Links between Join Processing and Convex Geometry

Christopher Ré
Stanford University
chrismre@cs.stanford.edu

This talk will survey some results on join processing that use inequalities from convex geometry. Recently, Ngo, Porat, Rudra, and Ré (NPRR) discovered the first relational join algorithm with worst-case optimal running time [8]. Since the seminal System R project [12], the dominant database optimizer paradigm optimizes a join query by examining each pair of joins and then combining these estimates using dynamic programming. In contrast, NPRR examines all relations *at the same time*. This change to the "one-join-at-a-time" dogma is important for performance: there are classes of queries for which *any* join-project plan is destined to be slower than the best possible run time by a polynomial factor in size of the data. NPRR's analysis makes a link from database theory to a pair of beautiful inequalities: one due to Atserias, Grohe, and Marx from computer science [2] and one due to Bollobás and Thomason from geometry [4, Theorem 2]. In a recent survey with Ngo and Rudra [9], we simplified the algorithms and the arguments for such worst-case optimal join algorithms, and I plan to present these simplified results. Additionally, I plan to describe the LeapFrog TrieJoin, a worst-case optimizer from LogicBlox [14]. Similar inequalities also underpins recent efforts to optimize join algorithms on parallel infrastructures like MapReduce [1, 3].

One possible direction for future work may be along the lines of "beyond worst-case analysis," an area that is gaining attention within theoretical computer science. Informally, beyond worst-case analysis aims to provide stronger measures of runtime optimality than the uniform measures that have traditionally been used in computer science [11]. Remarkably, "Beyond worst-case" is a place where database theoreticians have led the way: Fagin's seminal threshold algorithm is the first example of an instance optimal algorithm [6]. Inspired by this work, I will describe two recent results that use finer notions of optimality for combinatorial problems: (1) with Ngo, Ngo, and Rudra [7], we have developed geometrically inspired algorithm that has nearly instance optimal algorithms for some join queries (within an unavoidable $\log N$ factor). (2) With Sridhar et al. [13], we have used Renegar's conditioning theory [10] to solve linear programs that arise from combinatorial relaxations more efficiently than commercial systems: our analysis depends on a (geometric) notion of conditioning. Such relaxations have recently become of interest to the database community due to problems from computational advertising [5].

1. REFERENCES

[1] Foto N. Afrati and Jeffrey D. Ullman. Optimizing joins in a map-reduce environment. In *EDBT*, 2010.

[2] Albert Atserias, Martin Grohe, and Dániel Marx. Size bounds and query plans for relational joins. In *FOCS*, pages 739–748, 2008.

[3] Paul Beame, Paraschos Koutris, and Dan Suciu. Communication steps for parallel query processing. In *PODS*, pages 273–284, 2013.

[4] Bela Bollobás and Andrew Thomason. Projections of bodies and hereditary properties of hypergraphs. *Bull. London Math. Soc*, 27:417–424, 1995.

[5] Faraz Manshadi et al. A distributed algorithm for large-scale generalized matching. *PVLDB*, 6(9), 2013.

[6] Ronald Fagin, Amnon Lotem, and Moni Naor. Optimal aggregation algorithms for middleware. *J. Comput. Syst. Sci.*, 66(4):614–656, June 2003.

[7] Hung Q. Ngo, Dung T. Nguyen, Christopher Ré, and Atri Rudra. Removing the haystack to find the needle(s): Minesweeper, an adaptive join algorithm. *CoRR*, abs/1302.0914, 2013.

[8] Hung Q. Ngo, Ely Porat, Christopher Ré, and Atri Rudra. Worst-case optimal join algorithms: [extended abstract]. In *PODS*, 2012.

[9] Hung Q. Ngo, Christopher Re, and Atri Rudra. Skew strikes back: New developments in the theory of join algorithms. *SIGMOD Record*, 2014.

[10] James Renegar. Some perturbation theory for linear programming. *Math. Program.*, 65(1):73–91, May 1994.

[11] T Roughgarden. Cs369n: Beyond worst-case analysis. 2011.

[12] P. et al. Selinger. Access path selection in a relational database management system. In *SIGMOD*, pages 23–34, 1979.

[13] Srikrishna et al. Sridhar. An approximate, efficient LP solver for LP rounding. In *NIPS 26*. 2013.

[14] Todd L. Veldhuizen. Leapfrog triejoin: a worst-case optimal join algorithm. *ICDT*, 2014.