第一题：
截图如下图所示，可以看出优化后结果和**gt**差别在小数点后三位之后。

```
[100%] Built target testMonoBA
chahe@LD-OMEN:~/project/VisualImuOdometry/chapter5/build$ ./app/testMonoBA
0 order: 0
1 order: 6
2 order: 12

 ordered_landmark_vertices_ size : 20
iter: 0 , chi= 5.35099 , Lambda= 10000
iter: 1 , chi= 0.028125 , Lambda= 6666.67
iter: 2 , chi= 0.000121314 , Lambda= 4444.45
problem solve cost: 17.1135 ms
   makeHessian cost: 14.6899 ms

Compare MonoBA results after opt...
after opt, point 0 : gt 0.220938 ,noise 0.227057 ,opt 0.220953
after opt, point 1 : gt 0.234336 ,noise 0.314411 ,opt 0.234289
after opt, point 2 : gt 0.142336 ,noise 0.129703 ,opt 0.142387
after opt, point 3 : gt 0.214315 ,noise 0.278486 ,opt 0.214503
after opt, point 4 : gt 0.130629 ,noise 0.130064 ,opt 0.130576
after opt, point 5 : gt 0.191377 ,noise 0.167501 ,opt 0.191537
after opt, point 6 : gt 0.166836 ,noise 0.165906 ,opt 0.166937
after opt, point 7 : gt 0.201627 ,noise 0.225581 ,opt 0.201913
after opt, point 8 : gt 0.167953 ,noise 0.155846 ,opt 0.167979
after opt, point 9 : gt 0.21891 ,noise 0.209697 ,opt 0.218848
after opt, point 10 : gt 0.205719 ,noise 0.14315 ,opt 0.205592
after opt, point 11 : gt 0.127916 ,noise 0.122109 ,opt 0.127814
after opt, point 12 : gt 0.167904 ,noise 0.143334 ,opt 0.167902
after opt, point 13 : gt 0.216712 ,noise 0.18526 ,opt 0.216918
after opt, point 14 : gt 0.180009 ,noise 0.184249 ,opt 0.179961
after opt, point 15 : gt 0.226935 ,noise 0.245716 ,opt 0.227066
after opt, point 16 : gt 0.157432 ,noise 0.176529 ,opt 0.157563
after opt, point 17 : gt 0.182452 ,noise 0.14729 ,opt 0.182343
after opt, point 18 : gt 0.155701 ,noise 0.182258 ,opt 0.155721
after opt, point 19 : gt 0.14646 ,noise 0.240649 ,opt 0.146565
------------ pose translation ----------------
translation after opt: 0 : 4.84281e-14  2.06925e-14 -8.07495e-15 || gt: 0 0 0
translation after opt: 1 : -1.0718        4 0.866025 || gt:  -1.0718        4 0.866025
translation after opt: 2 :      -4   6.9282 0.866025 || gt:       -4   6.9282 0.866025
---------- TEST Marg: before marg------------
     100     -100        0
    -100  136.111 -11.1111
       0 -11.1111  11.1111
---------- TEST Marg: 将变量移动到右下角------------
     100        0     -100
       0  11.1111 -11.1111
    -100 -11.1111  136.111
---------- TEST Marg: after marg------------
 26.5306 -8.16327
-8.16327  10.2041
chahe@LD-OMEN: /project/VisualImuOdometry/chapter5/build$
```

Problem description:

Objective function

$$J(\boldsymbol{\theta}) \doteq \underbrace{\|\mathbf{r}^V(\boldsymbol{\theta})\|^2_{\Sigma_V}}_{\text{Visual}} + \underbrace{\|\mathbf{r}^I(\boldsymbol{\theta})\|^2_{\Sigma_I}}_{\text{Inertial}},$$

Global position and yaw(rotation around gravity and translation) are not observable

Method 1 Gauge fixation

Processing:

Fix the values of the parameter vector is equivalent to setting the corresponding columns of the jacobian of the residual vector to 0.

