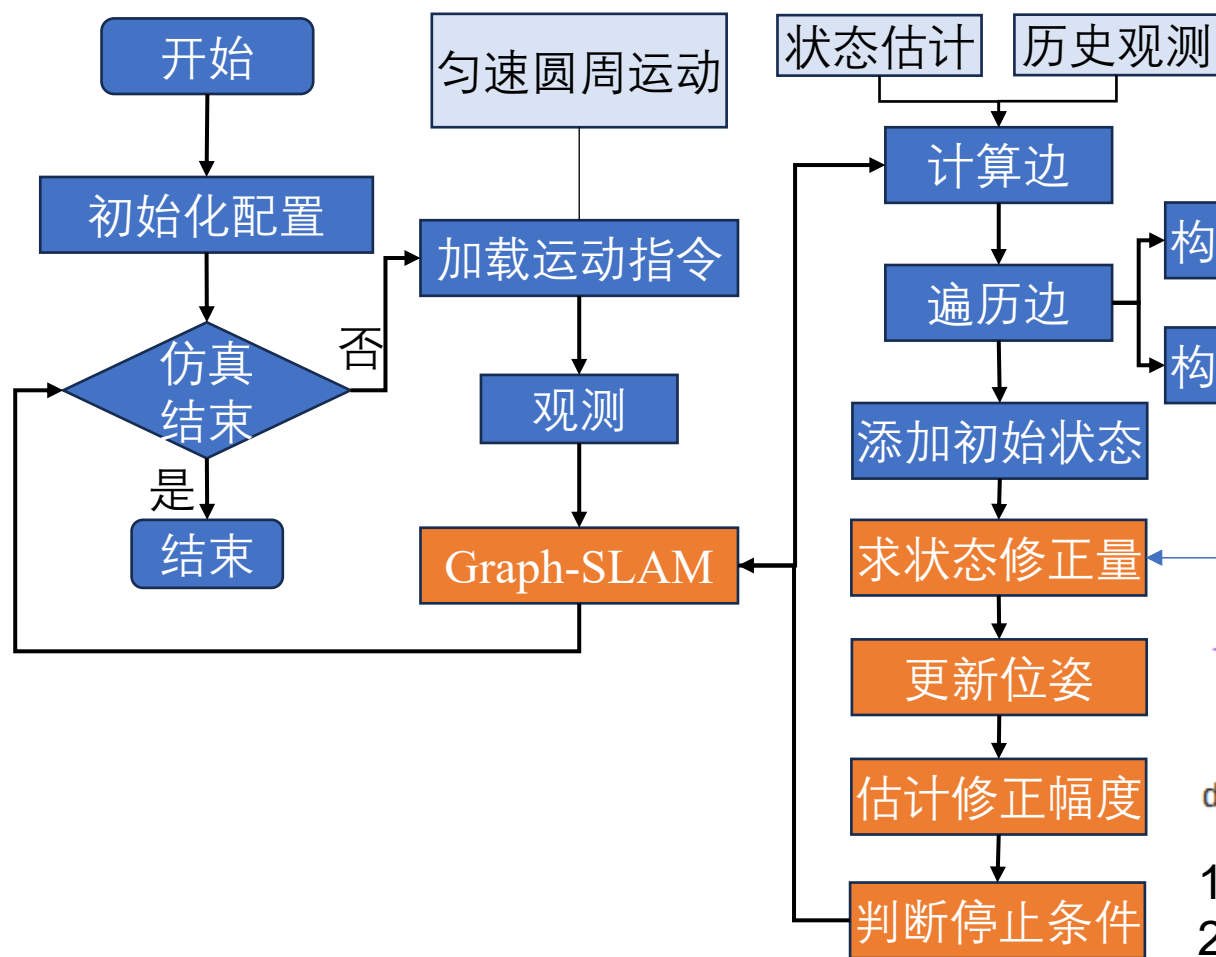
节点1的一阶误差项；
节点2的一阶误差项；

// compute the coefficient vector

$$\tilde{b}_{[i]} += \tilde{A}_{ij}^T \Omega_{ij} e_{ij} \quad \tilde{b}_{[j]} += \tilde{B}_{ij}^T \Omega_{ij} e_{ij}$$

Part 5 | SLAM

» Graph SLAM



$$dx = - \text{np.linalg.inv}(H) @ b$$

```
for i in range(nt):
    x_opt[0:3, i] += dx[i * 3:i * 3 + 3, 0]
```

```
diff = (dx.T @ dx)[0, 0]
```

1. 基于最大迭代轮次 控制优化轮次上限
2. 基于修正幅度阈值 早停

开始实验



北京航空航天大学
BEIHANG UNIVERSITY