Errata for "Analysis of two existing and one new dynamic programming algorithm for the generation of optimal bushy join trees without cross products"

Andreas Meister
University Magdeburg
Germany
andreas.meister@ovgu.de

Guido Moerkotte
University Mannheim
Germany
moerkotte@uni-mannheim.de

Gunter Saake
University Magdeburg
Germany
gunter.saake@ovgu.de

Algorithm 1: EnumerateCmp [1]

Precondition: `nodes in` V `are numbered according to an arbitrary enumeration`
Input : a connected query graph $G = (V, E)$, a connected subset S_1
Output : emits all complements S_2 for S_1 such that (S_1, S_2) is a csg-cmp-pair

1 $X = \mathcal{B}_{min(S_1)} \cup S_1$;
2 $N = \mathcal{N}(S_1) \setminus X$;
3 **foreach** $v_i \in N$ *by descending* i **do**
4 | **emit** $\{v_i\}$;
5 | EnumerateCsgRec(G, $\{v_i\}$, $X \cup N \rightarrow X \cup \mathcal{B}_i(\mathbf{N})$);

Algorithm 2: EnumerateCsgRec [1]

Input : a connected query graph $G = (V, E)$, a connected subset S, a set of excluded nodes X
Output: emits connected subsets

1 $N = \mathcal{N}(S) \setminus X$;
2 **foreach** $S' \subseteq N$, $S' \neq \emptyset$ **do**
3 | **emit**$(S \cup S')$;
4 **foreach** $S' \subseteq N$, $S' \neq \emptyset$ **do**
5 | EnumerateCsgRec(G, $(S \cup S')$, $(X \cup N)$);

1. PROBLEM

In the published version of `EnumerateCmp` in the Section 3.3 on Page 936 [1], see also Algorithm 1, a small error is included in Line 5. In the first call of EnumerateCsgRec, too many nodes $(X \cup N)$ will be excluded for the emission of complements, leading to the fact that, in general, not all complements will be emitted correctly.

We will discuss the error based on the emission of complements for the connected subset $S_1 = \{v_0\}$ for the query graph $G = (V, E)$ shown in Figure 1. The query graph consists of tables $V = \{v_0; v_1; v_2; v_3; v_4\}$ and joins E. The nodes are ranked based on an arbitrary enumeration.

EnumerateCmp emits all complements (cmp) S_2 for a connected subgraph (csg) S_1 to create all required csg-cmp-pairs. Since both csg and cmp are connected, disjunct and joinable with the corresponding element of a csg-cmp-pair, csg-cmp-pairs represent valid and needed calculations for the dynamic programming approach.

To emit complements, EnumerateCmp defines a set of excluded nodes $X(\{v_0\})$ given the connected subgraph S_1 $(\{v_0\})$, see Line 1. Nodes in X will not be considered during the emission of complements to avoid duplicates and correctness. X includes all nodes of G with a rank smaller than the smallest rank within S_1 $(\mathcal{B}_{min(S_1)} = \emptyset)$, as well as all nodes of S_1 $(\{v_0\})$. Afterwards, relevant nodes N $(\{v_1; v_2; v_3\})$ for the creation of complements will be determined by excluding X from the neighboring nodes of S_1 $(\mathcal{N}(S_1) = \{v_1; v_2; v_3\})$, see Line 2. Then, all complements based on the relevant nodes will be emitted, see Line 3-5. First, EnumerateCmp emits single node complements ($\{v_1\}$, $\{v_2\}$, $\{v_3\}$), see Line 4. Second, EnumerateCsgRec, see Algorithm 2, will recursively emit additional complements based on the single node complement $(\{v_i\})$ and excluded nodes $(X \cup N = \{v_0; v_1; v_2; v_3\})$, see Line 5.

