**Fast SLAM**

Fast SLAM algorithm is introduced by Montemerlo et al. in 2002 as first successful implementation of Rao-Blackwellised particle filter that could handle large maps or real-world problems. Each landmark is represented by 2x2 EKF, therefore each particle must maintain M individual EKFs. In total, there are N·M EKFs, where M is the total number of particles in the particle filter and N is the total number of landmarks.

**Key Steps of Fast SLAM 1.0**

Fast SLAM algorithm draws samples according to standard odometry model being used to localization. It extends the path posterior by sampling a new pose for each sample.

In the next step, it computes the importance weight:

Q: measurement covariance

z: current observation

ẑ: expected observation (calculated for each individual)

As last step, it updates the belief of observed landmarks using the EKF update rule, then resamples using the standard resampling operation.

**Computational Complexity Fast SLAM Implementation**

Update robot particles: *O(N)*

Incorporate an observation into Kalman filters: *O(N log M)*

Resample particle set: *O(N log M)*

Total: *O(N log M)*

(*where N is the number of particles and M is the number of map features*)

**Fast SLAM 2.0**

Second iteration of Fast SLAM proposed by Montemerlo et al. in 2003, which considers the measurements during the sampling.

This leads to proposal distribution being more peaked around the true state where the system is in. As a result, less samples are needed. Compared to Fast SLAM 1.0, Fast SLAM 2.0 is more robust and accurate, however it is more complex.