深蓝学院
shenlanxueyuan.com

从零手写VIO-第六期
大作业 思路讲解

主讲人 吴少腾

大作业

●更好的优化策略

- ➢ 选用更优的 LM 策略, 使得 VINS-Mono 在 MH-05 数据集上收敛速度更快或者精度更高.
- ➢ 实现 dog-leg 算法替换 LM 算法, 并测试替换后的 VINS-Mono 在 MH-05 上算法精度.

●更快的 MakeHessian

- ➢ 可以采用任何一种或多种加速方式 (如多线程, 如sse指令集等) 对信息矩阵的拼接函数加速, 并给出详细的实验对比报告.

更好的优化策略

深蓝学院
shenlanxueyuan.com

●更优的LM策略

1. 算法退出条件（problem.cc中被注释的代码）：

- 限制步数；
- 残差阈值；
- $\|\boldsymbol{\delta_p}\|$阈值；
- …

2. 更换阻尼策略

…

Input: A vector function $f: \mathcal{R}^m \rightarrow \mathcal{R}^n$ with $n \geq m$, a measurement vector $\mathbf{x} \in \mathcal{R}^n$ and an initial parameters estimate $\mathbf{p}_0 \in \mathcal{R}^m$.
Output: A vector $\mathbf{p}^+ \in \mathcal{R}^m$ minimizing $||\mathbf{x} - f(\mathbf{p})||^2$.
Algorithm:
$k := 0$; $\nu := 2$; $\mathbf{p} := \mathbf{p}_0$;
$\mathbf{A} := \mathbf{J}^T\mathbf{J}$; $\epsilon_\mathbf{p} := \mathbf{x} - f(\mathbf{p})$; $\mathbf{g} := \mathbf{J}^T\epsilon_\mathbf{p}$;
stop:=($||\mathbf{g}||_\infty \leq \varepsilon_1$); $\mu := \tau * \max_{i=1,\dots,m}(A_{ii})$;
while (not stop) and ($k < k_{max}$)
  $k := k + 1$;
  repeat
    Solve $(\mathbf{A} + \mu\mathbf{I})\delta_\mathbf{p} = \mathbf{g}$;
    if ($||\delta_\mathbf{p}|| \leq \varepsilon_2||\mathbf{p}||$)
      stop:=true;
    else
      $\mathbf{p}_{new} := \mathbf{p} + \delta_\mathbf{p}$;
      $\rho := (||\epsilon_\mathbf{p}||^2 - ||\mathbf{x} - f(\mathbf{p}_{new})||^2)/(\delta_\mathbf{p}^T(\mu\delta_\mathbf{p} + \mathbf{g}))$;
      if $\rho > 0$
        $\mathbf{p} = \mathbf{p}_{new}$;
        $\mathbf{A} := \mathbf{J}^T\mathbf{J}$; $\epsilon_\mathbf{p} := \mathbf{x} - f(\mathbf{p})$; $\mathbf{g} := \mathbf{J}^T\epsilon_\mathbf{p}$;
        stop:=($||\mathbf{g}||_\infty \leq \varepsilon_1$);
        $\mu := \mu * \max(\frac{1}{3}, 1 - (2\rho - 1)^3)$; $\nu := 2$;
      else
        $\mu := \mu * \nu$; $\nu := 2 * \nu$;
      endif
    endif
  until ($\rho > 0$) or (stop)
endwhile
$\mathbf{p}^+ := \mathbf{p}$;

LM算法流程[1]

[1] M. L. A. Lourakis and A. A. Argyros, "Is Levenberg-Marquardt the most efficient optimization algorithm for implementing bundle adjustment?," Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1, Beijing, 2005, pp. 1526-1531 Vol. 2, doi: 10.1109/ICCV.2005.128.

更好的优化策略

➢ 更换LM阻尼策略

① 采用第三章作业中的更新阻尼更新策略.

