SLAM Techniques and Algorithms

Jack Collier

Defence Research and Development Canada
Recherche et développement pour la défense Canada

Goals

What will we learn

- Gain an appreciation for what SLAM is and can accomplish
- Understand the underlying theory behind SLAM
- Understand the terminology and fundamental building blocks of SLAM algorithms
- Appreciate the deficiencies of SLAM and SLAM algorithms
- Won't focus on the math, but the concept
- Online non-linear feature based SLAM with unknown data association

Outline

- What is SLAM?
- Probabilistic basis of SLAM
- EKF SLAM
- Data Association
- Closing the Loop
- FastSLAM
- Sub-Mapping SLAM
- Resources
- Questions?

What is SLAM?

- Simultaneous Localization and Mapping
- Given an unknown environment and vehicle pose:
  - Move through the environment
  - Estimate the robot pose
  - Generate a map of environmental features
- Use the robot pose estimate to improve the map landmark position estimates
- Use the landmark estimates to improve the robot pose estimate

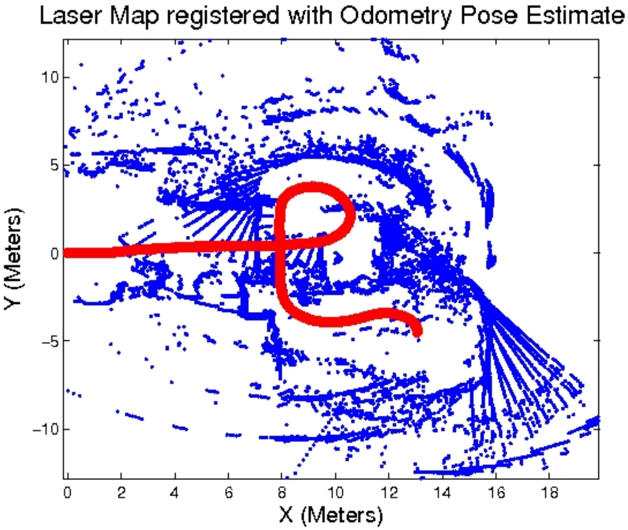
What is SLAM? and why do we need it?

- Robot motion models aren't accurate
- Wheel odometry error is cumulative
- IMU sensors errors

Maps are generated from sensors on the vehicle. If we can't accurately predict the robot pose then how can we produce an accurate map.

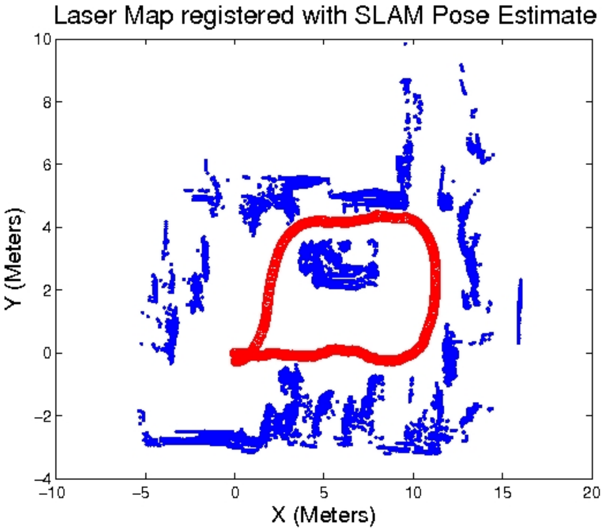
GPS can help but is not always available or reliable.

What is SLAM?
and why do we need it?



Canada

What is SLAM?
and why do we need it?



What is SLAM? The SLAM problem

Given:

- Robot control signal u_k (or measurement)
- A set of feature observations z_K (sensor measurements)

Estimate:

- Map of landmarks M_{k+1}
- Robot Pose V_{k+1}

Sources of Error:

- Control signal
- Mapping sensor
- Motion model
- Observation model

Probabilistic Basis of SLAM
Probability Theory

- Probability Theory gives us a framework for dealing with these sources of error
- The online SLAM theory is: $p(V_{k+1}, M_{k+1} | z_k, u_k)$
- Estimate the Joint Probability of V_{k+1} and M_{k+1} conditioned on z_k, u_k (Joint Conditional PDF)