EnumerateCsgRec starts by determining relevant nodes N for the creation of complements by excluding X from the neighboring nodes of the considered complement S, see Line 1. Afterwards, new complements will be created by combining the complement S with all subsets $S' \subseteq N$, see Line 2 and Line 3. Then, the next recursion step is executed by expanding the complement $S \rightarrow (S \cup S')$ and excluded set $X \rightarrow (X \cup N)$ for each subset $S' \subseteq N$, see Line 4 and Line 5.

Considering $S = \{v_1\}$ and $N = \{v_0; v_4\} \setminus \{v_0; v_1; v_2; v_3\} = \{v_4\}$, the complement $\{v_1; v_4\}$ will be emitted first. The next call of EnumerateCsgRec considering the expanded set $\{v_1; v_4\}$ should emit $\{v_1; v_2; v_4\}$ and $\{v_1; v_3; v_4\}$. As both nodes v_2 and v_3 are already included in the excluded nodes, these complements cannot be emitted. Consequently, also the complement $\{v_1; v_2; v_3; v_4\}$ cannot be emitted. Regarding $S = \{v_2\}$, we will see a similar behavior. First, the complement $\{v_2; v_4\}$ will be emitted. The next call of Enu-

Proceedings of the VLDB Endowment, Vol. 11, No. 10

DOI: https://doi.org/10.14778/3231751.3231756

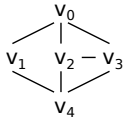

Figure 1: Sample graph [1].

merateCsgRec considering the expanded set $\{v_2; v_4\}$ should emit the complement $\{v_2; v_3; v_4\}$. As the node v_3 is already included in the excluded nodes, $\{v_2; v_3; v_4\}$ cannot be emitted. Only for $S = \{v_3\}$, all complements will be emitted correctly, as $\{v_3; v_4\}$ is the only required complement.

We want to highlight that the concept of EnumerateCmp and EnumerateCsgRec as well as the corresponding proofs are still valid [1]. The discussed error is only based on an erroneous node set $X \cup N$ in Line 5 of Algorithm 1. Therefore, the error also has no impact on acyclic graphs, such as linear or star queries. Considering two arbitrary nodes n_1 and n_2 in set N of a connected subgraph S_1, both nodes can safely be included into the set X of EnumerateCsgRec in case of an acyclic graph based on the following two reasons. First, a path exists from n_1 to n_2 through S_1. Otherwise, either n_1 or n_2 would not be included in N. Second, no other path from n_1 to n_2 exists. Hence, n_2 will not appear in any of the evaluated sets $\mathcal{N}(S)$ based on n_1, see Line 1 of Algorithm 2. As n_2 will not be an element of $\mathcal{N}(S)$, the result of $\mathcal{N}(S) \setminus X$ will be the same, whether n_2 is part of X or not. If n_2 would appear in $\mathcal{N}(S)$ of a complement based on n_1, a second path from n_1 to n_2 besides through S_1 would exists, as all nodes of S_1 are excluded in the first call. This would violate the property of an acyclic graph.

In cyclic query graphs, such as cyclic or clique queries, two or more paths can exits between the nodes n_1 and n_2. Therefore, the existence of n_2 in X influence the set of emitted complements. The more paths between n_1 and n_2 exists, the more complements will be lost based on the error. Hence, the number of 'lost' complements also depends on the number of cycles within a query graph. Therefore, cyclic graphs will suffer less compared to clique queries. However, as the emission of subgraphs is correct and at least the single-node complements will also be emitted correctly, even with the error, a valid optimization result will be provided. Nevertheless, as not all required csg-cmp-pairs are evaluated, an optimal optimization result cannot be guaranteed. The impact on the quality or efficiency of the optimization result depends on the use case, e.g. query graph, selectivities, table sizes, etc., and can vary from no to a significant impact.

