Rahul Goraniya

rahulgoraniya.github.io | rahulgoraniya1903@gmail.com

Summary

Computer Science graduate with research experience in hardness of approximation and parameterized algorithms, and a strong foundation in algorithms and complexity, logic, and formal languages.

Research Experience

Hardness of Approximation + Computational Geometry Oct 2024 - present

- Established a *PTAS lower bound* of the form $2^{O(1/\epsilon)^d} \cdot n^{O(1/\epsilon)^{1-\delta}}$ under the Exponential Time Hypothesis (ETH) for Matrix Tiling with $\leq$, an optimization variant of the well-known Grid Tiling with $\leq$ problem. This work formed my final-year project (ranked top 5% of cohort). This result extends the applicability of ETH-based lower bounds to approximation problems, providing a foundation for deriving hardness results for a broad range of geometric and graph-theoretic problems.
- Applied this result to derive a *tight PTAS lower bound* of the same form for Covering Points with Squares in $\mathbb{R}^2$ — a continuous covering problem where square centers can be chosen freely. To the best of my knowledge, this constitutes the first PTAS lower bound for a continuous covering problem.
- Preparing the final version of this work for submission to the Symposium on Computational Geometry (SoCG); the manuscript is nearly complete and a working draft is available on my website.

Parameterized Complexity + Algorithmic Graph Theory Jan 2025 - present

- Working with Dr. Rajesh Chitnis and his PhD students on the parameterized complexity of disjoint path problems on planar graphs. Proved W[1]-hardness for two variants of the k-Disjoint Paths problem, each by distinct fairness constraints, for both vertex-disjoint and edge-disjoint variants.
- Developed matching tight algorithms for both problem variants, establishing optimality within standard complexity assumptions.
- This project is currently being prepared for submission to SoCG as well.

Education

2022 – 2025 **B.Sc. Computer Science**, University of Birmingham

First Class Honours (4.25/4.25 GPA)

Thesis: *ETH-based hardness for approximation via L-reductions from* Matrix Tiling with $\leq$ — **Ranked in the Top 5% of Cohort** by achieving 84 marks

Relevant Modules:

Year 1: Data Structures and Algorithms (48R%), Mathematics and Logic (69%), Theories of Computation (73%)

Year 2: Functional Programming in Haskell (82%), Security and Networks (Cryptography) (84%)

Year 3: Algorithms and Complexity (79%), Programming Languages: Principles, Design and Implementation (77%), Advanced Functional Programming (Agda and Type Theory) (67%), Game Theory (80%), Final Year Project (84%)

2021 **Class 12**, St. Ann's School, Ahmedabad, India

Percentage: 91.2%

Ongoing Research Projects

Gap-creating Self-reductions

Exploring the use of self-reduction technique to transform hard decision problems into optimization problems with approximation gaps. The approach involves constructing larger instances from multiple copies of a known hard instance using combinatorial and probabilistic structures, thereby creating "gaps" that can establish hardness of approximation results from complexity assumptions without inherent gaps.

Extra Academic Preparation and Miscellaneous Activities

- Studied the *Parameterized algorithms* textbook independently, gaining a strong foundation in core topics such as kernalization, algorithm design, graph theory, and lower-bound proofs.
- Conducted introductory self-study in Homotopy Type Theory and Category Theory, which deepened my knowledge and interest in foundations of mathematics and logic.
- Regularly attended the weekly theory seminars organized by the Mathematics and Theoretical Computer Science groups at the University of Birmingham. These seminars provided exposure to different research areas and academic discussion. Notable talks included Dr. Sayan Bhattacharya's presentation on *Vizing's Theorem*, Dr. Saurav Chakraborty's work on *Distinct Elements in Streams*, and Dr. Parinya Chalermsook's overview of his research directions.
- Member of the Agda Theorem Proving Club during my final year, where we collaboratively formalized mathematical results from first principles using dependent type theory.

Skills

Programming Languages: Agda, Haskell, Java, SQL, Neo4j, C.

Tools and Technologies: LaTeX, Git.