Probabilistic Basis of SLAM
SLAM Probability toolkit

- Conditional Probability $p(x|y) = \frac{p(x,y)}{p(y)}$
- Product Rule $p(x,y) = p(x|y)p(y)$
- Bayes Rule $p(x|y) = \frac{p(y|x)p(x)}{p(y)}$ or $p(x|y) = \eta p(y|x)p(x)$
- Gaussian PDF $p(x) = det(2\pi\Sigma)^{-\frac{1}{2}} exp-\frac{1}{2}(x-\bar{x})^T\Sigma^{-1}(x-\bar{x})$

Probabilistic Basis of SLAM
SLAM as a Bayesian filter

$$p(V_{k+1}, M_{k+1} | z_{k+1}, u_{k+1}) =$$

Probabilistic Basis of SLAM
SLAM as a Bayesian filter

$$p(V_{k+1}, M_{k+1}|z_{k+1}, u_{k+1}) =$$
$$\eta p(z_{k+1}|V_{k+1}, M_{k+1})p(V_{k+1}, M_{k+1}|z_k, u_{k+1}) \text{ (Bayes Filter)}$$

Probabilistic Basis of SLAM
SLAM as a Bayesian filter

$$
\begin{aligned}
&p(V_{k+1}, M_{k+1}|z_{k+1}, u_{k+1}) = \\
&\eta p(z_{k+1}|V_{k+1}, M_{k+1}) p(V_{k+1}, M_{k+1}|z_k, u_{k+1}) \text{ (Bayes Filter)} = \\
&\eta p(z_{k+1}|V_{k+1}, M_{k+1}) \int p(V_{k+1}, M_{k+1}|V_k, z_k, u_{k+1}) p(V_k|z_k, u_{k+1}) dv_k
\end{aligned}
$$

Probabilistic Basis of SLAM
SLAM as a Bayesian filter

$$
\begin{aligned}
&p(V_{k+1}, M_{k+1} | z_{k+1}, u_{k+1}) = \\
&\eta p(z_{k+1} | V_{k+1}, M_{k+1}) p(V_{k+1}, M_{k+1} | z_k, u_{k+1}) \text{ (Bayes Filter)} = \\
&\eta p(z_{k+1} | V_{k+1}, M_{k+1}) \int p(V_{k+1}, M_{k+1} | V_k, z_k, u_{k+1}) p(V_k | z_k, u_{k+1}) dv_k \\
&= \eta p(z_{k+1} | V_{k+1}, M_{k+1}) \int p(V_{k+1} | V_k, u_k) p(V_k, M_k | z_k, u_k) dv_k
\end{aligned}
$$

- No closed form solution
- Approximate the solution using an Extended Kalman Filter

Probabilistic Basis of SLAM
SLAM as a Bayesian filter

$p(V_{k+1}, M_{k+1}|z_{k+1}, u_{k+1}) =$
$\eta p(z_{k+1}|V_{k+1}, M_{k+1})p(V_{k+1}, M_{k+1}|z_k, u_{k+1})$ (Bayes Filter) $=$
$\eta p(z_{k+1}|V_{k+1}, M_{k+1}) \int p(V_{k+1}, M_{k+1}|V_k, z_k, u_{k+1})p(V_k|z_k, u_{k+1})dv_k$
$= \eta p(z_{k+1}|V_{k+1}, M_{k+1}) \int p(V_{k+1}|V_k, u_k)\mathbf{p(V_k, M_k|z_k, u_k)}dv_k$

- Prior probability:
  Prior state and covariance estimates from the last filter iteration

Probabilistic Basis of SLAM
SLAM as a Bayesian filter

$$p(V_{k+1}, M_{k+1}|z_{k+1}, u_{k+1}) =$$
$$\eta p(z_{k+1}|V_{k+1}, M_{k+1})p(V_{k+1}, M_{k+1}|z_k, u_{k+1}) \text{ (Bayes Filter) } =$$
$$\eta p(z_{k+1}|V_{k+1}, M_{k+1}) \int p(V_{k+1}, M_{k+1}|V_k, z_k, u_{k+1})p(V_k|z_k, u_{k+1})dv_k$$
$$= \eta p(z_{k+1}|V_{k+1}, M_{k+1}) \int \mathbf{p(V_{k+1}|V_k, u_k)} p(V_k, M_k|z_k, u_k)dv_k$$

