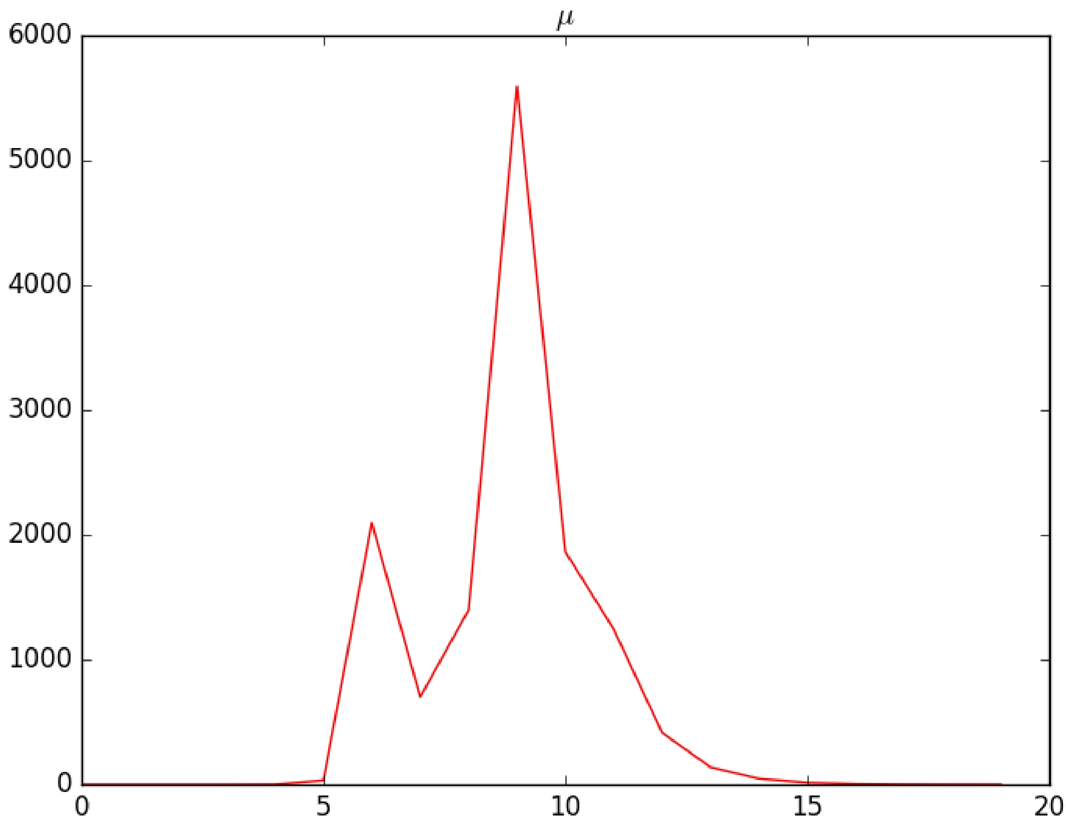
手写 VIO Report 3

Name: Zhang Qingqi
ID Number: Qingai
25 July 2020

1 绘制样本代码中 μ 的变化曲线



Figuur 1: μ 曲线图

2 实现论文中的三种阻尼因子更新策略, 并完成参数估计

2.1 论文中的 3 种更新策略

4.1.1 Initialization and update of the L-M parameter, λ, and the parameters $\mathbf{p}$

In `lm.m` users may select one of three methods for initializing and updating λ and $\boldsymbol{p}$.

1. $\lambda_0 = \lambda_o$; λ_o is user-specified [8].
   use eq'n (13) for $\boldsymbol{h}_{\mathsf{lm}}$ and eq'n (16) for ρ
   if $\rho_i(\boldsymbol{h}) > \epsilon_4$: $\boldsymbol{p} \leftarrow \boldsymbol{p} + \boldsymbol{h}$; $\lambda_{i+1} = \max[\lambda_i / L_\downarrow, 10^{-7}]$;
   otherwise: $\lambda_{i+1} = \min[\lambda_i L_\uparrow, 10^7]$;