2. SOLUTION

To solve the discussed problem, we need to adapt the set of excluded nodes within Line 5 of Algorithm 1. We need to include only neighboring nodes with a smaller rank than v_i into the set of exclude nodes, besides X. We can use $\mathcal{B}_i(W) = \{v_j | v_j \in W, j \leq i\}$ defined in the paper [1]. Instead of using $X \cup N$, we need to use $X \cup \mathcal{B}_i(N)$ to enumerate all complements correctly. Coming back to the example of the enumeration of the sample graph. With the adapted variant of EnumerateCmp, the set of excluded nodes are adapted based on the considered single node complements: $v_1 : X \cup \mathcal{B}_1(\mathbf{N}) = \{v_0\}$; $v_2 : X \cup \mathcal{B}_2(\mathbf{N}) = \{v_0; v_1\}$; $v_3 : X \cup \mathcal{B}_3(\mathbf{N}) = \{v_0; v_1; v_2\}$. Hence, the evaluation of EnumerateCsgRec based on $S = \{v_1\}$ will also emit the missing complements $\{v_1; v_2; v_4\}$, $\{v_1; v_3; v_4\}$, $\{v_1; v_2; v_3; v_4\}$, because v_2 and v_3 will not be in the initial set of excluded nodes. Both nodes will only be included in the set of excluded nodes after the nodes were included in emitted complements. Similar, for $S = \{v_2\}$, the complement $\{v_2; v_3\}$ will be emitted correctly, because v_3 will not be included in the initial set of excluded nodes.

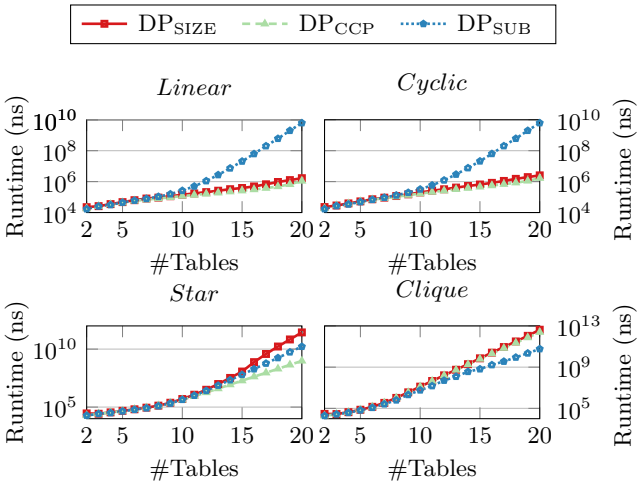

Figure 2: Optimization time for different query topologies

Given the correction, the number of evaluated csg-cmp-pairs follows the required number of calculations determined by Ono and Lohman [2] for both cyclic and acyclic query graphs. Based on the provided proofs [1], all emitted csg-cmp-pairs are unique and valid. Therefore, a correct enumeration of calculations is performed.

3. EVALUATION

We also performed an evaluation with our own implementation for DP_{SIZE}, DP_{SUB}, and DP_{CCP} [1]. DP_{SIZE} enumerates join pairs based on the number of contained tables. DP_{SUB} enumerates join pairs based on the set of contained tables. DP_{CCP} enumerates join pairs based on the query graph. We used similar use cases to the original paper: four different query topologies (linear, cyclic, star, and clique) with a maximal table number of 20. We performed the optimization on 30 random generated queries and aggregated the runtimes using the average. We show the result in Figure 2. We achieve similar results to the published results. For linear, cyclic and star queries, DP_{CCP} is superior to DP_{SIZE} and DP_{SUB}. In contrast to the other topologies, in clique queries all subsets of nodes are connected. Hence, DP_{SUB} only evaluates required join pairs similar to DP_{CCP}. Nevertheless, DP_{SUB} achieves better results compared to DP_{CCP} based on the reduced overhead of the used enumeration.

4. REFERENCES

[1] G. Moerkotte and T. Neumann. Analysis of Two Existing and One New Dynamic Programming Algorithm for the Generation of Optimal Bushy Join Trees Without Cross Products. VLDB, pages 930–941. VLDB End., 2006.

[2] K. Ono and G. M. Lohman. Measuring the Complexity of Join Enumeration in Query Optimization. VLDB, pages 314–325. Morgan Kaufmann, 1990.