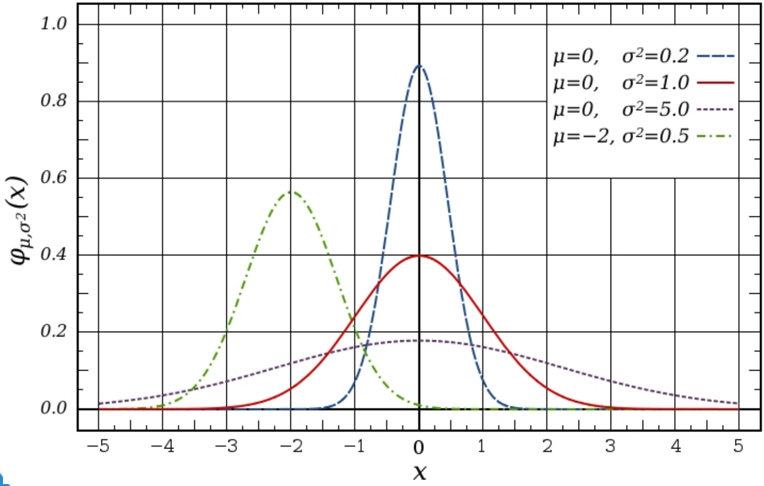
- Motion Model:
  Probabilistic motion model estimates the new vehicle pose covariance estimates from the prior estimate and the control

Probabilistic Basis of SLAM
SLAM as a Bayesian filter

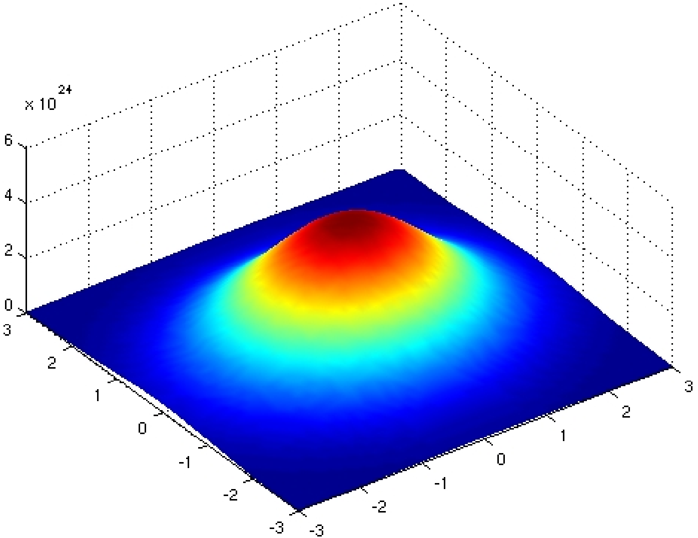
$$
\begin{aligned}
& p(V_{k+1}, M_{k+1}|z_{k+1}, u_{k+1}) = \\
& \eta p(z_{k+1}|V_{k+1}, M_{k+1}) p(V_{k+1}, M_{k+1}|z_k, u_{k+1}) \text{ (Bayes Filter)} = \\
& \eta p(z_{k+1}|V_{k+1}, M_{k+1}) \int p(V_{k+1}, M_{k+1}|V_k, z_k, u_{k+1}) p(V_k|z_k, u_{k+1}) dv_k \\
& = \eta \mathbf{p(z_{k+1}|V_{k+1}, M_{k+1})} \int p(V_{k+1}|V_k, u_k) p(V_k, M_k|z_k, u_k) dv_k
\end{aligned}
$$

- Measurement Model:
  Measurement model gives the expected value of the feature observations.

Probabilistic Basis of SLAM
Gaussian PDF



Probabilistic Basis of SLAM
Gaussian PDF



EKF SLAM

Simultaneous Localization and Mapping:

- Joint estimate both robot pose and position of unique landmarks
- Landmark estimates improve robot pose estimates and vice versa
- Use standard Extended Kalman Filtering techniques
- Assumes Gaussian noise and error.

$$M_k = \begin{pmatrix} x_1^m \\ y_1^m \\ \vdots \\ x_n^m \\ y_n^m \end{pmatrix}, \quad V_k = \begin{pmatrix} x_k^r \\ y_k^r \\ \theta_k^r \end{pmatrix}$$

EKF SLAM Algorithm

$$X_k = \begin{pmatrix} V_k \\ M_k \end{pmatrix} \quad , \quad P_k = \begin{pmatrix} P_v & P_{vm} \\ P_{vm}^T & P_m \end{pmatrix} \qquad (1)$$

- **Prediction** - A prediction of the new state vector and covariance matrix is calculated from the previous state and covariance, and the new control u_k.