2. $\lambda_0 = \lambda_o \max\left[\mathsf{diag}[\boldsymbol{J}^\mathsf{T}\boldsymbol{W}\boldsymbol{J}]\right]$; λ_o is user-specified.
   use eq'n (12) for $\boldsymbol{h}_{\mathsf{lm}}$ and eq'n (15) for ρ
   $\alpha = \left(\left(\boldsymbol{J}^\mathsf{T}\boldsymbol{W}(\boldsymbol{y} - \hat{\boldsymbol{y}}(\boldsymbol{p}))\right)^\mathsf{T} \boldsymbol{h}\right) / \left((\chi^2(\boldsymbol{p}+\boldsymbol{h}) - \chi^2(\boldsymbol{p}))/2 + 2\left(\boldsymbol{J}^\mathsf{T}\boldsymbol{W}(\boldsymbol{y} - \hat{\boldsymbol{y}}(\boldsymbol{p}))\right)^\mathsf{T} \boldsymbol{h}\right)$;
   if $\rho_i(\alpha\boldsymbol{h}) > \epsilon_4$: $\boldsymbol{p} \leftarrow p + \alpha\boldsymbol{h}$; $\lambda_{i+1} = \max[\lambda_i/(1+\alpha), 10^{-7}]$;
   otherwise: $\lambda_{i+1} = \lambda_i + |\chi^2(\boldsymbol{p}+\alpha\boldsymbol{h}) - \chi^2(\boldsymbol{p})|/(2\alpha)$;

3. $\lambda_0 = \lambda_o \max\left[\mathsf{diag}[\boldsymbol{J}^\mathsf{T}\boldsymbol{W}\boldsymbol{J}]\right]$; λ_o is user-specified [9].
   use eq'n (12) for $\boldsymbol{h}_{\mathsf{lm}}$ and eq'n (15) for ρ
   if $\rho_i(\boldsymbol{h}) > \epsilon_4$: $\boldsymbol{p} \leftarrow \boldsymbol{p} + \boldsymbol{h}$; $\lambda_{i+1} = \lambda_i \max[1/3, 1 - (2\rho_i - 1)^3]$; $\nu_i = 2$;
   otherwise: $\lambda_{i+1} = \lambda_i \nu_i$; $\quad \nu_{i+1} = 2\nu_i$;

For the examples in section 4.4, method 1 [8] with $L_\uparrow \approx 11$ and $L_\downarrow \approx 9$ exhibits good convergence properties.

Figuur 2: μ 更新策略

2.2 参数估计结论

在 *backend/problem.cc* 的 $IsGoodStepInLM()$ 函数中添加 3 种更新策略, 通过 `choose_method` 变量可以选择要执行的策略, 结论: 1,3 这 2 种方法从计算时间和结果来看无明显差别, 方法 2 计算时间远超方法 1,3, 且结果无明显提升.

```
Test CurveFitting start...
iter: 0 , chi= 719.475 , Lambda= 0.001
iter: 1 , chi= 91.395 , Lambda= 0.000111111
stop while delta_x is too small 1.2016e-07 or false_cnt reaches 10--0
problem solve cost: 0.210628 ms
   makeHessian cost: 0.099255 ms
-------After optimization, we got these parameters :
 1.61039  1.61853 0.995178
-------ground truth:
1.0,  2.0,  1.0
```

Figuur 3: method 1

```
Test CurveFitting start...
iter: 0 , chi= 719.475 , Lambda= 0.001
iter: 1 , chi= 161.183 , Lambda= 0.000599999
iter: 2 , chi= 99.1494 , Lambda= 0.000359999
iter: 3 , chi= 92.2566 , Lambda= 0.000215999
iter: 4 , chi= 91.4908 , Lambda= 0.0001296
iter: 5 , chi= 91.4057 , Lambda= 7.77598e-05
iter: 6 , chi= 91.3962 , Lambda= 4.66559e-05
stop while delta_x is too small 1.167e-05 or false_cnt reaches 10--11
problem solve cost: 0.896281 ms
   makeHessian cost: 0.315733 ms
-------After optimization, we got these parameters :
 1.60848  1.61614 0.993805
-------ground truth:
1.0,  2.0,  1.0
```

Figuur 4: method 2

```
Test CurveFitting start...
iter: 0 , chi= 719.475 , Lambda= 0.001
iter: 1 , chi= 91.395 , Lambda= 0.000333333
stop while delta_x is too small 1.19984e-07 or false_cnt reaches 10--0
problem solve cost: 0.262198 ms
   makeHessian cost: 0.119292 ms
-------After optimization, we got these parameters :
 1.61039  1.61853 0.995178
-------ground truth:
1.0,  2.0,  1.0
```

