DOI:10.1145/3623277

To view the accompanying paper, visit doi.acm.org/10.1145/3610940

Technical Perspective
Maximum Flow through a Network: A Storied Problem and a Groundbreaking Solution

By Shang-Hua Teng

IN 2022, A team of computer scientists presented a groundbreaking algorithm for the maximum flow problem: How does one transport the most supplies from a source node to a sink node in a network while respecting link capacities? This result has a wide impact on algorithmic theory because this storied problem has broad theoretical significance and practical applications.

Maximum flow is an exemplary theoretical model of a real-world scenario. In an interdisciplinary collaboration, Ted Harris, a RAND mathematician, and General Frank Ross, a former chief of the Army's Transportation Corps in Europe, aided by George Dantzig, formulated the problem when studying rail transportation in the 1950s. The flow problem is intrinsically related to the minimum-cut problem via the mathematical duality: "maximum flow is equal to minimum cut." While the flow measures how well two nodes are connected, the dual cut measures how much capacity must be destroyed to disconnect them. Both are central in optimization and have multiple fundamental applications, including bipartite matching and divide-and-conquer-based approximations. They are also tools for solving practical tasks, including image processing, DNA sequence alignment, circuit design, and finite-element simulation.

The flow problem has significant pedagogical value. It offers rich material for teaching algorithmic paradigms: greedy, iterative, multilevel, and mathematical programming. Together with von Neumann's minimax theorem for zero-sum games and Yao's minimax principle for randomized algorithms, the max-flow min-cut theorem is a vivid illustration of linear programming duality. Static in formulation and dynamic in imagination, as network models have become ubiquitous in computing, the flow/cut framework resonates with students of diverse scientific interests: from protein interaction to machine translation; supply chains to Internet economics; and statistical learning to knowledge discovery.

Network flow is a magnet for algorithmic research, attracting generations of scientists building on each other's insights. Since the first algorithms by Ford and Fulkerson, progress on these problems has been intertwined with major advances in algorithmic theory, such as shortest-path search, randomization, mathematical programming, and Laplacian solvers. Two decades ago, for example, research for min-cut led to graph sparsification, a concept that facilitated breakthrough scalable algorithms for electrical flows. This, in turn, intensified the search for faster flow/cut algorithms, which led to scalable max-flow approximation in undirected networks. Located at the intersection of discrete graph theory and continuous mathematical programming, network flow has inspired creative interplay between combinatorial and continuous methods. The new solution also exploited this interplay.

In the age of Big Data, practical applications require algorithms to be not just polynomial but scalable, that is, within a low poly-logarithmic factor of linear time. The following paper comes within striking distance of answering the outstanding question: "Does maximum flow have a scalable algorithm?"

The authors presented the paper at the FOCS plenary session on November 2, 2022. That evening, as I struggled to find words for this Technical Perspective, I caught game four of the 2022 World Series on TV. Only then did I realize I was watching a once-in-a-half-century event in baseball: four pitchers delivered the second no-hitter in World Series history. It was not lost on me that the first Fall Classic no-hitter—a perfect game by the Yankees' Don Larsen—was in 1956, the very year Ford and Fulkerson finalized their celebrated work "Maximal Flow Through a Network" at RAND. "Miracle … miracle," I heard repeated in the background, as I returned to my writing, still charmed by the serendipity. Within hours, I witnessed two hard-fought achievements in seemingly unconnected worlds, each co-authored by dedicated people with contributions from others in the field.

The no-hitter was not defined just by a single strikeout or brilliant defense. Likewise, this work is a tour de force of multiple dimensions. At the top, it introduces an insightful integration of iterative approximation with advanced data structures. Instead of aiming for faster convergence, the team took an audacious step by using a method that requires a linear number of iterations. This outer loop enjoys better network locality: It repeatedly improves flows around cycles, guided by low-stretch spanning trees, a concept formulated decades ago for on-line algorithms and instrumental for scalable Laplacian solvers. The team then uses low-stretch spanning trees to design an intricate data structure with low amortized complexity for supporting flow improvements, showing that slow and steady wins the race of maximum-flow computation.

Simplification and improvement are still needed to make this algorithm applicable in the real world. But by breaking long-standing super-linear complexity barriers, it offers hope for scalable max-flow algorithms. Each breakthrough motivates future advances. Just like results for electrical flows have accelerated the search for scalable maximum-flow algorithms, this work will expedite research on other fundamental problems, including developing almost linear-time algorithms for general linear programming. C

Shang-Hua Teng is the Seeley G. Mudd Professor of Computer Science and Mathematics at the University of Southern California. The two-time Gödel Prize recipient met then USC mathematics professor Ted Harris while attending the university in the 1980s.