

Equality Saturation for Logical Query Optimization

James You
james.you@uwaterloo.ca
University of Waterloo
Waterloo, Ontario, Canada

Abstract

Equality saturation is a recent approach to term rewriting that avoids destructive rewriting by maintaining classes of equivalent terms. However, this approach to term rewriting comes at the cost of algorithmic time complexity. Interestingly, Equality saturation can be expressed as a query optimization problem. As such, techniques and lessons borrowed from query optimization have produced similar asymptotic improvements when applied to equality saturation. While query optimization techniques have been applied to problems in equality saturation, the reverse has been less studied. We propose the implementation and design of a novel query optimizer based on equality saturation to investigate how it may improve the state-of-the-art in query optimization.

1 Background

1.1 Term Rewriting

In compiler optimization, term rewriting is a simple but powerful technique used to transform programs. Term rewriting is a form of equational reasoning on *terms* using *rules*. Informally, terms can be considered analogous to expressions. Given a set of rules for rewriting two equivalent terms, term rewriting systems repeatedly apply these rules to terms until they cannot be rewritten further. A term that cannot be rewritten further is said to be in *normal form*. Thus, term rewriting systems should have two desirable properties:

Termination. A system terminates if there exists no infinite sequence of rewrites for any given term. This property implies that a normal form exists for a given term.

Confluence. A system is confluent if the order in which rewriting rules are applied to terms does not produce different terms. This implies that for any normal form, it is unique.

When a rewriting system has both these properties, it is *convergent*. However, most term rewriting systems are not convergent. In compiler optimization, the lack of confluence and termination properties is called the *phase ordering problem*.

1.2 Equality Saturation

The phase ordering problem remains a longstanding issue in the design of optimizing compilers[4]. Equality saturation aims to mitigate this problem through the use of the *e-graph* data structure[10]. E-graphs maintain equivalence relations between terms. Equivalent terms are said to belong to the same equivalence class in an e-graph. Equality saturation is a term rewriting system that applies rewrite rules to equivalence classes.

The process of finding all candidate equivalence classes for a rewrite rule is known as *e-matching*. Rewriting would subsequently be applied to all terms of the equivalence class. A *rebuilding* step occurs after rewriting that further simplifies equivalence classes that are themselves equivalent to one another. The final step of equality saturation is *extraction*; Extraction finds a desired term in the program using a cost-model. The nature of the cost-model presents an opportunity for physical query plan construction, as it could capture specific details of a physical machine to lower a logical query plan. However, it is out of the scope of this proposal.

2 Applications

Logical query rewriting problems rely on relational algebra equivalences. Existing query optimizers rewrite queries with manually scheduled phases[2], these approaches express a phase ordering. As a result, existing query optimizers are also constrained by phase ordering problems. The combination of these circumstances presents an opportunity for the application of equality saturation to design of query optimizers.

Classic Query Optimization as Term Rewriting. Classical query optimizers such as Volcano/Cascades[6, 7] already utilize equivalence data structures (referred to as a memo table) to find optimization opportunities. Interestingly, there is a correspondence between memo tables and e-graphs. However, e-graphs could provide an even more compact representation of equivalent queries as the rebuilding process further simplifies equivalence.

Contextual Rewriting. Query optimizations, such as [13] or [5], depend on the semantics of their sub-queries or their surrounding context. This presents a challenge for term rewriting systems as they do not typically reason about contextual information for rewriting. Contextual equality saturation[8] is an active topic of study in current academic literature.

Rewriting Recursive Queries. Egglog[14] has shown a surprising connection between datalog[3], equality saturation, and query optimization. This presents new opportunities for rewriting recursive queries as datalog contains monotonic semantics that are simpler to reason about. There is already ongoing work in current literature on using equality saturation combined with datalog semantics to optimize recursive queries[11].

2.1 Evaluation

We will evaluate our query optimizer on the widely accepted JOB[9] and JOB-complex[12] benchmarks. While it is unlikely that the proposed query optimizer would improve upon any individual query optimization. There is an opportunity for equality saturation to provide measurable performance improvements by eliminating phase ordering issues in query optimization.