Figuur 5: method 3

3 公式推导

3.1 推导 $f_{!5}$

$$f_{15} = \frac{\partial \delta \alpha_{b_{k+1}}}{\partial \delta b_k^g}$$

$$\alpha_{b_i b_{k+1}} = \alpha_{b_i b_k} + \beta_{b_i b_k}\delta t + \frac{1}{2} a \delta t^2$$

$$= \alpha_{b_i b_k} + \beta_{b_i b_k}\delta t + \frac{1}{4}\left(q_{b_i b_k}(a^{b_k} - b_k^a) + \underbrace{q_{b_i b_{k+1}}}_{q_{b_i b_k} \otimes \begin{bmatrix}1 \\ \frac{1}{2}\omega\delta t\end{bmatrix}}(a^{b_{k+1}} - b_k^a)\right)\delta t^2$$

推导只与此项相关

$$\omega = \frac{1}{2}\left((\omega^{b_k} - b_k^g) + (\omega^{b_{k+1}} - b_k^g)\right)$$

$$\omega = \underbrace{\frac{1}{2}(\omega^{b_k} + \omega^{b_{k+1}})}_{\bar{\omega}} - b_k^g$$

$$f_{15} = \frac{\partial \alpha_{b_{k+1}}}{\partial \delta b_k^g} = \frac{\partial \frac{1}{4} q_{b_i b_k} \otimes \begin{bmatrix}1 \\ \frac{1}{2}(\omega - \delta b_k^g)\delta t\end{bmatrix} \cdot (a^{b_{k+1}} - b_k^a)\delta t^2}{\partial \delta b_k^g}$$

$$= \frac{1}{4}\frac{\partial R_{b_i b_k}\exp([(\omega - \delta b_k^g)\delta t]_\times)(a^{b_{k+1}} - b_k^a)\delta t^2}{\partial \delta b_k^g}$$

$$= \frac{1}{4}\cdot\frac{\partial R_{b_i b_k}\exp([\omega\delta t]_\times)\underbrace{\exp([-J_r(\omega\delta t)\delta b_k^g \delta t]_\times)}_{\boxed{I + (-J_r(\omega\delta t)\delta b_k^g \delta t)}}(a^{b_{k+1}} - b_k^a)\delta t^2}{\partial \delta b_k^g}$$

$$= \frac{1}{4}\frac{\partial -R_{b_i b_{k+1}}([(a^{b_{k+1}} - b_k^g)\delta t^2]_\times)(-J_r(\omega\delta t)\delta b_k^g \delta t)}{\partial \delta b_k^g}$$

$$= -\frac{1}{4}R_{b_i b_{k+1}}([(a^{b_{k+1}} - b_k^g)\delta t^2]_\times)\cdot(-\underbrace{J_r(\omega\delta t)}_{I}\delta t)$$

$$= -\frac{1}{4}R_{b_i b_{k+1}}([(a^{b_{k+1}} - b_k^g)\delta t^2]_\times)\cdot(-\delta t)$$

Figuur 6: 公式 1

3.2 推导 g_{12}

$$g_{12} = \frac{\partial\, \alpha_{b_i b_{k+1}}}{\partial\, n_k^g}$$

$$\alpha_{b_i b_{k+1}} = \alpha_{b_i b_k} + \beta_{b_i b_k}\delta t + \frac{1}{4}\left(q_{b_i b_k}(a^{b_k} - b_k^a) + \boxed{q_{b_i b_k} \otimes \begin{bmatrix} 1 \\ \frac{1}{2}\omega\delta t \end{bmatrix} (a^{b_{k+1}} - b_k^a) \cdot \delta t^2}\right.$$

$$\omega = \frac{1}{2}\left((\bar{\omega}^{b_k} + n_k^g - b_k^g) + (\bar{\omega}^{b_{k+1}} + n_{k+1}^g - b_k^g)\right)$$

$$= \boxed{\frac{1}{2}\underbrace{(\bar{\omega}^{b_k} + \bar{\omega}^{b_{k+1}})}_{\bar{\omega}} + \frac{1}{2}n_k^g} + \frac{1}{2}n_{k+1}^g - b_k^g$$

$$g_{12} = \frac{\partial\, \alpha_{b_i b_{k+1}}}{\partial\, n_k^g} = \frac{1}{4}\,\frac{\partial\, q_{b_i b_k} \otimes \begin{bmatrix} 1 \\ \frac{1}{2}(\omega + \frac{1}{2}n_k^g)\delta t \end{bmatrix} \cdot (a^{b_{k+1}} - b_k^a) \cdot \delta t^2}{\partial n_k^g}$$

$$= \frac{1}{4}\,\frac{\partial\, R_{b_i b_k} \exp([(\omega + \frac{1}{2}n_k^g)\delta t]_\times)(a^{b_{k+1}} - b_k^a) \cdot \delta t^2}{\partial n_k^g}$$

