Optimizing RDF Chain Queries using Genetic Algorithms

Alexander Hogenboom, Viorel Milea, Flavius Frasincar, and Uzay Kaymak
{hogenboom, milea, frasincar, kaymak}@ese.eur.nl

Econometric Institute
Erasmus University Rotterdam
PO Box 1738, NL-3000 DR
Rotterdam, the Netherlands

In an Electronic Commerce environment, Semantic Web technologies are promising enablers for large-scale knowledge-based systems, as they facilitate machine-interpretability of data through effective data representation. Fast query engines are required for efficient real-time querying of large amounts of data, usually represented using the Resource Description Framework (RDF). An RDF model is a collection of RDF facts declared as a collection of triples, each of which consists of a subject, a predicate, and an object. These triples can be visualized using an RDF graph, which is a node and directed-arc diagram, in which each triple is represented as a node-arc-node link. RDF sources can be queried using SPARQL. The execution time of a query depends on the order in which parts of the query paths are executed. The query optimization challenge addressed here is to determine the right join order, hereby optimizing the overall response time.

In the context of the Semantic Web, two-phase optimization (2PO) has been proposed to optimize RDF query paths [3]. However, other algorithms have not yet been used for RDF query path determination, while genetic algorithms (GAs) have proven to be more effective than SA in similar problems [2]. A GA is an optimization algorithm simulating biological evolution according to the principle of survival of the fittest. A set of chromosomes, representing solutions, is exposed to evolution, consisting of selection, crossovers , and mutations. The main goal we pursue consists of investigating whether an approach based on GAs outperforms 2PO in RDF query path determination. As a first step, we focus on the performance of such algorithms when optimizing a special class of SPARQL queries, RDF chain queries (where the WHERE statement only contains a set of chained RDF node-arc-node patterns), on a single source.

We assess the performance of a GA compared to 2PO on a single source. Each algorithm is tested on chain queries varying in length from 2 to 20 predicates. Each experiment is iterated 100 times. For relatively small chain queries containing up to about 10 predicates, 2PO turns out to require the least time for query optimization. For bigger chain queries, a GA converges faster to the solution. Furthermore, a GA tends to find better solutions of more consistent quality than 2PO does, especially for larger queries. When a time limit of 1 second is set (allowing the algorithms to perform at least a couple of iterations while assuming this to be an acceptable maximum waiting time in a real-time environment), a GA tends to generate solutions of even better quality compared to 2PO. The consistency in solution quality of RCQ-GA, as opposed to 2PO, is not clearly affected by a time limit.

In his talk, Alexander Hogenboom (PhD student at Erasmus University Rotterdam) will present these results, as further detailed in [1]. He will show that in optimizing query paths for chain queries in a single-source RDF query execution environment, the performance of a GA compared to 2PO is positively correlated with solution space complexity and environmental restrictiveness (a time limit). The proposed GA outperforms 2PO in solution quality, execution time needed, and consistency of solution quality.

References

[1] Alexander Hogenboom, Viorel Milea, Flavius Frasincar, and Uzay Kaymak. RCQ-GA: RDF Chain Query Optimization using Genetic Algorithms. In *Tenth International Conference on E-Commerce and Web Technologies (EC-Web 2009)*, pages 181–192, 2009.

[2] Michael Steinbrunn, Guido Moerkotte, and Alfons Kemper. Heuristic and Randomized Optimization for the Join Ordering Problem. *The VLDB Journal*, 6(3):191–208, 1997.

[3] Heiner Stuckenschmidt, Richard Vdovjak, Jeen Broekstra, and Geert Jan Houben. Towards Distributed Processing of RDF Path Queries. *International Journal of Web Engineering and Technology*, 2(2-3):207–230, 2005.