

华中科技大学

本科生毕业设计（论文）开题报告

题 目：基于 GPU 的 SSSP 算法优化研究

院 系 计算机科学与技术学院

专业班级 CS1808

姓 名 马忠平

学 号 U201814719

指导教师 张宇

2022 年 3 月

开题报告填写要求

一、 开题报告主要内容：

1. 课题来源、目的、意义。
2. 国内外研究现况及发展趋势。
3. 预计达到的目标、关键理论和技术、主要研究内容、完成课题的方案及主要措施。
4. 课题研究进度安排。
5. 主要参考文献。

二、 报告内容用小四号宋体字编辑，采用 A4 号纸双面打印，封面与封底采用浅蓝色封面纸（卡纸）打印。要求内容明确，语句通顺。

三、 指导教师评语、教研室（系、所）或开题报告答辩小组审核意见用蓝、黑钢笔手写或小四号宋体字编辑，签名必须手写。

四、 理、工、医类要求字数在 3000 字左右，文、管类要求字数在 2000 字左右。

五、 开题报告应在第八学期第二周之前完成。

一、课题来源

社交网络正在变得无处不在，其数据量正在急剧增加。流行的在线社交网络网站，如微信、QQ、Facebook 和 Twitter 等，如今都拥有数亿活跃用户。谷歌的新社交网络 Google+吸引了 2500 万独立用户，并在推出后的第一个月以每天约 100 万访问者的速度增长。实现这些海量图形的在线和交互式查询处理，特别是快速捕获和发现实体之间的关系，正在成为从社会科学到广告和营销研究再到国土安全的新兴应用不可或缺的组成部分。

最短路径计算是管理和查询社交网络最基本但最关键的问题之一。社交网络网站 LinkedIn 开创了著名的最短路径服务"你如何连接到 A"，它提供了你和用户 A 之间的友谊链的精确描述。微软的 Renlifang (EntityCube) 记录了超过 1000 万个实体（人员，位置，组织）的十多亿个关系，允许用户检索两个实体之间的最短路径，如果它们的距离小于或等于 6。新出现的在线应用程序"Six Degrees"提供了一种互动方式来展示您如何与 Facebook 网络中的其他人建立联系。此外，最短路径计算在确定信任和发现网络游戏中的朋友方面也很有用。

图是一种重要的数据结构，用于表达现实世界中广泛存在的关联性数据，单源点最短路径 (SSSP) 是数据结构传统问题。然而随着图规模的不断增加，现有的 SSSP 算法通常面临着高额的数据访问开销、低 GPU 利用率等问题，无法满足实际需求。因此，设计一种高效的 SSSP 算法是一个急需解决的问题。

另一方面，具有多个 GPU 的节点正在成为高性能计算的首选平台，对多 GPU 高性能平台的高效应用研究也提上日程。

二、课题目的和意义

了解 SSSP (单源最短路径) 算法和 GPU 的基础相关知识，学习并了解现有的 SSSP 算法，基于现有 SSSP 算法，探索新的 SSSP 算法。

三、国内外研究现况和趋势

最短路径计算是管理和分析大型社交网络的最基本操作之一。最短路径计算最着名的方法之一是 Dijkstra 算法。它计算加权图中的单源最短路径，可以使用 $O(|E| + |V|\log|V|)$ 时间实现。如果是无权图（与许多社交网络一样），则广度优先搜索 (BFS) 过程可以计算 $O(|E| + |V|)$ 中的最短路径。但是，将这些方法应用

于具有数百万个顶点的社交网络的成本高得令人望而却步。

（1）Hub-Acccelerator 框架

虽然现有的技术对于在大型但稀疏的道路网络上找到最短路径非常有效，但社交图具有完全不同的特征：它们通常是非空间的，非加权的，无尺度的，并且它们除了巨大的尺寸之外还表现出小世界的属性。特别是，集线点（即具有大量连接的顶点）的存在会破坏搜索空间，从而使最短路径计算变得异常具有挑战性。在本文中，我们介绍了一组以集线点为中心的新技术，统称为 Hub-Acccelerator 框架，以计算 k 度最短路径（如果两个顶点的距离在 k 范围内，则在两个顶点之间找到最短路径）。这些技术使我们能够通过大大限制集线器的扩展范围（使用新颖的省距集线器网络概念）或完全修剪在线搜索中的集线器（使用 Hub2 标记方法）来显著减少搜索空间。Hub-Acccelerator 方法比 BFS 和最先进的近似最短路径方法 Sketch 快两个数量级以上。Hub-Network 方法不会引入额外的索引成本和轻度的预计算成本；Hub2-Labeling 的索引大小和索引构建成本也适中，优于或可与近似索引 Sketch 方法相媲美。