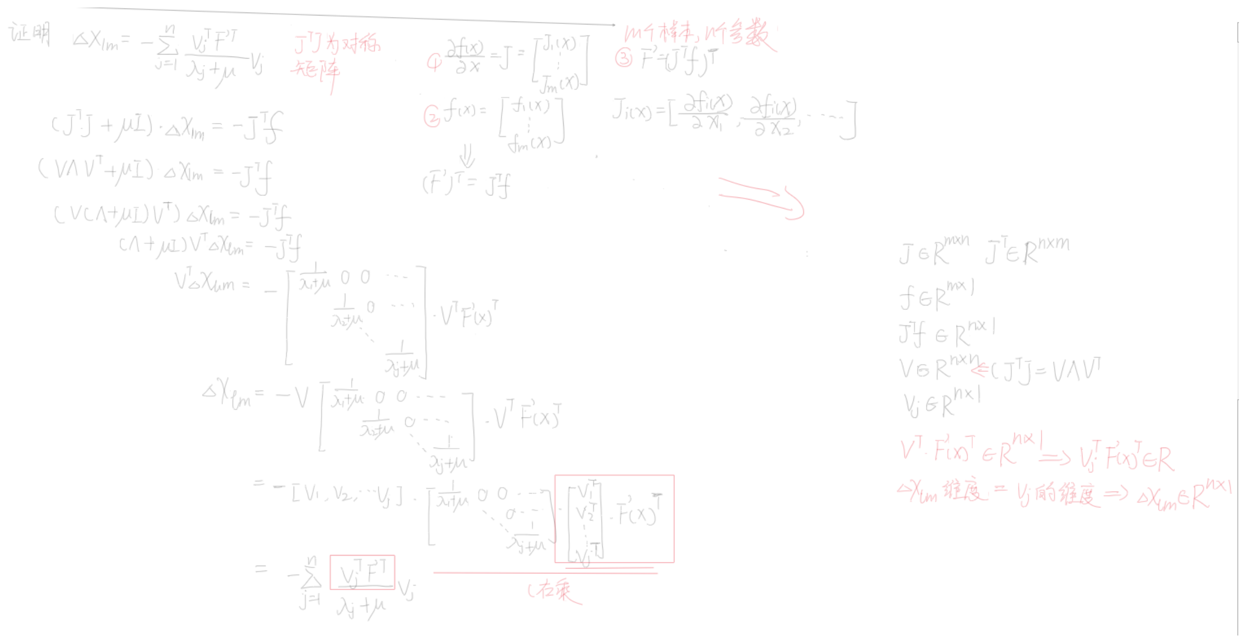
$$= \frac{1}{4} \cdot \frac{\partial\, R_{b_i b_k} \exp([\omega\delta t]_\times) \exp([J_r(\omega\delta t) \cdot \frac{1}{2}n_k^g \delta t]_\times) \cdot (a^{b_{k+1}} - b_k^a) \cdot \delta t^2}{\partial n_k^g}$$

$$= \frac{1}{4} \cdot \frac{R_{b_i b_{k+1}} [(a^{b_{k+1}} - b_k^a) \cdot \delta t^2]_\times \cdot [\overset{I}{J_r}(\omega\delta t) \cdot \frac{1}{2}n_k^g \delta t]}{\partial\, n_k^g}$$

$$= \frac{1}{4} \cdot R_{b_i b_{k+1}} [(a^{b_{k+1}} - b_k^a) \cdot \delta t^2]_\times \left(\frac{1}{2}\delta t\right)$$

Figuur 7: 公式 2

3.3 推导公式 9



Figuur 8: 公式 3

由推导过程，可知 ${v_j}^T F'^T$ 为标量可移动到 v_j 之前.