12.14 Daily review：

1. 杨老师发了两位前辈的稀疏矩阵LU分解论文，fpga上实现，供学习
2. 建立了自己的gitpage blog，算初步建立

建立教程是来自知乎上某个教程

1. 找到一个前辈的学习reveiw，和自己的方向很像，值得学习https://singularitykchen.github.io/blog/2019/12/15/2019-12-09-15-weekly-review/
2. 找到一个老师指导科研的blog，是一个计算机学院的老师，有参考价值，可以多学习

<https://scientist-with-logic.github.io/%E4%B8%80%E7%AF%87%E8%AE%BA%E6%96%87%E9%87%8C%E9%9D%A2%E7%9A%84%E6%96%B9%E6%B3%95-%E5%BA%94%E8%AF%A5%E7%90%86%E8%A7%A3%E5%88%B0%E4%BB%80%E4%B9%88%E7%A8%8B%E5%BA%A6-%E5%88%AB%E4%BA%BA%E8%AE%BA%E6%96%87%E6%97%A0%E6%B3%95%E5%A4%8D%E7%8E%B0-%E5%B0%B1%E6%98%AF%E9%80%A0%E5%81%87%E4%B9%88/>

学习：

有些论文不可以复现，为什么审稿还可以过？

审稿人更加关心的是论文给大家提供的新的信息。也就是，论文给这个研究方向提供了哪些新的insight？

一个insight是一篇论文的基础。有了这个insight，这篇论文的方法才是可行的。比如说，在AirCloud那篇论文里面的insight就是：PM2.5在空间上的分布是连续的。有了这个insight，我们就可以设计出不同的模型来预测各个位置的PM2.5了

所以，我们真正需要验证和掌握的是论文里面的insight。至于这个被验证成立了，我们对于论文的理解也就达到了一个更加深入的层次。我们也就明白了为什么作者要把论文里面的方法设计成这个样子。正是作者利用了他们发现的insight，才能设计出相应的方法。

idea的背景：

求解大型稀疏线性方程组是工程计算中所经常遇到的重点问题。例如在电路分析、有限元分析、偏微分方程求解、力学计算等诸多领域，问题最终都可以归结为线性方程组的求解。以SPICE(Simulation Program with Integrated Circuit Emphasis)[1]为例,SPICE是当今集成电路设计和验证领域应用最为广泛的晶体管级电路模拟器。由于近年来超大规模集成电路的快速发展，布局后提取出的电路矩阵维度可以轻易地达到百万数量级。但另一方面，由于电路本身是物理连接，不会存在一个节点与过多电路元器件相连接的情形，因此所产生的电路矩阵必然是十分稀疏的。SPICE仿真的核心步骤就是根据提取出的电路矩阵，构建大型稀疏线性方程组，通过求解这个方程组，从而获得相应的仿真信息。

所谓的大型稀疏线性方程组求解，可以表示为如下形式的矩阵方程：

其中，为N阶矩阵，为待求解的N维列向量，为常数项，同样也是N维列向量。

LU分解，就是将一个矩阵分解为一个下三角矩阵和上三角矩阵的相乘形式：

因为L和U矩阵的特殊性，求解这两个方程耗时很短。故而整个矩阵求解过程的关键就在于如何快速高效地进行矩阵的LU分解。

LU分解中复杂而强烈的数据依赖关系使得传统的LU分解算法在面对并行计算时显得捉襟见肘，所以研究高效的并行LU分解算法，以减少求解稀疏矩阵的时间。

一个具体的研究问题：

别人是怎么做的：

别人的方法中存在什么问题：

别人的方法出现这些问题的原因是什么：

你发现了什么新的现象或者信息，并利用这一现象提出一个新的方法来避免前人的问题：

你的方法里面真正的挑战是什么：

为了克服挑战，你提出的具体的设计是什么：