

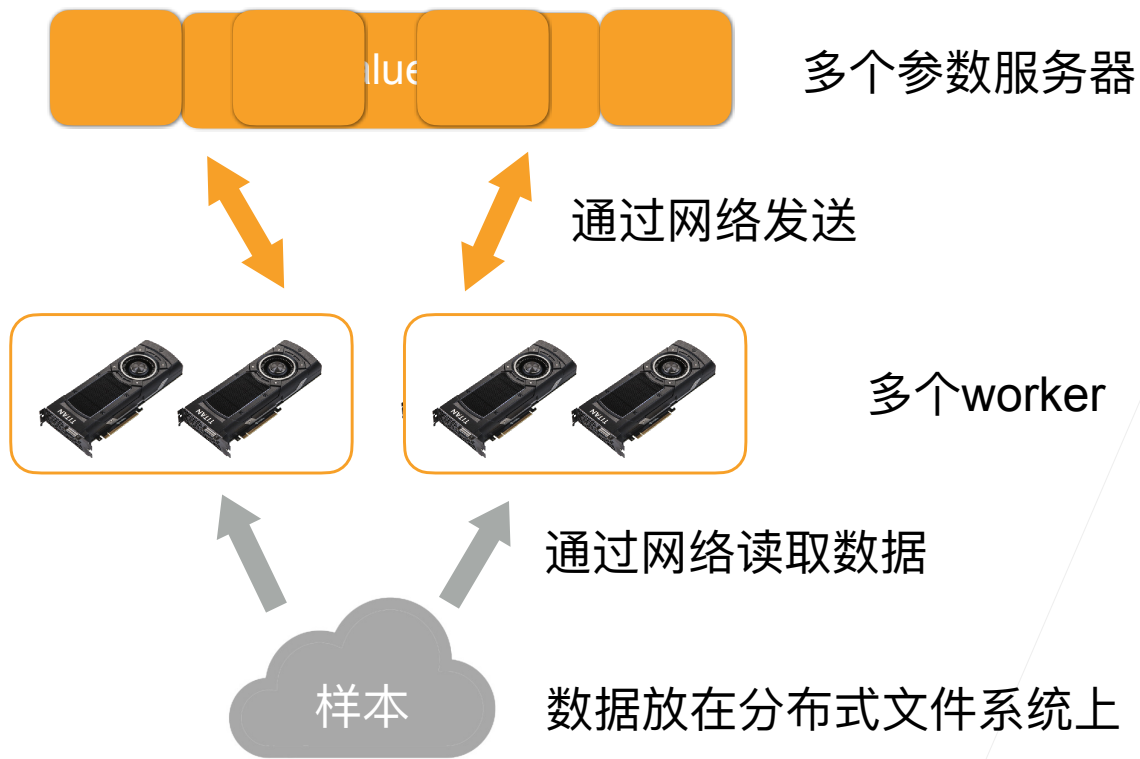


分布式计算



2015年在CMU装的高性价比集群

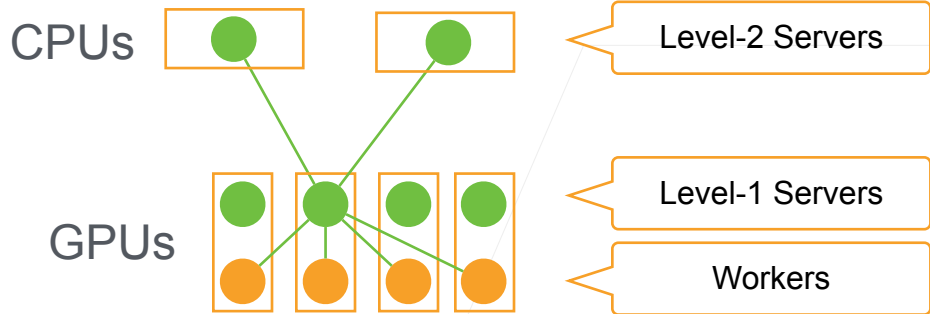
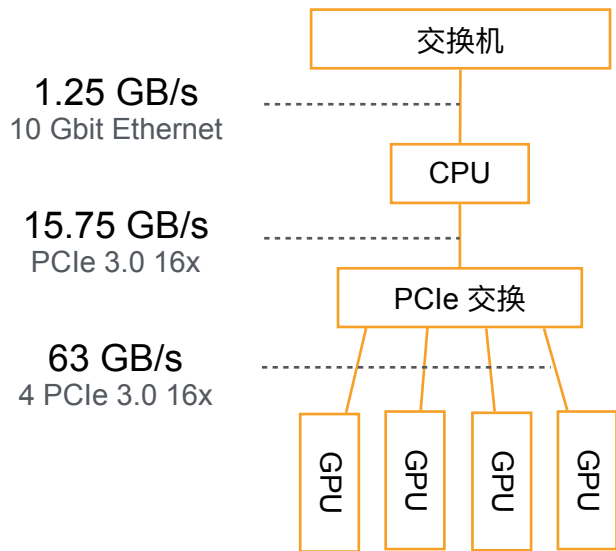
分布式计算