EKF SLAM Algorithm

$$X_k = \begin{pmatrix} V_k \\ M_k \end{pmatrix} \quad , \quad P_k = \begin{pmatrix} P_v & P_{vm} \\ P_{vm}^T & P_m \end{pmatrix} \qquad (1)$$

- **Data Association** - Find matches between the current landmarks M_k and the new set of observed features z_k.

EKF SLAM Algorithm

$$X_k = \begin{pmatrix} V_k \\ M_k \end{pmatrix} \quad , \quad P_k = \begin{pmatrix} P_v & P_{vm} \\ P_{vm}^T & P_m \end{pmatrix} \tag{1}$$

- **Measurement Update** - Calculate the Kalman gain for each observed landmark and update the state and covariance values based on that Kalman gain and the measurement innovation.

EKF SLAM Algorithm

$$X_k = \begin{pmatrix} V_k \\ M_k \end{pmatrix} \quad , \quad P_k = \begin{pmatrix} P_v & P_{vm} \\ P_{vm}^T & P_m \end{pmatrix} \tag{1}$$

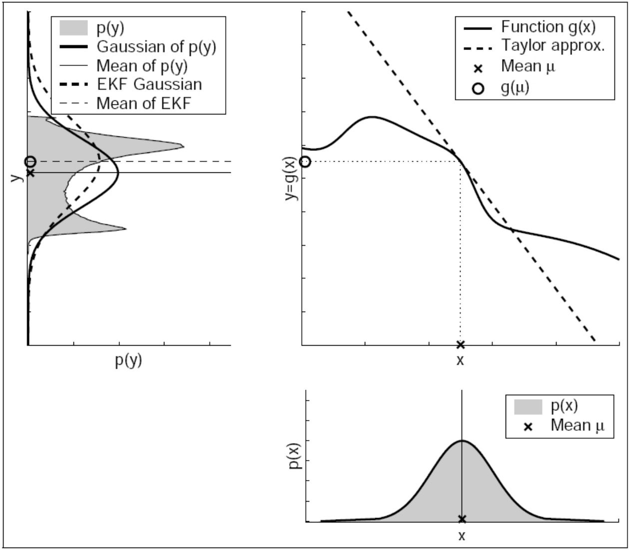
- **Augmentation** - Integrate newly observed landmarks into the state vector and covariance matrix.

EKF SLAM
Linearization

- Kalman Filters work under a linear Gaussian assumption
- Robot motion and measurement functions are non-linear
- Use Taylor Expansion to linearize (Approximate a Gaussian)
  $g \approx g + g'$
- g' is Jacobian of the function with respect to its variables

EKF SLAM

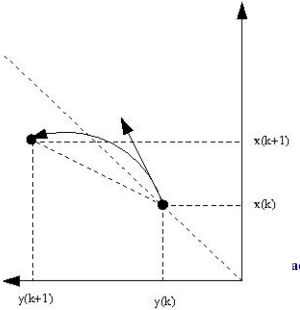
Linearization



EKF SLAM
Prediction

- Predict the robot motion from the control signal
- Linearize to approximate the Gaussian

$$\bar{V}_{k+1} = V_k + \Delta V$$



EKF SLAM
Prediction

- Predict the robot motion from the control signal
- Linearize to approximate the Gaussian

$$V_{k+1} = \begin{pmatrix} x_k^r \\ y_k^r \\ \theta_k^r \end{pmatrix} + \begin{pmatrix} -\frac{v_k}{\omega_k} sin(\theta_k^r) + \frac{v_k}{\omega_k} sin(\theta_k^r + \omega_k \Delta t) \\ \frac{v_k}{\omega_k} cos(\theta_k^r) - \frac{v_k}{\omega_k} cos(\theta_k^r + \omega_k \Delta t) \\ \omega_k \Delta t \end{pmatrix}$$