② 利用滑动窗口算法的性质，marginalize一帧后增加一帧，问题结构变化不大。**因此可利用上一次求解的结果优化当前问题：**

记录上一次求解后的阻尼因子μ和残差平方和chi值，作为当前问题迭代初值.

a. 若将μ代入当前问题，优化后新的chi值变**小**了，说明μ是一个**较好**的初值，可以取阻尼因子$\mu = {}^{1}/_{2}\,\mu$，使LM算法更接近**高斯牛顿法**。

b. 若将μ代入当前问题，优化后新的chi值变**大**了，说明μ是一个**较差**的初值，可以取阻尼因子$\mu = 2\mu$，使LM算法更接近**最速下降法**。

更好的优化策略

●Dog Leg算法

属于Trust Region类优化算法：

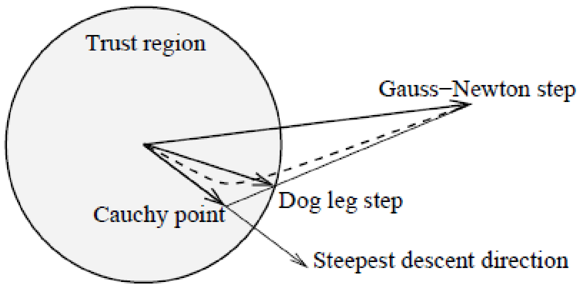
- 在以当前点为中心，半径Δ 的区域（信赖域）内，将目标函数做二阶近似
- 分别计算**最速下降法**和**高斯牛顿法**的最优增量$\boldsymbol{\delta}_{\mathrm{sd}}$和$\boldsymbol{\delta}_{\mathrm{gn}}$，根据信赖域大小选择合适的增量

要点：

① 计算$\boldsymbol{\delta}_{\mathrm{sd}}$和$\boldsymbol{\delta}_{\mathrm{gn}}$；

② 选取合适的Dog Leg步长$\boldsymbol{\delta}_{\mathrm{dl}}$；

③ 更新信赖域半径Δ；



Dog Leg算法示意图[1]

[1] M. L. A. Lourakis and A. A. Argyros, "Is Levenberg-Marquardt the most efficient optimization algorithm for implementing bundle adjustment?," Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1, Beijing, 2005, pp. 1526-1531 Vol. 2, doi: 10.1109/ICCV.2005.128.

更好的优化策略

深蓝学院 shenlanxueyuan.com

① 计算$\boldsymbol{\delta}_{\mathrm{sd}}$和$\boldsymbol{\delta}_{\mathrm{gn}}$；

第三章中：

$$\boldsymbol{f}(\boldsymbol{x}+\Delta\boldsymbol{x}) \approx \boldsymbol{f}(\boldsymbol{x})+\boldsymbol{J}\Delta\boldsymbol{x}$$

$$F(\boldsymbol{x}+\Delta\boldsymbol{x})=\frac{1}{2}\left\|\boldsymbol{f}(\boldsymbol{x}+\Delta\boldsymbol{x})^{\mathrm{T}}\boldsymbol{f}(\boldsymbol{x}+\Delta\boldsymbol{x})\right\|_2^2 \approx F(\boldsymbol{x})+\Delta\boldsymbol{x}^{\mathrm{T}}\boldsymbol{J}^T\boldsymbol{f}+\frac{1}{2}\Delta\boldsymbol{x}^{\mathrm{T}}\boldsymbol{J}^T\boldsymbol{J}\Delta\boldsymbol{x} \qquad s.t. \|\Delta\boldsymbol{x}\| \le \Delta$$

最速下降法$\boldsymbol{\delta}_{\mathrm{sd}}=-\alpha\boldsymbol{d}$

增量方向：$\boldsymbol{d}=F'(\boldsymbol{x})=\left(\boldsymbol{J}^{\mathrm{T}}\boldsymbol{f}\right)^{\mathrm{T}}$