GPU 机器架构



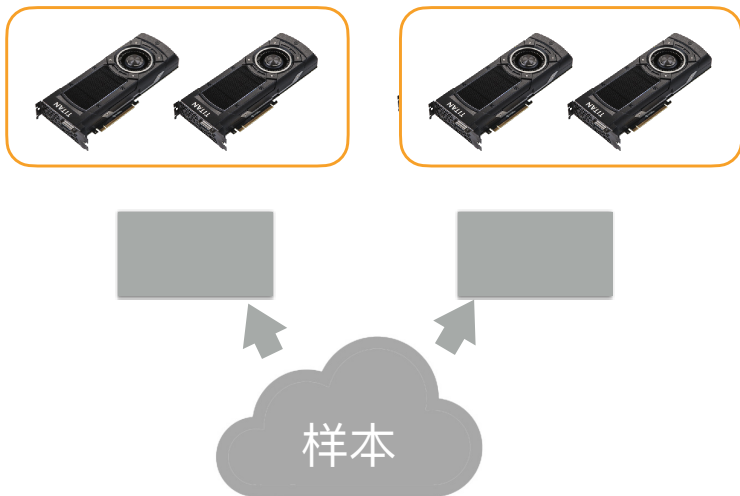
层次化的参数服务器



计算一个小批量



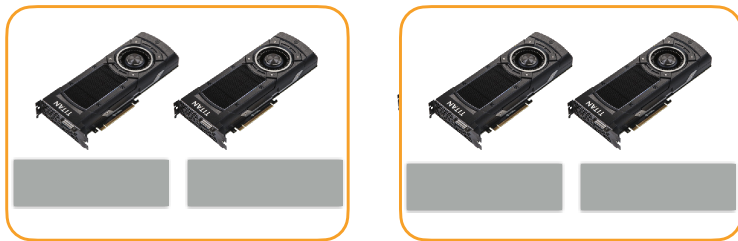
- 每个计算服务器读取小批量中的一块



计算一个小批量

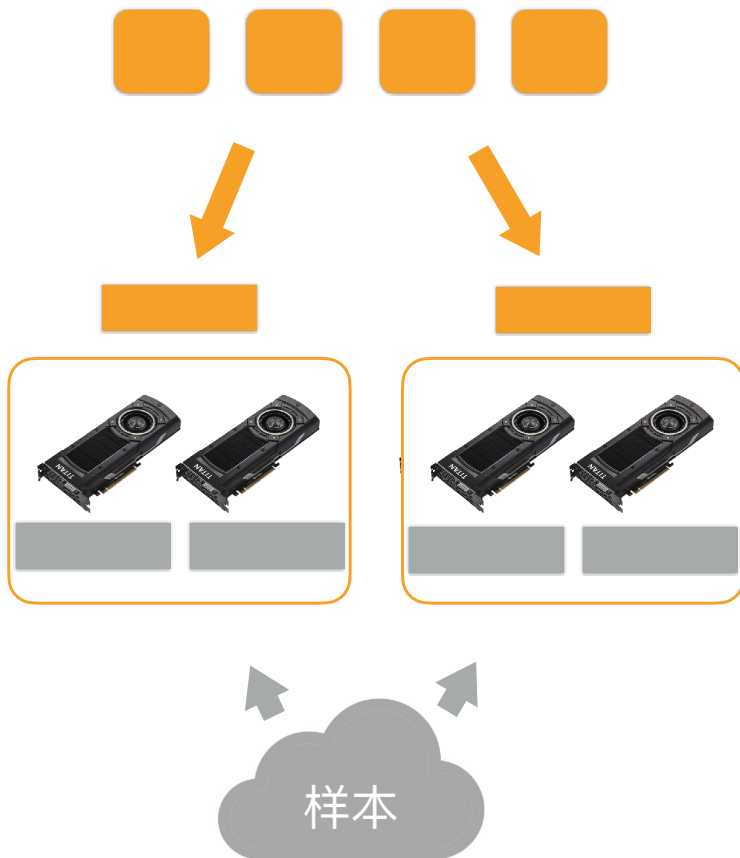


- 进一步将数据切分到每个GPU上





计算一个小批量

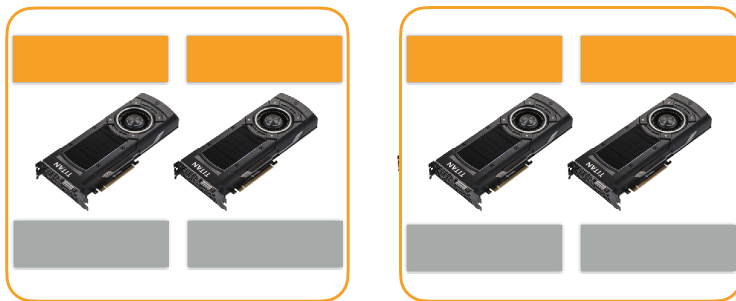


- 每个worker从参数服务器那里获取模型参数

计算一个小批量



