Robust Least Squares for SLAM

⇒ For Least Square we have assumption of

(No Outliers!)

⇒ Optimization is Sensitive to outliers.

⇒ Data association is ambiguous and not always perfect.

MaxMixtures or Dealing with multiple Modes

(By: Pratik Agarwal, Edwin Olson, Wolfram, Burgard)

⇒ We can express a multi-modal belief by a sum of Gaussians.

$$P(Z|x) = \eta \exp\left(-\frac{1}{2} e_{ij}^T \Omega_{ij} e_{ij}\right)$$

$$\Downarrow$$

$$P(Z|x) = \sum_k \omega_k \eta_k \exp\left(-\frac{1}{2} e_{ij_k}^T \Omega_{ij_k} e_{ij_k}\right)$$

{Sum of Gaussians with K modes}

⇒ During the error minimization, we consider the negative log likelihood

$$-\log P(Z|X) = \frac{1}{2} e_{ij}^T \Omega_{ij} e_{ij} - \log \eta$$

$$\Downarrow$$

$$-\log P(Z|X) = -\log \sum_k \omega_k \eta_k \exp\left(-\frac{1}{2} e_{ij_k}^T \Omega_{ij_k} e_{ij_k}\right)$$

The log cannot be moved inside the sum!

★ <u>Max-Mixture Approximation</u>

⇒ Instead of computing the sum of Gaussians at X, compute the maximum of the Gaussians.

$$P(Z|X) = \sum_k \omega_k \eta_k \exp\left(-\frac{1}{2} e_{ij_k}^T \Omega_{ij_k} e_{ij_k}\right)$$

$$\simeq \max_k \omega_k \eta_k \exp\left(-\frac{1}{2} e_{ij_k}^T \Omega_{ij_k} e_{ij_k}\right)$$

• $\log P(Z|X)$

$$= \max_k \left(-\frac{1}{2} e_{ij_k}^T \Omega_{ij_k} e_{ij_k}\right) + \log(\omega_k \eta_k)$$

$$-\log P(z|x) = \min_K \frac{1}{2} e_{ij_K}^T \Omega_{ij_K} e_{ij_K} - \log(\omega_K \eta_K)$$

★ Integration

⇒ With the max mixture formulation, the log likelihood again results in local quadratic forms.

⇒ Easy to integrate in the optimizer:

1. Evaluate all K components
2. Select the component with the maximum log likelihood
3. Perform the optimization as before using only the max components (as a Single Gaussian)

★ Max-Mixture and Outliers

⇒ MM formulation is useful for multimodel constraints

⇒ Outliers: One mode represents the main constraint and a Second model uses a flat Gaussian for the outlier hypothesis.

Dynamic Covariance Scaling

$$X^* = \underset{X}{\operatorname{argmin}} \sum_{ij} e_{ij}(x)^T \Omega_{ij} e_{ij}(x)$$

$$\Downarrow$$

$$X^* = \underset{X}{\operatorname{argmin}} \sum_{ij} e_{ij}(x)^T (\underbrace{s_{ij}^2}_{\text{Scaling parameter}} \Omega_{ij}) e_{ij}(x)$$

$$s_{ij} = \min\left(1, \frac{2\Phi}{\Phi + \underbrace{\chi_{ij}^2}_{\text{Error term (without scaling)}}}\right)$$

★ <u>DCS for dealing with outliers</u>

⇒ Add an additional weighting term to the error function.

⇒ The weight depends on the error value.

⇒ <u>Idea</u>: "Weight down constraints that are far away from the mean estimate"

Least Square with Robust Kernels

★ Optimizing with Outliers

⇒ Assuming a Gaussian error in the constraints is not always realistic.

↳ Large errors are problematic.

★ Robust M-Estimators

⇒ $\rho(e)$ function used to define the PDF.

$$P(e) = \exp(-\rho(e))$$

⇒ Minimizing the neg. log likelihood

$$x^* = \underset{x}{\arg\min} \sum_i \rho(e_i(x))$$

★ Different Rho Functions

$\rho(e) = e^2$ {Gaussian}

$\rho(e) = |e|$ {L1 norm}

⇒ Huber M-estimator

$$\rho(e) = \begin{cases} \frac{e^2}{2} & \text{if } |e| < c \\ c\left(|e| - \frac{c}{2}\right) & \text{otherwise} \end{cases}$$

⇒ Gradient at optimum goes to zero.

$$\frac{\delta(\rho(e_i(x)))}{\delta x} = \rho'(e_i(x)) \frac{\delta e_i(x)}{\delta x} = 0$$

⇒ For weighted least square:

$$\frac{1}{2} \cdot \frac{\delta(w_i e_i^2(x))}{\delta x} = w_i e_i(x) \frac{\delta e_i(x)}{\delta x} = 0$$

⇒ We can use weighted least squares if we set the weight using the Kernel as:

$$\boxed{w_i = \frac{1}{e_i(x)} \rho'(e_i(x))}$$

Generalized Robust Kernels (2019)

$$\rho(e, \alpha, c) = \frac{|\alpha-2|}{\alpha}\left(\left(\frac{(e/c)^2}{|\alpha-2|} + 1\right)^{\alpha/2} - 1\right)$$

↳ By changing the α, you are changing the shape of the loss function.

★ Adaptive Robust Loss function

⇒ Jointly optimize over x and α

$$(x^*, \alpha^*) = \underset{(x,\alpha)}{\operatorname{argmin}} \sum_{i=1}^{N} \rho(e_i(x), \alpha)$$

⇒ Define a probability distribution for the general loss function

$$P(e, \alpha, c) = \frac{1}{cZ(\alpha)} e^{-f(e,\alpha,c)}$$

$$Z(\alpha) = \int_{-\infty}^{\infty} e^{-f(e,\alpha,1)} de$$

⇒ Adaptive loss function defined by:

$$\mathcal{L}(e, a, c) = -\log P(e, \alpha, c)$$
$$= f(e, \alpha, c) + \log c Z(\alpha)$$

⇒ $Z(\alpha)$ approaches ∞ for negative α.

⇒ We can limit the range of outliers to maintain a pdf.

$$Z(\alpha) = \int_{-\tau}^{\tau} e^{-f(e,\alpha,1)} de$$

⇒ Problems in practice:

- New Jacobians need to be computed.
- α can dominate the parameter estimation for complex problem
- Sensitive to initial guess

EM-Based Optimization with the Adaptive Robust Kernel

⇒ Solve via Expectation-Maximization

⇒ E-Step : 1D line search problem

$$\alpha^t = \arg\max_{\alpha} \sum_{i=1}^{N} \log P(e_i(x^{t-1}), \alpha^{t-1}, c)$$

⇒ M-Step : Minimize as weighted least squares.

$$x^t = \arg\min_{x} \sum_{i=1}^{N} \rho_a(e_i(x), \alpha^t, c)$$

⇒ EM-based estimation provides better results than the joint optimization.