增量步长：$\dfrac{dF(\boldsymbol{x}-\alpha\boldsymbol{d})}{d\alpha}=0 \qquad \alpha=\dfrac{\left(\boldsymbol{J}^{\mathrm{T}}\boldsymbol{f}\right)^{\mathrm{T}}\boldsymbol{J}^{\mathrm{T}}\boldsymbol{f}}{(\boldsymbol{J}^{\mathrm{T}}\boldsymbol{f})^{\mathrm{T}}(\boldsymbol{J}^{\mathrm{T}}\boldsymbol{J})\boldsymbol{J}^{\mathrm{T}}\boldsymbol{f}}$

$$\boldsymbol{\delta}_{\mathrm{sd}}=\frac{-\left(\boldsymbol{J}^{\mathrm{T}}\boldsymbol{f}\right)^{\mathrm{T}}\boldsymbol{J}^{\mathrm{T}}\boldsymbol{f}}{(\boldsymbol{J}^{\mathrm{T}}\boldsymbol{f})^{\mathrm{T}}(\boldsymbol{J}^{\mathrm{T}}\boldsymbol{J})\boldsymbol{J}^{\mathrm{T}}\boldsymbol{f}}\cdot\left(\boldsymbol{J}^{\mathrm{T}}\boldsymbol{f}\right)^{\mathrm{T}}$$

对应代码中 $\dfrac{b^{\mathrm{T}}b}{b^{\mathrm{T}}Hb}\cdot b$

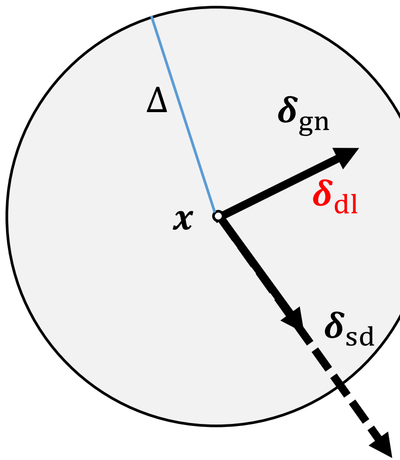
高斯牛顿法$\boldsymbol{\delta}_{\mathrm{gn}}$

$$\boldsymbol{J}^{\mathrm{T}}\boldsymbol{J}\boldsymbol{\delta}_{\mathrm{gn}}=-\boldsymbol{J}^{\mathrm{T}}\boldsymbol{f}$$

对应代码中 $Hx=b$

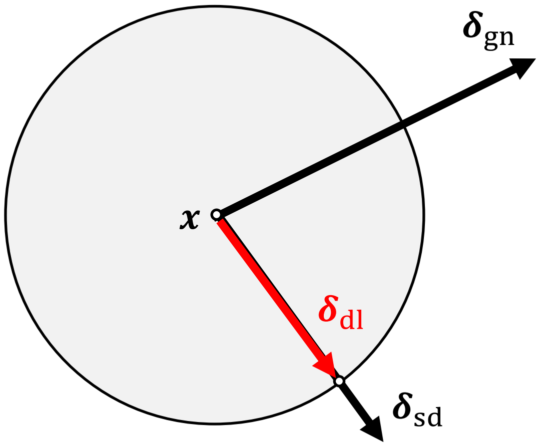
更好的优化策略

② 选取合适的Dog Leg步长$\boldsymbol{\delta}_{\mathrm{dl}}$



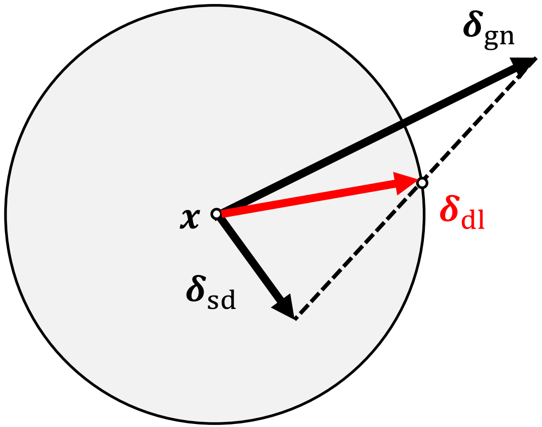
若$\|\boldsymbol{\delta}_{\mathrm{gn}}\| \le \Delta, \|\boldsymbol{\delta}_{\mathrm{sd}}\| \in \mathrm{R}$

$$\boldsymbol{\delta}_{\mathrm{dl}} = \boldsymbol{\delta}_{\mathrm{gn}}$$



若$\|\boldsymbol{\delta}_{\mathrm{gn}}\| > \Delta, \|\boldsymbol{\delta}_{\mathrm{sd}}\| > \Delta$

$$\boldsymbol{\delta}_{\mathrm{dl}} = \frac{\boldsymbol{\delta}_{\mathrm{sd}}}{\|\boldsymbol{\delta}_{\mathrm{sd}}\|}\Delta$$