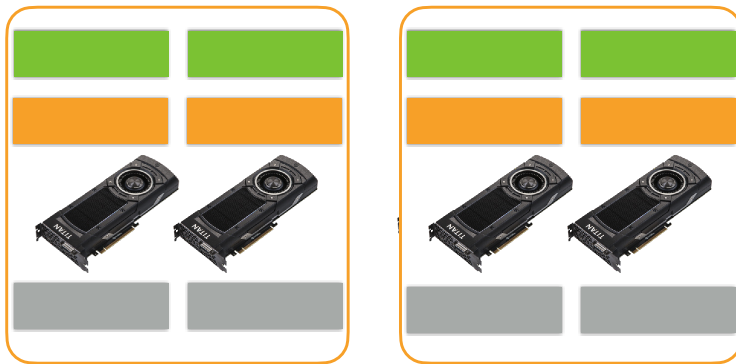
- 复制参数到每个GPU上



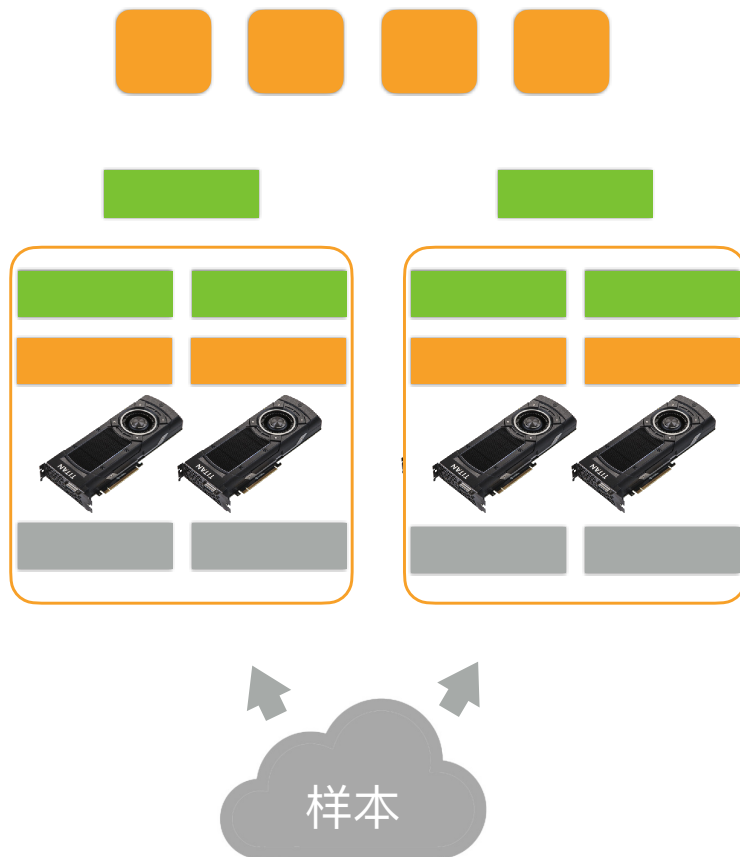
计算一个小批量



- 每个GPU计算梯度



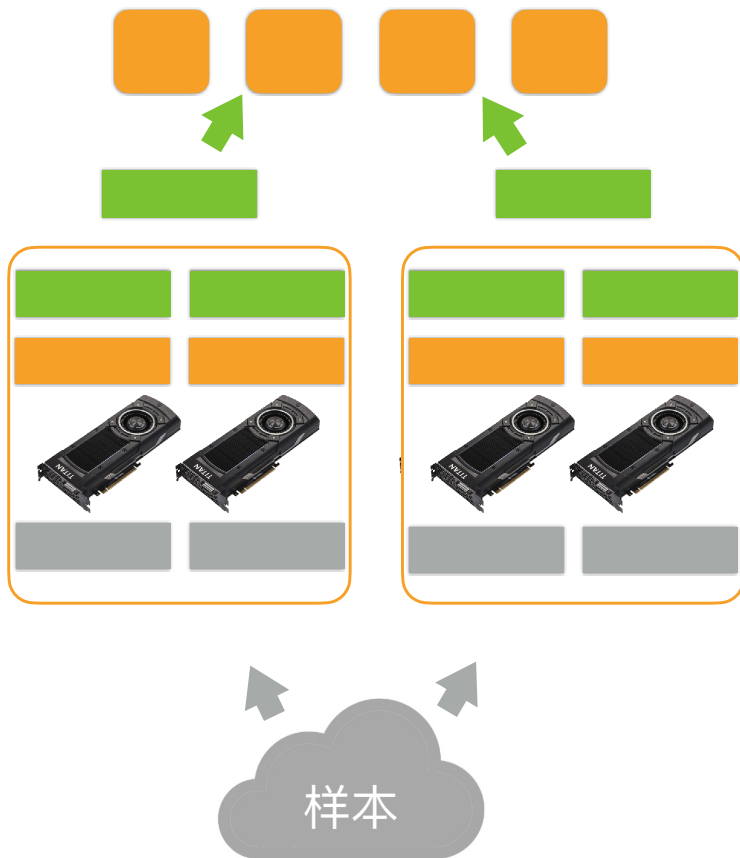
计算一个小批量



- 讲所有GPU上的梯度求和

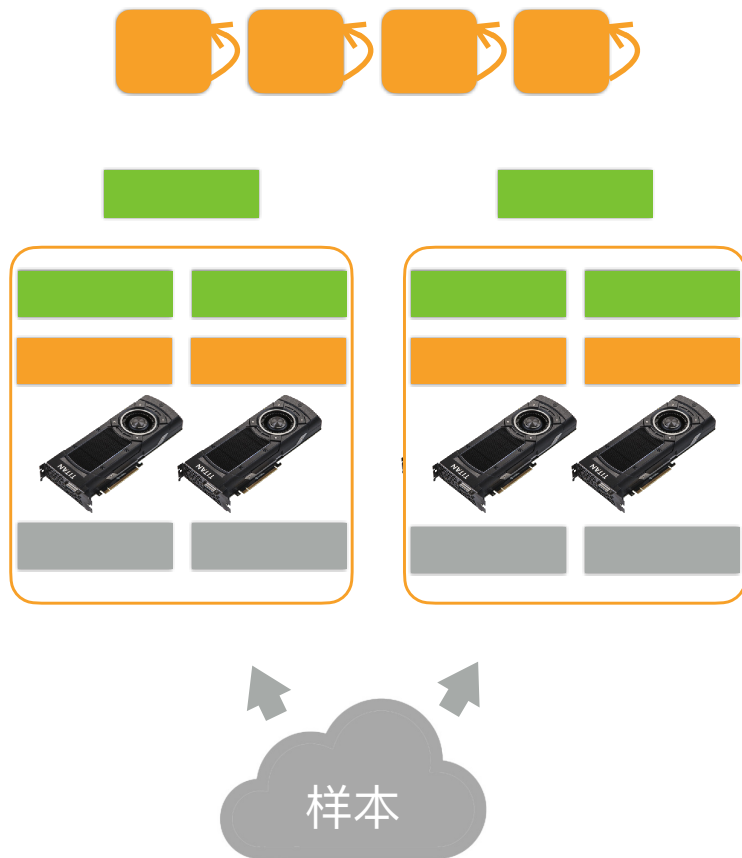


计算一个小批量



- 梯度传回服务器

计算一个小批量



- 每个服务器对梯度求和，并更新参数



同步 SGD

