实验心得

16337328 郑宇骁

本次实验我主要负责P2P协议的编写。过程中遇到了许许多多的问题，包括大文件接收可能会损坏、性能低下等各种问题。全程使用socket编程让我真真切切感受到了应用层协议所做的各种各样的工作，需要留意的各种各样的细节。包括从socket中取出数据，如何保证关闭连接前取出了所有的数据这一看似很简单的问题。由于我们的文件分块都是定长的，所以我一开始采用从socket中取足一个文件段的数据才关闭连接的方法。但是很快我们发现了这一方法的局限性，就是文件的最后一个分段大概率是不到一个完整分段的长度的。于是我们又增加了通过分块号和文件总大小计算剩余应接收的文件分块大小，从而在接收最后一个分块时能够正常地关闭连接。多线程接收也是一个很有挑战性的任务，我认为P2P的优势就是利用不对称网络下行带宽大于上行带宽这一点，从多个对等方并行接收数据从而克服上行带宽的瓶颈达到较快的下载速度，而如果是从多个对等方轮流接收数据，则并不能达到这一效果。但在如何分配各个对等方下载的文件部分这一点上，我们也花费了一些心思。一开始我们准备实现与老师的PPT上相同的模型，即每个对等方为下载方提供相同大小的文件的一部分。但经过考虑，我们认为这一模型并不能达到很好的性能。因为对等方之间的速度有快有慢，会出现较快的对等方已经完成了自己那一部分的传输，而较慢的对等方还未完成传输。这时，较快的对等方只能等待传输完成而不能利用自己的带宽加速传输，这无疑是一种浪费。经过思考，我们建立了当前的模型，将文件切分为较小的块，对等方只要完成当前分块的传输，调度程序立刻就会从尚未传输的分块里调配新的任务。从实验截图也可以看出来，这一模型下较快的对等方会承担更多的传输任务，从而提高了带宽的利用率。经过我们的初步测试，在有两个对等方同时提供文件的情况下，该P2P模型要比单纯的平均分配的P2P模型快上约30%，但影响传输速率的因素较多，比如我们在测试过程中发现，笔记本电脑插上电源的情况下接收文件要大大快于使用电池的情况，初步分析可能是Windows电源计划限制了电池模式下的CPU与网卡性能。基于以上原因，性能的估计只能是一个粗略的数字，但毋庸置疑的是，这种模型对性能的提升是显著的。

这次实验对我们组的所有人来说都是陌生的，我们先前没有任何课程涉及过网络编程相关的知识，因此，这次实验给我们带来了相当大的压力。但也正因为如此，这次实验带给我们的收获也是巨大的。我们可以清晰地感受到在亲手编写协议的过程中，书本上扁平的知识变成了一个个立体的，具象化的概念。感谢老师，能够给我们提供如此宝贵的实践机会。