EKF SLAM
Prediction

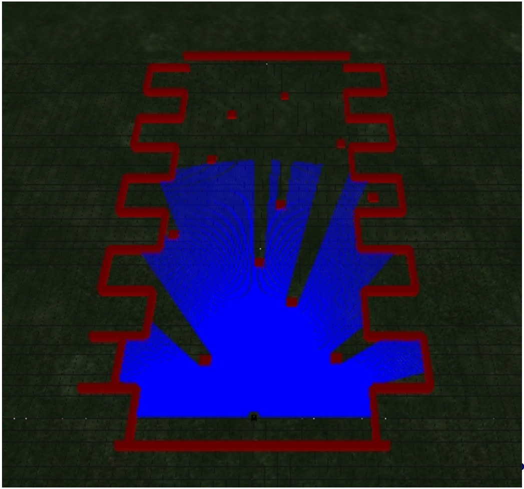
- Predict the robot motion from the control signal
- Linearize to approximate the Gaussian

$$\bar{P}_{v(k+1)} = G_k P_{v(k)} G_k^T$$
$$\bar{P}_{vm(k+1)} = R_k Q R_k^T$$
$$\bar{P}_{k+1} = \begin{pmatrix} \bar{P}_{v(k+1)} & \bar{P}_{vm(k+1)} \\ \bar{P}^T_{vm(k+1)} & P_m \end{pmatrix}$$

G linearizes the state transition function
R maps additional motion noise into the state space

EKF SLAM
Feature Extraction Algorithms

Extract stable salient features from environment):



EKF SLAM
Feature Extraction Algorithms

What do we want in feature extraction algorithm?

EKF SLAM
Feature Extraction Algorithms

What do we want in feature extraction algorithm?

- Stable features
- Outlier rejection
- Accuracy
- Speed

EKF SLAM
Feature Extraction Algorithms

What do we want in feature extraction algorithm?

- Expectation Maximization

EKF SLAM
Feature Extraction Algorithms

What do we want in feature extraction algorithm?

- RANSAC

EKF SLAM
Feature Extraction Algorithms

What do we want in feature extraction algorithm?

- Split and Merge

EKF SLAM
Feature Extraction Algorithms

What do we want in feature extraction algorithm?

- Hough Transform

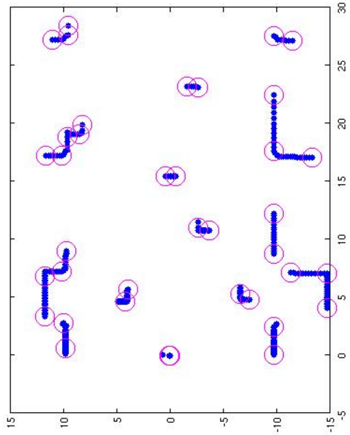
EKF SLAM
Feature Extraction Algorithms

What do we want in feature extraction algorithm?

- **Incremental Line Fitting**

EKF SLAM
Laser Feature Extraction

- Extract features from Raw laser data
- Incremental algorithm (line tracking)
    - Fit a line to a set of points
    - Compute the residual
    - If below error threshold add another point and repeat
    - If above the last point added is a feature



EKF SLAM
Data Association

- Find matches between features and landmarks
- For each feature
  - Calculate the predicted feature for each landmark (measurement model)
  - compute the Mahalanobis Distance:
  - choose the feature/landmark with the lowest distance (Maximum Likelihood) below some threshold
- End for

EKF SLAM
Data Association

- Measurement Model:

$$z_j^m = \begin{pmatrix} r_j^m \\ \beta_j^m \end{pmatrix} = \begin{pmatrix} \sqrt{\Delta y^2 + \Delta x^2} \\ \arctan(\Delta y, \Delta x) \end{pmatrix}$$

EKF SLAM
Data Association

- Measurement Model:

$$z_j^m = \begin{pmatrix} r_j^m \\ \beta_j^m \end{pmatrix} = \begin{pmatrix} \sqrt{\Delta y^2 + \Delta x^2} \\ \arctan(\Delta y, \Delta x) \end{pmatrix}$$

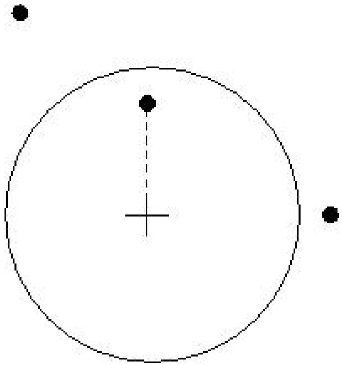
- Mahalanobis Distance normalizes feature/landmark distance based on their covariances:

$$d_{ij} = (z_i - z_j^m)^T (H_j P_{k+1}^- H_j{}^T + R)^{-1} (z_i - z_j^m)$$

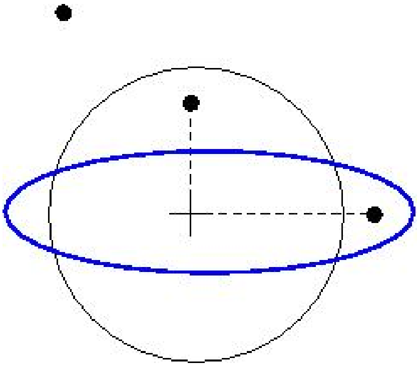
EKF SLAM
Why Mahalanobis Distance?