References

- [1] Franz Baader and Tobias Nipkow. 1998. *Term rewriting and all that*. Cambridge University Press, USA.
- [2] Edmon Begoli, Jesús Camacho-Rodríguez, Julian Hyde, Michael J. Mior, and Daniel Lemire. 2018. Apache Calcite: A Foundational Framework for Optimized Query Processing Over Heterogeneous Data Sources. In *Proceedings of the 2018 International Conference on Management of Data* (Houston, TX, USA) (*SIGMOD '18*). Association for Computing Machinery, New York, NY, USA, 221–230. doi:10.1145/3183713.3190662
- [3] S. Ceri, G. Gottlob, and L. Tanca. 1989. What You Always Wanted to Know About Datalog (And Never Dared to Ask). *IEEE Trans. on Knowl. and Data Eng.* 1, 1 (March 1989), 146–166. doi:10.1109/69.43410
- [4] Cliff Click and Keith D. Cooper. 1995. Combining analyses, combining optimizations. *ACM Trans. Program. Lang. Syst.* 17, 2 (March 1995), 181–196. doi:10.1145/201059.201061
- [5] César Galindo-Legaria and Arnon Rosenthal. 1997. Outerjoin simplification and reordering for query optimization. *ACM Trans. Database Syst.* 22, 1 (March 1997), 43–74. doi:10.1145/244810.244812
- [6] Goetz Graefe. 1995. The Cascades Framework for Query Optimization. *IEEE Data(base) Engineering Bulletin* 18 (1995), 19–29. <https://api.semanticscholar.org/CorpusID:260706023>
- [7] G. Graefe and W.J. McKenna. 1993. The Volcano optimizer generator: extensibility and efficient search. In *Proceedings of IEEE 9th International Conference on Data Engineering*. 209–218. doi:10.1109/ICDE.1993.344061
- [8] Tyler Hou, Shadaj Laddad, and Joseph M. Hellerstein. 2025. Towards Relational Contextual Equality Saturation. arXiv:2507.11897 [cs.PL] <https://arxiv.org/abs/2507.11897>
- [9] Viktor Leis, Andrey Gubichev, Atanas Mirchev, Peter Boncz, Alfons Kemper, and Thomas Neumann. 2015. How good are query optimizers, really? *Proc. VLDB Endow.* 9, 3 (Nov. 2015), 204–215. doi:10.14778/2850583.2850594
- [10] Ross Tate, Michael Stepp, Zachary Tatlock, and Sorin Lerner. 2009. Equality saturation: a new approach to optimization. *SIGPLAN Not.* 44, 1 (Jan. 2009), 264–276. doi:10.1145/1594834.1480915
- [11] Yisu Remy Wang, Mahmoud Abo Khamis, Hung Q. Ngo, Reinhard Pichler, and Dan Suciu. 2022. Optimizing Recursive Queries with Program Synthesis. In *Proceedings of the 2022 International Conference on Management of Data* (Philadelphia, PA, USA) (*SIGMOD '22*). Association for Computing Machinery, New York, NY, USA, 79–93. doi:10.1145/3514221.3517827
- [12] Johannes Wehrstein, Timo Eckmann, Roman Heinrich, and Carsten Binnig. 2025. JOB-Complex: A Challenging Benchmark for Traditional Learned Query Optimization. arXiv:2507.07471 [cs.DB] <https://arxiv.org/abs/2507.07471>
- [13] Weipeng P. Yan and Per-Åke Larson. 1995. Eager Aggregation and Lazy Aggregation. In *Proceedings of the 21th International Conference on Very Large Data Bases (VLDB '95)*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 345–357.
- [14] Yihong Zhang, Yisu Remy Wang, Oliver Flatt, David Cao, Philip Zucker, Eli Rosenthal, Zachary Tatlock, and Max Willsey. 2023. Better Together: Unifying Datalog and Equality Saturation. *Proc. ACM Program. Lang.* 7, PLDI, Article 125 (June 2023), 25 pages. doi:10.1145/3591239