Hub-Acccelerator 的设计旨在利用节点枢纽进行最短路径计算，而无需完全扩展其邻域。Hub-Acccelerator 方法平均比 BFS 和最先进的近似最短路径方法快两个数量级以上。

（2）Groute 编程模型

具有多个 GPU 的节点正在成为高性能计算的首选平台。但是，大多数应用程序都是使用批量同步编程模型编写的，这对于受益于低延迟异步通信的不规则算法可能不是最佳选择。有人提出了异步多 GPU 编程的结构，并描述了它们在称为 Groute 的精简运行时环境中的实现。Groute 还实现了通用的集合操作和分布式工作列表，无需大量编程工作即可开发不规则的应用程序。我们证明，这种方法实现了最先进的性能，并为 8-GPU 和异构系统上的一套不规则应用程序提供了强大的扩展能力，使某些算法的加速速度提高了 7 倍以上。

Groute 编程模型提供了多种构造来促进异步多 GPU 编程。通过在标准 C++ 和 CUDA 上实现精简运行时环境可实现 Groute 编程模型，从而实现异步多 GPU 编程。环境由三层组成，如图 1 所示。底层包含节点拓扑和 GPU 间通信的低级管理，中间层实现 Groute 通信构造，使用拓扑优化内存传输路径，顶层实现异步

常规和不规则应用中常用的高级操作。

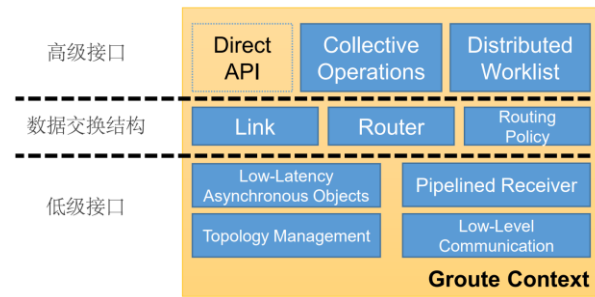


图 1 Groute 库

表 1 列出了这些构造及其编程接口的简述。

表 1 Groute 编程接口

构造	描述
基本构造	
Context	表示运行时的单例
Endpoint	可以通信的实体（例如 GPU、CPU、路由器）
Segment	封装缓冲区、其大小和元数据的对象。
通讯设置	
Link (Endpoint src, Endpoint dst, int packetsize, int numbuffers)	将 src 连接到 dst，使用多缓冲区和 numbuffers 缓冲区和数据包大小的数据包。
Router (int numinputs, int numoutputs, RoutingPolicy policy)	多个端点连接在一起，实现动态通信。
通讯调度	
EndpointList RoutingPolicy (Segment message, Endpoint source, EndpointList routerdst)	程序员定义的函数，用于根据发送终结点和路由器目标终结点的列表确定可能的消息目标。路由器将按可用性选择目的地。
异步对象	
PendingSegment	当前正在接收的段。
DistributedWorklist (Endpoint src, EndpointList workers)	管理由路由器和每 GPU 链路组成的全对全工作项分发。

Groute 应用程序的运行包括两个阶段：数据流图构造和异步计算。

Groute 程序首先指定计算的数据流图。这个有向图中的节点（我们称之为端点）表示（i）CPU 和 GPU 等物理设备，或（ii）称为路由器的虚拟设备，它们是实现复杂通信模式的抽象节点。数据流图中的边缘表示端点之间的通信链路，只要没有自循环（端点直接连接到自身）或多图中的多个相同边（即路由

器只能有一个到同一端点的传出边), 就可以创建边。注意, 为了支持多任务处理, 可以从同一物理设备创建多个虚拟终结点。

发送和接收方法允许端点在链路上发送和接收数据;收到数据后, 端点可以使用回调对其执行操作。创建路由器时, 程序员会指定路由策略, 以确定接收到输入时的行为。例如, 输入可以发送到单个终结点, 也可以根据其可用性发送到终结点的子集。创建链路时, 将指定该链路的分组和多缓冲策略。

