Carnegie Mellon
THE ROBOTICS INSTITUTE

SLAM: Nonlinear Least-Squares

Robot Localization and Mapping
16-833

Michael Kaess

March 15, 2021

Nonlinear Least-Squares

- On the board:
  - Linearization
  - Gradient descent
  - Gauss-Newton
  - Levenberg-Marquardt
  - Powell's Dog-Leg

Nonlinear -> Linear Least Squares

Taylor series expansion:

$$h_i(X_i) = h_i(X_i^0 + \Delta_i) \approx h_i(X_i^0) + H_i \Delta_i$$

Measurement Jacobian: $H_i \triangleq \left.\dfrac{\partial h_i(X_i)}{\partial X_i}\right|_{X_i^0}$

State update vector: $\Delta_i \triangleq X_i - X_i^0$

Linear least-squares problem:

$$\begin{aligned}
\Delta^* &= \underset{\Delta}{\operatorname{argmin}} \sum_i \left\| h_i(X_i^0) + H_i \Delta_i - z_i \right\|_{\Sigma_i}^2 \\
&= \underset{\Delta}{\operatorname{argmin}} \sum_i \left\| H_i \Delta_i - \underbrace{\left\{ z_i - h_i(X_i^0) \right\}}_{\text{Prediction error}} \right\|_{\Sigma_i}^2
\end{aligned}$$

Simplifying to Quadratic Form

Original term with Mahalanobis Distance:

$$\begin{aligned}\Delta^* &= \underset{\Delta}{\operatorname{argmin}} \sum_i \left\| h_i(X_i^0) + H_i\Delta_i - z_i \right\|_{\Sigma_i}^2 \\ &= \underset{\Delta}{\operatorname{argmin}} \sum_i \left\| H_i\Delta_i - \left\{ z_i - h_i(X_i^0) \right\} \right\|_{\Sigma_i}^2\end{aligned}$$

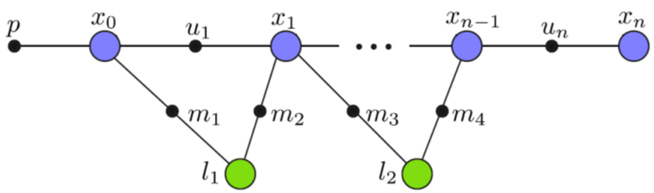
Simplification:

$$\begin{aligned}A_i &= \Sigma_i^{-1/2} H_i \\ b_i &= \Sigma_i^{-1/2} \left(z_i - h_i(X_i^0) \right)\end{aligned}$$

Quadratic form:

$$\begin{aligned}\Delta^* &= \underset{\Delta}{\operatorname{argmin}} \sum_i \| A_i\Delta_i - b_i \|_2^2 \\ &= \underset{\Delta}{\operatorname{argmin}} \| A\Delta - b \|_2^2\end{aligned}$$

SLAM as a Sparse Least-Squares Problem



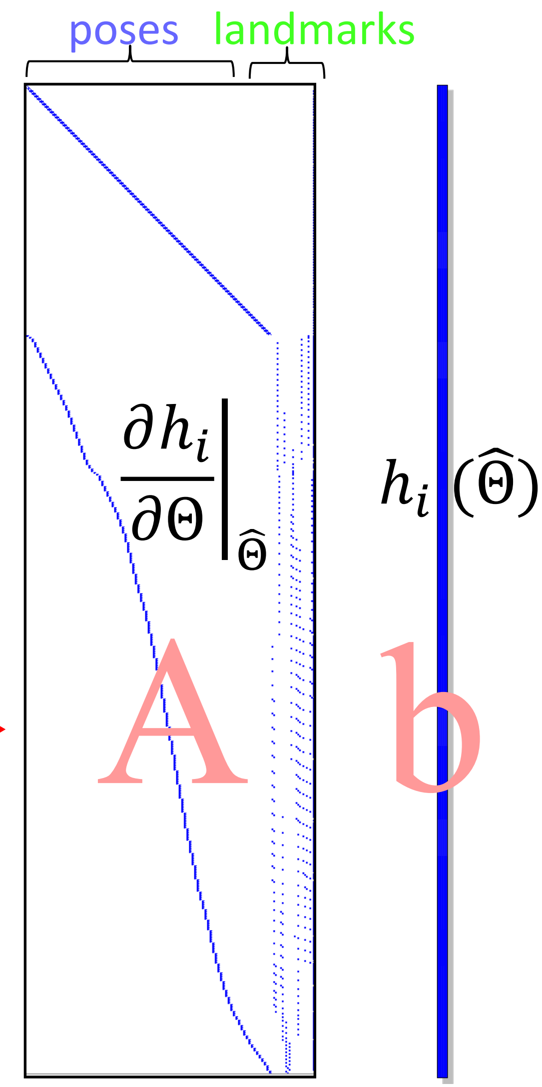
$$\operatorname{argmax}_{\Theta} \prod_{z \in Z} p(z|\Theta)$$

