Anurag Chakraborty

+1 226 8997477 a8chakra@uwaterloo.ca https://www.linkedin.com/in/anu7ag/
https://anuchak.github.io/

Education

University of Waterloo **2024 – Present**
PhD, Computer Science *Waterloo, Canada*

University of Waterloo **2022 – 2024**
MMath (Thesis), Computer Science (GPA: 91.25/100) *Waterloo, Canada*
Thesis: *Robust Recursive Query Parallelization in Graph Database Management Systems*

Jadavpur University **2016 – 2020**
BE, Computer Science and Engineering (GPA: 8.48/10) *Kolkata, India*

Technical Skills

Languages: C/C++, Java, Python
Technologies/Frameworks: Linux Perf, Valgrind, Intel VTune, GDB, Apache Spark, HDFS, Docker, AWS (S3, EC2, EMR, Lambda, ECS, Fargate, API Gateway), git

Research Experience

Graduate Researcher | *Advisor: Dr. Semih Salihoglu* **September 2022 - Present**

- Currently working on Kùzu, a new DBMS focused on graph workloads being built at University of Waterloo.
- Kùzu is a single node disk-based columnar graph database management system (GDBMS). It incorporates an out-of-core buffer manager to support processing datasets larger than memory, a vectorized query execution engine with morsel-driven scheduling, multi-version concurrency control for transactions and an on-disk compressed sparse row graph index for analytics queries.
- For my masters thesis, my research focus was on efficient scheduling policies for handling recursive joins in Kùzu. I improved recursive query runtimes in Kùzu upto **12x** by implementing a hybrid morsel-driven scheduling policy that better parallelizes recursive computations. The key insight of my thesis was that significant query speed-ups can be achieved by parallelizing a single recursive computation while concurrently executing multiple recursive computations.
- To further improve performance I added optimizations such as sharing scans of buffer manager pages across recursive computations, that helps reduce I/O calls and decompression cost of pages. This led to a further improvement of **3-5x** in runtime.
- My current research focus is on adding support for graph paths as a native type in Kùzu, more efficient scheduling policies for bulk path-finding in recursive join queries and compression of intermediate results for recursive joins, which should significantly reduce memory footprint.

Undergraduate Researcher | *Advisor: Dr. Nandini Mukherjee* **July 2019 - June 2020**

- Our objective was performance enhancement of query execution based on MapReduce paradigm, for which we developed a custom filesystem: Node Guided Map-Reduce (NGMR)
- NGMR filesystem maintains an additional two level index structure at the Master & Client Node, to keep track of particular keys from large datasets. During query execution this metadata is used to reduce shuffling cost of intermediate results.
- This project was part of my Undergraduate Thesis and resulted in a *paper*.

Work Experience

Goldman Sachs **July 2020 – July 2022**
Software Engineer *Bangalore, India*

- Part of the Finance Platforms Engineering team
- Primarily worked on a Distributed Calculation Engine on top of Apache Spark
- Worked on migrating On-Prem Spark Job Server to AWS EMR with support for long-running Spark Contexts
- Migrated On-Prem Calculation Engine to Cloud (Fargate + ECS), integration with API Gateway