若$\|\boldsymbol{\delta}_{\mathrm{gn}}\| > \Delta, \|\boldsymbol{\delta}_{\mathrm{sd}}\| \le \Delta$

$$\boldsymbol{\delta}_{\mathrm{dl}} = \boldsymbol{\delta}_{\mathrm{sd}} + \alpha(\boldsymbol{\delta}_{\mathrm{gn}} - \boldsymbol{\delta}_{\mathrm{sd}})$$

更好的优化策略

深蓝学院 shenlanxueyuan.com

② 选取合适的Dog Leg步长$\boldsymbol{\delta}_{\mathrm{dl}}$[1]

$$\boldsymbol{\delta}_{\mathrm{dl}} = \begin{cases} \kappa\boldsymbol{\delta}_{\mathrm{sd}} & 0 \le \kappa \le 1 \\ \boldsymbol{\delta}_{\mathrm{sd}} + (\kappa - 1)(\boldsymbol{\delta}_{\mathrm{gn}} - \boldsymbol{\delta}_{\mathrm{sd}}) & 1 \le \kappa \le 2 \end{cases}$$

$if\ \|\boldsymbol{\delta}_{\mathrm{gn}}\| \le \Delta$

$$\kappa = 2$$

$else\ if\ \|\boldsymbol{\delta}_{\mathrm{sd}}\| \ge \Delta$

$$\kappa = \frac{\Delta}{\|\boldsymbol{\delta}_{\mathrm{sd}}\|}$$

$else\ \|\boldsymbol{\delta}_{\mathrm{sd}}\| < \Delta$

$$\kappa = ? \qquad \|\boldsymbol{\delta}_{\mathrm{sd}} + (\kappa - 1)(\boldsymbol{\delta}_{\mathrm{gn}} - \boldsymbol{\delta}_{\mathrm{sd}})\| = \Delta$$

[1] M. L. A. Lourakis and A. A. Argyros, "Is Levenberg-Marquardt the most efficient optimization algorithm for implementing bundle adjustment?," Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1, Beijing, 2005, pp. 1526-1531 Vol. 2, doi: 10.1109/ICCV.2005.128.

更好的优化策略

深蓝学院 shenlanxueyuan.com

③ 更新信赖域半径Δ[2]

原函数值变化

下降比 $$\rho = \frac{F(\boldsymbol{x}+\Delta\boldsymbol{x}) - F(\boldsymbol{x})}{m_k(\boldsymbol{x}+\Delta\boldsymbol{x}) - m_k(\boldsymbol{x})}$$

信赖域内二阶近似函数值变化

$if\ \rho \le 0.25$

$$\Delta = \frac{\Delta}{4}$$

近似效果不好，缩小信赖域

$else\ \ \rho \ge 0.75$

$$\Delta = \max\{\Delta, 3\|\boldsymbol{\delta}_{\mathrm{dl}}\|\}$$

近似效果较好，扩大信赖域

[2] J. Dennis and R. Schnabel. Numerical Methods for Unconstrained Optimization and Nonlinear Equations. Classics in Applied Mathematics. SIAM Publications, Philadelphia, 1996.

更快的 MakeHessian

Hessian矩阵的拼接过程非常适合并行计算

记录矩阵拼接函数调用**总次数**和拼接部分的**总时间**，对比平均值

方法：

① OpenMP 最便捷，相关博客也最多。

http://supercomputingblog.com/openmp/openmp-tutorial-the-basics/

② SSE Instruction Set

http://supercomputingblog.com/optimization/getting-started-with-sse-programming/

③ Nvidia CUDA 较复杂，可参考CUDA官方文档

https://cuda-tutorial.readthedocs.io/en/latest/tutorials/tutorial01/

深蓝学院
shenlanxueyuan.com

祝大家牛年快乐，万事胜意!