Gaussian noise

$$\operatorname{argmin}_{\Theta} \sum_i \|h_i(\Theta)\|^2$$

$$\operatorname{argmin}_{\theta} \|A\theta - b\|^2$$

Sparse measurement Jacobian



Steepest Descent

Cost function:

$$g(X) \stackrel{\Delta}{=} \sum_i \|h_i(X_i) - z_i\|^2_{\Sigma_i}$$

$$g(X) \approx \|A(X - X^t) - b\|_2^2$$

Steepest descent step:

$$\Delta_{sd} = -\alpha \, \nabla g\left(X\right)|_{X=X^t}$$

gradient: $\nabla g\left(X\right)|_{X=X^t} = -2A^T b$

Gauss-Newton

Cost function:

$$g(X) \approx \|A(X - X^t) - b\|_2^2$$

Gauss-Newton step:

$$A^T A \Delta_{gn} = A^T b$$

Levenberg-Marquardt

Levenberg:

$$(A^T A + \lambda I)\Delta_{lb} = A^T b$$

Levenberg-Marquardt:

$$\left(A^T A + \lambda \operatorname{diag}(A^T A)\right) \Delta_{lm} = A^T b$$

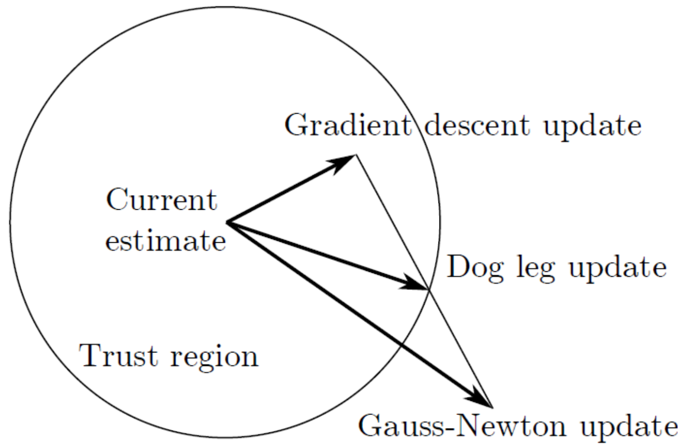
Levenberg-Marquardt

Algorithm 2.1 The Levenberg-Marquardt algorithm

function $\mathrm{LM}(g(), X^0)$ ▷ quadratic cost function $g()$, ▷ initial estimate X^0
  $\lambda = 10^{-4}$
  $t = 0$
  repeat
    $A, b \leftarrow$ linearize $g(X)$ at X^t
    $\Delta \leftarrow$ solve $\left(A^T A + \lambda\,\mathrm{diag}(A^T A)\right) \Delta = A^T b$
    if $\mathrm{g}(X^t + \Delta) < \mathrm{g}(X^t)$ **then**
      $X^{t+1} = X^t + \Delta$ ▷ accept update
      $\lambda \leftarrow \lambda / 10$
    else
      $X^{t+1} = X^t$ ▷ reject update
      $\lambda \leftarrow \lambda * 10$
    $t \leftarrow t + 1$
  until convergence
  return X^t ▷ return latest estimate

Powell's Dog-Leg Algorithm

Key idea: Explicitly maintain a trust region



Gain ratio:

$$\rho = \frac{g(X^t) - g(X^t + \Delta)}{L(0) - L(\Delta)}$$

where $L(\Delta) = A^T A \Delta - A^T b$