Why Mahalanobis Distance:

EKF SLAM
Why Mahalanobis Distance?



EKF SLAM
Why Mahalanobis Distance?



Points which lie on the same covariance ellipse have are equidistant

EKF SLAM
Update

- For each data association
  - Calculate Kalman Gain:
    $K_j = \bar{P}_{k+1} {H_j}^T (H_j \bar{P}_{k+1} {H_j}^T + R)^{-1}$
    H linearizes the measurement model
  - Update state and covariance estimates:
    $$X_{k+1} = \bar{X}_{k+1} + K_j(z_i - {z_j}^m)$$
    $$P_{k+1} = \bar{P}_{k+1} - K_j H_j \bar{P}_{k+1}$$
- End for

- Innovation vector - Larger $\longrightarrow$ bigger correction
- Kalman Gain - Weights the innovation vector.
  - Smaller measurement error $\longrightarrow$ higher gain (trust the measurement)
  - Smaller covariance $\longrightarrow$ smaller gain (trust the prediction)

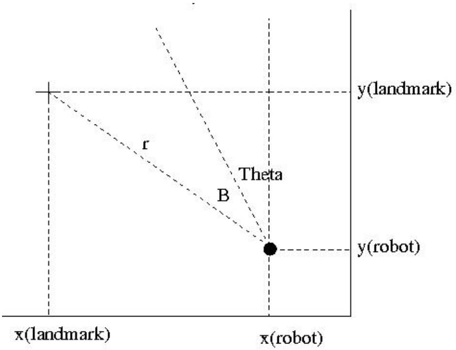
EKF SLAM
Augment

- State Vector and Covariance matrix grow as new landmarks are observed
- If a feature has no matches, add it to the state vector as a new landmark

EKF SLAM
Augment

New Landmark Locations:

$$x_i^m = x_k^r + r_i \cos(\theta_k^r + \beta_i)$$
$$y_i^m = y_k^r + r_i \sin(\theta_k^r + \beta_i)$$



EKF SLAM
Augment

Append to state vector:

$$X = \begin{pmatrix} X \\ x_i^m \\ y_i^m \end{pmatrix}$$

EKF SLAM
Augment

Original Covariance Matrix:

$$P = \begin{pmatrix} P_v & P_{vm} \\ {P_{vm}}^T & P_m \end{pmatrix}$$

EKF SLAM
Augment

Augmented Covariance Matrix:

$$P_a = \begin{pmatrix} P_v & P_{vm} & {P_v}^T {F_i}^T \\ {P_{vm}}^T & P_m & {P_{vm}}^T {F_i}^T \\ F_i P_v & F_i P_{vm} & F_i P_v {F_i}^T + M_i R {M_i}^T \end{pmatrix}$$

F and M are the Jacobians which linearize the new landmark equations with respect to vehicle pose and measurement variables respectively

EKF SLAM
Other Issues

- Can use provisional landmark list to reject spurious features
- Pruning (removing old/irrelevant landmarks)
- Landmark Signatures (reflectivity/color/shape/etc.)
- Landmark minimum distance
- Intelligent update (Only update *relevant* landmarks)
- Sub-Mapping (more on this later)

EKF SLAM
Advantages

- Straightforward application of the EKF
- Large body of research to pull from
- Works reasonably well for small number of features and distinct landmarks

EKF SLAM Disadvantages

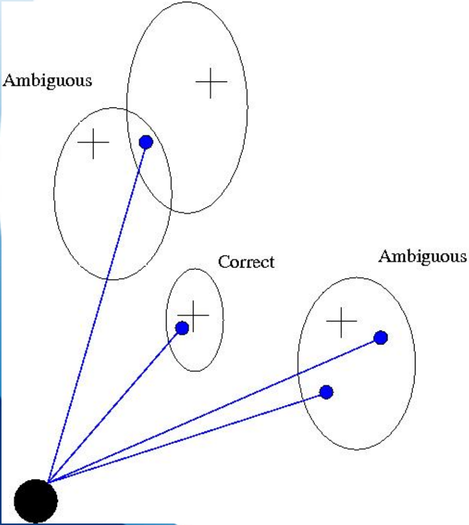
- Complexity Quadratic with number of features
- No guarantee of convergence in non-linear case
- Make hard decisions about data associations
- Can't correct for erroneous data associations
- Need sufficiently distinct landmarks (low clutter)
- Gaussian Assumption

