Fall 2012
Georgia Tech, CSE 8803-MGA
Tu/Th 12:05pm - 1:25pm, Classroom: CoC Bldg. 102

Massive Graph Analysis

http://www.cc.gatech.edu/~bader/COURSES/GATECH/CSE8803-MGA-Fall2012/

Instructor: Prof. David A. Bader, KACB 1320, 404-385-0004, bader@cc
Office Hours: Tuesday 1:30pm-2:30pm
Teaching Assistant: TBD, TBD@gatech.edu
TA Office Hours: (TBD) Monday/Friday 11:30am - 12:30pm, Klaus Room 1343

Course Description:

Emerging real-world graph problems include detecting community structure in large social networks, improving the resilience of the electric power grid, and detecting and preventing disease in human populations. Unlike traditional applications in computational science and engineering, solving these problems at scale often raises new challenges because of sparsity and the lack of locality in the data, the need for additional research on scalable algorithms and development of frameworks for solving these problems on high performance computers, and the need for improved models that also capture the noise and bias inherent in the torrential data streams. In this course, students will be exposed to the opportunities and challenges in massive data-intensive computing for applications in computational biology, genomics, and security.

This course will introduce students to designing high-performance and scalable algorithms for massive graph analysis. The course focuses on algorithm design, complexity analysis, experimentation, and optimization, for important "big data" graph problems. Students will develop knowledge and skills concerning:

- the design and analysis of massive-scale graph algorithms employed in real-world data-intensive applications, and
- performance optimization of applications using the best practices of algorithm engineering.

Pre-requisites: design and analysis of algorithms (CS 3510).

Grading:

(25 %)	Midterm
(25 %)	Final
(25 %)	Project
(20 %)	Homework
(5 %)	Class participation

CLASS POLICIES

1. Class announcements will be sent to the Georgia Tech T-Square mailing list, see http://t-square.gatech.edu/.

2. Please let me know as soon as possible if you will need to re-schedule an exam, or have any special needs during the semester.

3. Each student must read and abide by the Georgia Tech Academic Honor Code, see www.honor.gatech.edu.

4. Plagiarizing is defined by Webster's as "to steal and pass off (the ideas or words of another) as one's own: use (another's production) without crediting the source." If caught plagiarizing, you will be dealt with according to the GT Academic Honor Code.

5. All homework must be submitted on-time through T-Square. Homework is due by 5PM on the given due date. Late homeworks will not be accepted without a legitimate excuse and approval from the instructor.

6. When working on homework, you may work with other students in the class. However, each student must upload their own copy of the homework to T-Square with the collaborators names annotated on every copy of the submission.

7. No collaboration is permitted on exams. The midterm and final exams will be in-class, closed-book exams. You will be allowed to take a "cheat sheet" (double-sided 8.5 x 11 sheet of paper) into each exam.

8. Unauthorized use of any previous semester course materials, such as tests, quizzes, homework, projects, and any other coursework, is prohibited in this course. Using these materials will be considered a direct violation of academic policy and will be dealt with according to the GT Academic Honor Code.

Coverage of Topics

An increasingly fast-paced, digital world has produced an ever-growing volume of petabyte-sized datasets. At the same time, terabytes of new, unstructured data arrive daily. As the desire to ask more detailed questions about these massive streams has grown, parallel software and hardware have only recently begun to enable complex analytics in this non-scientific space.

In this course, we will discuss the open problems facing us with analyzing this "data deluge". Students will learn the design and implementation of algorithms and data structures capable of analyzing spatio-temporal data at massive scale on parallel systems. Students will understand the difficulties and bottlenecks in parallel graph algorithm design on current systems and will learn how multithreaded and hybrid systems can overcome these challenges. Students will gain hands-on experience mapping large-scale graph algorithms on a variety of parallel architectures using advanced programming models.

(Course co-designed with E. Jason Riedy and David Ediger).

Network Analysis

1. Introduction to Graph Theory & Data Structures
2. Motivating Applications in Data
3. Opportunities & Challenges
4. Parallel, Multicore, & Multithreaded Architectural Support for Graph Processing
5. Mapping Graph Algorithms to Architectures
6. Open Discussion

Static Parallel Algorithms

1. Programming Models
2. Parallel Prefix & List Ranking
3. Graph Search, Spanning Tree, Connected Components
4. Minimum Spanning Tree Matroid Algorithms
5. Social Networking Algorithms
6. Betweenness Centrality
7. Community Detection

Dynamic Parallel Algorithms

1. Streaming Data Analysis
2. Data Structures for Streaming Data
3. Tracking Clustering Coefficients
4. Tracking Connected Components
5. Anomaly Detection

Programming & Software

1. Programming Environments: OpenMP, MPI, MapReduce, CUDA/OpenCL, UPC, X10
2. Graph Libraries: PBGL, MTGL, LEDA, STXXL, SNAP, GraphCT, STING, Pregel
3. Advanced Topics

Fall 2012, Tentative Course Schedule

Week	Date	Lec	Topic
1	21 Aug	1	Motivation for Massive-Scale Graphs
	23 Aug	2	Definitions, Data Structures, Graph Algorithms
2	28 Aug	3	Graph Traveral and Breadth-First Search
	30 Aug	4	Spanning Trees and Connected Components
3	4 Sep	5	Multicore and Multithreaded Parallel Computers
	6 Sep	6	Mapping Parallel Graph Algorithms to Architectures, Cost Models
4	11 Sep	7	Streaming Graph Analysis and STINGER (Guest Lec.: J. Riedy)
	13 Sep	8	Tracking Clustering Coeff. and Conn. Components (Guest Lec.: J. Riedy)
5	18 Sep	9	Parallel Breadth-First Search of Massive Graphs (Guest Lec.: D. Ediger)
	20 Sep	10	Parallel Betweenness Centrality (Guest Lec.: D. Ediger)
6	25 Sep	11	Map-Reduce Algorithms, Hadoop
	27 Sep	12	Graph Twiddling and other analytics on Map-Reduce
7	2 Oct	13	Web-scale malware detection & fraud detection (Guest Lecturer: P. Chao)
	4 Oct	14	Interactive graph exploration & mining (Guest Lecturer: P. Chao)
8	9 Oct	15	Anomaly Detection in Massive Graphs
	11 Oct	16	**Midterm**
9	16 Oct	-	**Fall Break**
	18 Oct	17	Massive-Graphs in Computational Biology, Genome Assembly
10	23 Oct	18	Finding Community Stucture in Large Graphs
	25 Oct	19	Modularity and Conductance
11	30 Oct	20	Modularity Algorithms, Normalizations, and Resolution Limit
	1 Nov	21	Parallel Modularity Approaches
12	6 Nov	22	Clustering in Weighted Networks
	8 Nov	23	Graph Databases and Complex Queries
13	13 Nov	24	Project Presentations, Part 1
	15 Nov	25	Project Presentations, Part 2
14	20 Nov	26	Hybrid Graph Approaches for Massive-Scale
	22 Nov	-	**Thanksgiving Break**
15	27 Nov	27	Analyzing Open Source Data Streams (e.g. Twitter)
	29 Nov	28	Web Knowledge Graph
16	4 Dec	29	GPU Graph Algorithms with CUDA
	6 Dec	30	Other Graph Libraries: PBGL, MTGL, LEDA, STXXL, SNAP, Pregel
17	11 Dec	-	**Final Exam** (11:30am-2:20pm)