[并行计算库Dask官方教程（中文翻译） - 知乎 (zhihu.com)](https://zhuanlan.zhihu.com/p/397820823)

[**https://www.modb.pro/db/213727**](https://www.modb.pro/db/213727)

**超大数据计算的两种方法：**分布式计算(内存匹配数据量)，核外计算(低成本内存最大化利用)

**-核外计算：将数据拆分为多份，**每次仅读取部分数据进入内存计算，最终将各部分的计算结果综合在一起。解决了数据量大于内存容量的问题，以时间成本降低硬件成本。

**dask封装了类似numpy和pandas的API，但底层是使用核外计算的并行式计算框架，除了使用block algorithm进行核外计算，还通过多进程的方式使用多个CPU/GPU核来加速计算。**

**-block algorithm:** 将原数据拆分为多个小集合并进行分治处理，解决了数据量大于内存容量的问题。

**dask graph:** 一定顺序任务元组(task)和值(value)构成的哈希表，实质是一种对计算图的描述

**-dask graph task:** (callable func,arguments)元组，[0]一定是callable func

某种意义上说，dask是并行式计算/分布式计算版本的numpy和pandas

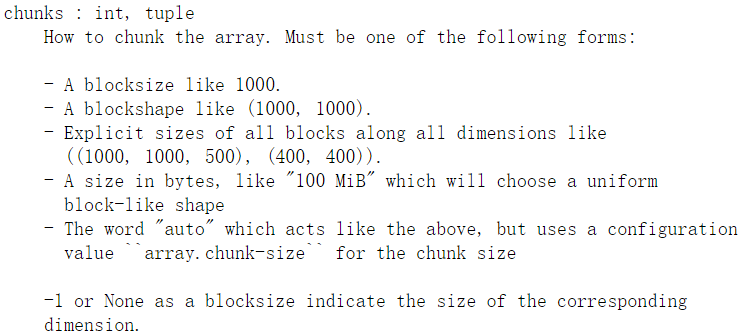
dask.array.Array

dask array是一种与numpy.array相近的容器，但在被用于运算操作时遵循block algorithm将数据拆分为多个chunk并依次进行核外计算&并行计算，且会预先生成对应的dask graph

import dask.array as da

**创建**

x = da.from\_array(array,chunks) #从array创建dask.array



x=da.from\_npy\_stack(path) #从.npy文件创建dask.array

**方法**

dask.array.Array拥有大部分numpy.array的功能

x.dask #查看dask array的dask graph

x.compute() #执行dask graph

x.chunks #chunk的形状/大小等

x.name #dask graph中结果的标签名

x.dtype #数据类型

调用计算方法如x.sum()或下标索引仅仅是生成dask graph，想要得到结果还需要执行x.compute()

由于Python Global Interpreter Lock, dask Array底层实行的是进程调度而非线程调度

dask.dataframe

import dask.dataframe as dd

df = dd.read\_csv(file\_path)

df具有pandas.DataFrame的大部分功能

**性能报告**

with performance\_report(filename="dask-report.html"):

dask操作

#会自动将执行dask操作时的存储占用、计算速度等情况写入一个html文件