EKF SLAM Variants

- Unscented KF SLAM
  - Linearization using Sigma points
  - Preserves the true mean and covariance of the posterior
- Extended Information Filter SLAM
  - Dual representation of EKF
  - Less complex measurement update

Data Association

Erroneous data association WILL cause SLAM to fail!



- 1 feature to 2 landmarks
- 2 features to 1 landmark
- clutter
- Symmetry
- Reduce the max Mahalanobis distance?

Data Association

How do determine the correspondence between a feature and landmark?

- Individual Compatibility
- Joint Compatibility Branch and Bound
- Combined Constrained Data Association
- Randomized Joint Compatibility
- Multi hypothesis Data Association

Data Association
Individual Compatibility

- Calculate Mahalanobis distance from each feature to landmark
- Choose feature with smallest distance to landmark within threshold (validation gate)
- Disregard ambiguous associations
- Advantage: Simple
- Disadvantage: Can be error prone especially in clutter

Data Association Joint Compatibility Branch and Bound

- Consider multiple associations simultaneously
- Find the largest set of matches which correspond
- Joint Mahalanobis distance measurement with joint validation gate
- Branch and bound interpretation tree search
- Advantage: Robust
- Disadvantage: Tree search computation/May be Ambiguous sets

Data Association
Joint Compatibility Branch and Bound

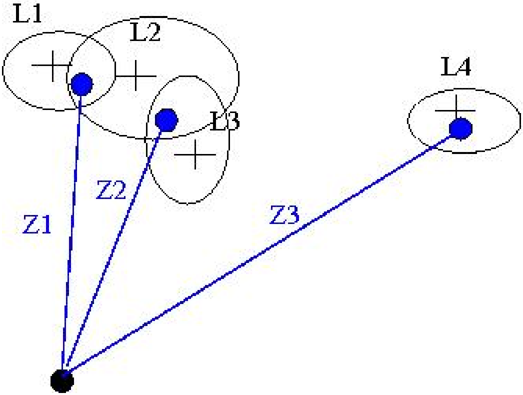
Potential Matches

$z1 \longrightarrow L1$ or $L2$

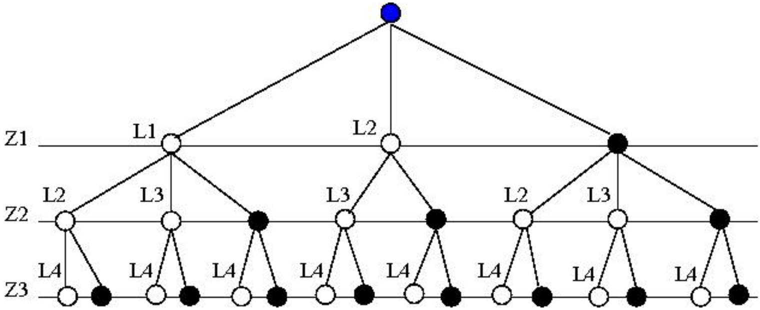
$z2 \longrightarrow L2$ or $L3$

$z3 \longrightarrow L4$

Null



Data Association Joint Compatibility Branch and Bound



Data Association Combined Constrained Data Association

- Correspondence Graph technique
- Nodes are individually compatible associations
- Edges are compatible associations
- Find the maximum clique of associations that are mutually compatible
- Consider both relative and absolute constraints
- Advantage: Can work with no pose info (relative constraints)
- Disadvantage: May be ambiguous sets

Data Association
Randomized Joint Compatibility

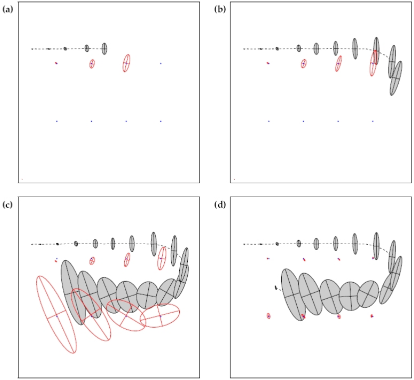
- JCBB with RANSAC
- Uses relative constraints
- Randomly select m of the n feature measurements
- Advantage: Reduced complexity, no pose necessary
- Disadvantage: Don't always know P_{fail} and P_{good}

Closing the Loop
What is loop closing?