四、课题目标

本课题主要关注 GPU 硬件的充分联合利用和高效率单源点最短路径求解算法的研究实现及优化。

五、问题描述

常见的图描述即可用于描述 SSSP 问题。

定义图 $G=(V,E)$, 其中 V 为节点集, $E=\{(v_i,v_j)|v_i,v_j\in V, |v_i,v_j|=w\}$ 为边集, 其中 w 为边权值 (对于无权图, $w=1$)。定义 $\text{dist}(s,v)$, 表示从 s 经过若干条边到达 v 节点所经过的所有边的权值的和。单源点路径问题的目标是求解对于特定 s 和每个 v ($v\in V$) 的最小的 $\text{dist}(s,v)$ 以及所经过的边的序列。

六、研究手段

首先是文献研究, 阅读文献并从中获得启发是非常重要的部分, 无论是算法思想还是硬件架构模式都能从相关文献中获得。两篇参考文献将作为此次毕设的重要文献参考, 当然, 这不是说仅限于使用该两篇文献资料。在算法研究过程中, 应该根据所遇到的问题或者一些新的想法, 积极利用各种检索工具以及图书馆系统, 查阅更多相关的文献资料, 以期针对相关知识获得更深入和更广泛的理解。

研究的主要方向为算法的设计优化与实现, 因此经常性的实验是必不可少的过程, 每一次实验的结果将作为算法设计优化的根据。

最后是算法实现过程中每一个阶段的总结和下一步的规划表, 为避免时间和算力资源的浪费, 需要对实验步骤进行实验前的规划和试验后的总结。为了使毕业设计的时间线明确清晰, 拟将每一时间段的毕设完成情况和相关总结规划同步到远端云库中, 这样也便于版本控制和发生意外后数据的恢复。

七、风险分析

风险来源一：对相关知识了解尚不深入，对毕设尚未有一个明确大体的把握。由于真正开始毕设的时间并不是很长，尚未投入必要的时间，对相关文献的理解还不够深入。除此之外，可能还需要其他知识的补充。如果在进行毕设的过程中发现了需要投入时间进行补充的理论知识，则需要尽快入手相关知识教材，调整毕设时间表，尽快完成欠缺知识的学习。在毕设过程中应经常与导师保持联系，积极向导师请教和交流。

风险来源二：测试时可能缺乏硬件设备。本课题需要多 GPU 架构计算机的支持，不可能随时获得实验机会，因此需要在测试前对实验进行完善的规划和设计，在较少的实验次数的情况下尽量保证测试的完备性。

。

八、课题研究进度规划

课题的研究进度安排表如下。

表 2 课题研究进度安排表

时间	工作任务
2022 年 3 月	研究相关文献，深入理解相关知识，对文献中的算法进行初步实现。对于涉及的陌生知识，安排时间进行自主学习。
2022 年 4 月	在实现算法的基础上深入研究，查阅资料，提出并总结归纳可行的优化方向。对算法进行初步优化，根据优化后的运行结果决定下一步的优化方向。
2019 年 5 月	对算法进行进一步优化，完成最终的程序设计编写，并实验得出最终结果。书写最终报告，准备最终答辩

上述进度安排仅仅是大致的时间计划表，在进行实验的每一个阶段，都应该将时间表进行细化，最好能细化到每一周甚至每一天。细化后的时间表将在云端仓库实时更新。

九、主要参考文献

- [1] Ruoming Jin et al. Hub-Accelerator: Fast and Exact Shortest Path Computation in Large Social Networks[J]. CoRR, 2013, abs/1305.0507.
- [2] Ben-Nun T , Sutton M , Pai S , et al. Groute: Asynchronous Multi-GPU Programming Model with Applications to Large-scale Graph Processing[J]. ACM Transactions on Parallel Computing, 2020, 7(3):1-27.

华中科技大学本科生毕业设计（论文）开题报告评审表

姓名	马忠平	学号	U201814719	指导教师	张宇
院（系）专业	计算机科学与技术学院计算机科学与技术				
<div>指导教师评语</div> <div>1. 学生前期表现情况。</div> <div>2. 是否具备开始设计（论文）条件？是否同意开始设计（论文）？</div> <div>3. 不足及建议。</div>					
<div>指导教师（签名）：</div> <div>年 月 日</div>					
教研室（系、所）或开题报告答辩小组审核意见					
<div>教研室（系、所）或开题报告答辩小组负责人（签名）：</div> <div>年 月 日</div>					