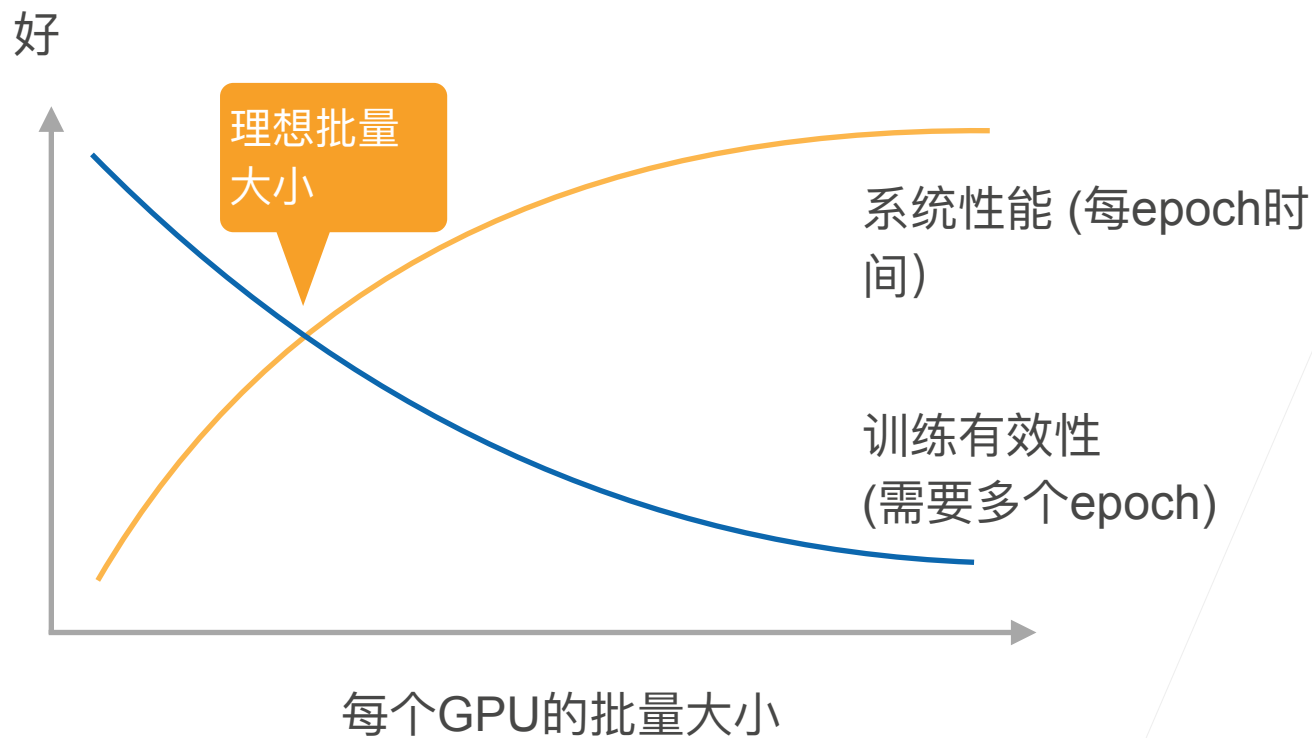
- 这里每个worker都是同步计算一个批量，称为同步SGD
- 假设有 n 个GPU，每个GPU每次处理 b 个样本，那么同步SGD等价于在单 GPU 运行批量大小为 nb 的SGD
- 在理想情况下， n 个GPU可以得到相对个单GPU的 n 倍加速



性能

- t_1 = 在单GPU上计算 b 个样本梯度时间
- 假设有 m 个参数，一个 worker 每次发送和接收 m 个参数、梯度
 - t_2 = 发送和接收所用时间
- 每个批量的计算时间为 $\max(t_1, t_2)$
 - 选取足够大的 b 使得 $t_1 > t_2$
 - 增加 b 或 n 导致更大的批量大小，导致需要更多计算来得到给定的模型精度

性能的权衡





实践时的建议

- 使用一个大数据集
- 需要好的GPU-GPU和机器-机器带宽
- 高效的数据读取和预处理
- 模型需要有好的计算（FLOP） 通讯（model size） 比
 - Inception > ResNet > AlexNet
- 使用足够大的批量大小来得到好的系统性能
- 使用高效的优化算法对对应大批量大小

总结



- 分布式同步数据并行是多GPU数据并行在多机器上的拓展
- 网络通讯通常是瓶颈
- 需要注意使用特别大的批量大小时收敛效率
- 更复杂的分布式有异步、模型并行