- Even with the best SLAM algorithm pose uncertainty will increase as the vehicle moves.
- This pose uncertainty means that landmark locations further from the map origin have a higher uncertainty
- Revisiting a previously observed landmarks significantly reduces uncertainty in robot and landmark pose estimates
- Errors in landmark estimates are correlated with robot pose
- New pose info necessarily improved landmark estimates

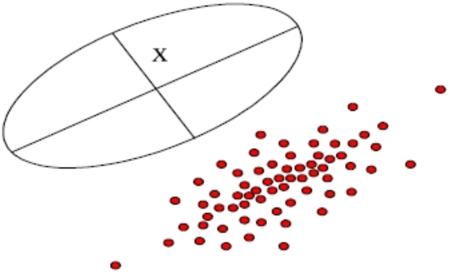
Closing the Loop
How to Close the loop

- Map Matching Techniques (feature matching)
- Can use pose estimate to constrain the search
- Expectation Maximization Algorithm
- Non-linear Optimization



FastSLAM

- Represent probability distributions by set of particles
- Rao-Blackwellized particle filter (low dimensional EKFs)
- Each particle maintains its own pose and map estimates (multi-hypothesis)
- Each landmark has its own EKF
- N landmarks and M particles we have $MxN + 1$ filters



FastSLAM Algorithm

Do M times:

- Retrieve pose from particle k
- Predict a new pose (motion model)
- Data Association
- Measurement Update
- Importance Weight

END

Re-sample with Replacement using the importance weight

Raw Odometry

FastSLAM

FastSLAM
Advantages

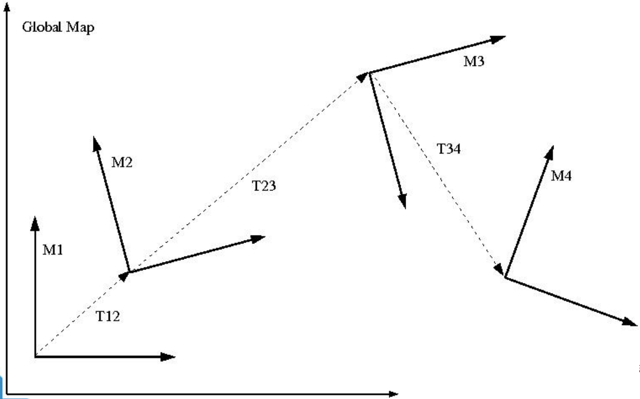
- Time logarithmic to number of landmarks
- Multi-hypothesis data association (robustness)
- No linearization of non-linear motion models
- Solves both full SLAM and Online SLAM problems

FastSLAM Disadvantages

- Throws away correlations
- Over optimistic pose estimate
- Harder to close the loop

Sub-Mapping SLAM

- Partition SLAM into sub-maps
- Optimally Fuse at a global level periodically
- Reduces EKF linearization errors
- Reduces computational complexity
- Closing the loop techniques for sub-maps



Sub-Mapping SLAM Algorithms

- ATLAS
- Hybrid-SLAM
- Decoupled Stochastic Mapping
- Hierarchical SLAM
- Conditionally In dependant SLAM
- Divide and Conquer

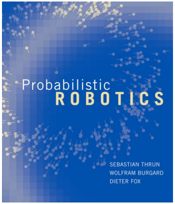
EKF SLAM vs. D&C SLAM

Other Topics

- Visual SLAM
  - SIFT, Harris Corners, SURF, etc. for features
  - No range to features
  - Inverse Depth
- Muti-robot SLAM
  - Teams map building
  - Information sharing/cooperation
- Learning for SLAM
  - Adaptive SLAM
  - Learnt sensor, observation, motion models
- 6-DOF SLAM/Mesh Models
- Feature Extraction Algorithms
- Scan matching

Resources

- Probabilistic Robotics



- SLAM Summer School 2009

  http://www.acfr.usyd.edu.au/education/summerschool.shtml

- openslam.org (MATLAB, C, Different algorithms, data sets, etc.)

Questions?