Method 2 Gauge prior

Processing:

Add a penalty to the objective function. But how to choose the corresponding information matrix ?

$$\|\mathbf{r}_0^P\|^2_{\Sigma_0^P}, \quad \text{where} \quad \mathbf{r}_0^P(\boldsymbol{\theta}) \doteq (\mathbf{p}_0 - \mathbf{p}_0^0,\ \Delta\boldsymbol{\phi}_{0z}).$$

$$\Sigma_0^P = \sigma_0^2\, \mathrm{I},$$

$$\|\mathbf{r}_0^P\|^2_{\Sigma_0^P} = w^P \|\mathbf{r}_0^P\|^2, \text{ with } w^P = 1/\sigma_0^2.$$

When the w =0, it is equal to free gauge.
When the w is infinitely large, it is equal to the fix gauge.

Method 3 Free Gauge

Allow the optimization to change the unobservable states freely during the iterations.

Processing:

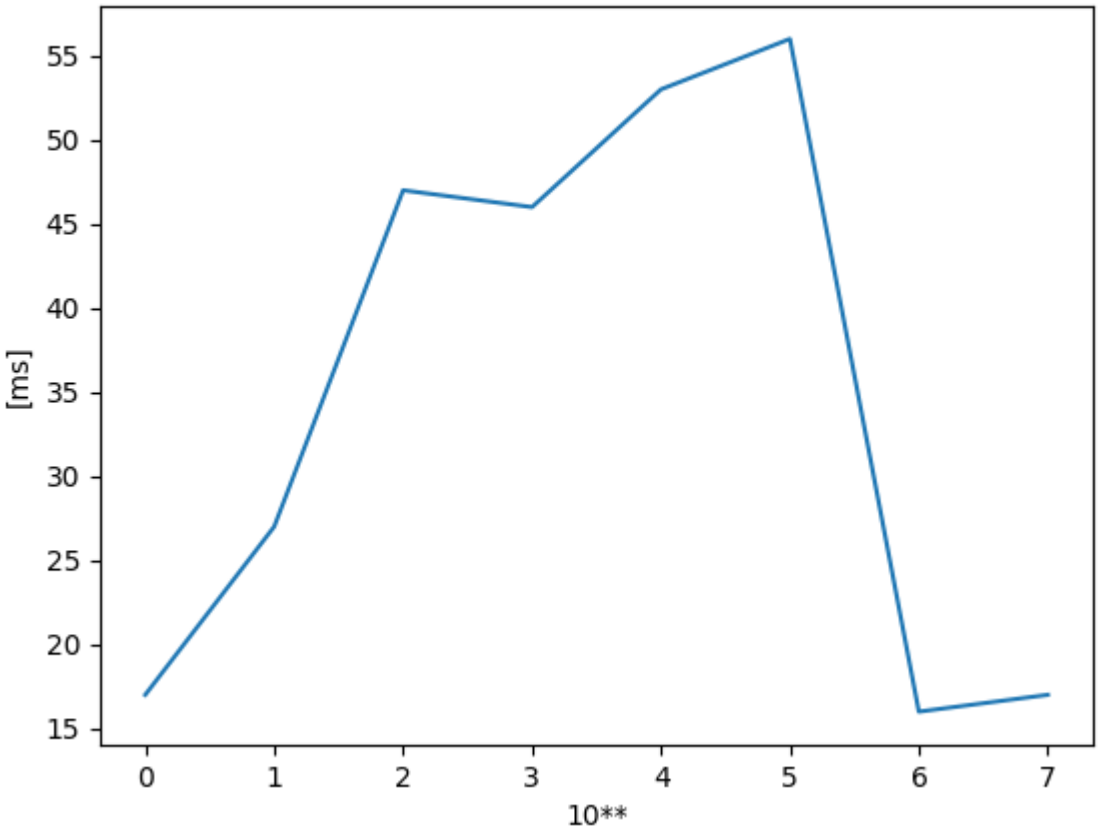
Use the pseudoinverse of the singular hessian or add some damping (Levenberg-Marquardt)

实验效果：
最后结论：

- 三种方法的精确度几乎一致
- 使用gauge prior 方法，不适当的权重会增加没必要的计算负担
- 提供适当的权重，gauge prior的效果和fix的方式一致
- Free gauge方法略快于其他两种。

精确度:
针对三种方法，包括添加含参数的先验，最后得到的准确度一致，当先验的参数权重不断增加后,RMSE会保持稳定,而在此之前，也没有明显规律，从图中观察会有震荡的变化。

第三题：
将数据统计绘图，可以看出，当weight极大时，求解时间较短，保持相对稳定。精度在这一过程没有明显变化



图：weight权重与Hessian求解计算时间关系图