Parallel-Correctness and Transferability for Conjunctive Queries

Tom J Ameloot[1], Gaetano Geck[2], Bas Ketsman[1], Frank Neven[1], Thomas Schwentick[2]

[1] Hasselt University & transnational University of Limburg [2] TU Dortmund University

Introduction

Big Data: Data too large to be processed on one server

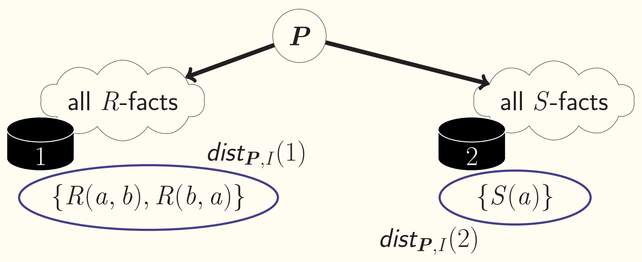
Today systems: Hadoop, Spark, ..., and many others

Common Strategy:

- Data is stored in a distributed way
- Query evaluation:
  Multiple rounds with reshuffling

Distribution Policies

Network $\mathcal{N}$ is a finite set of nodes
Instance $I = \{R(a,b), R(b,a), S(a)\}$



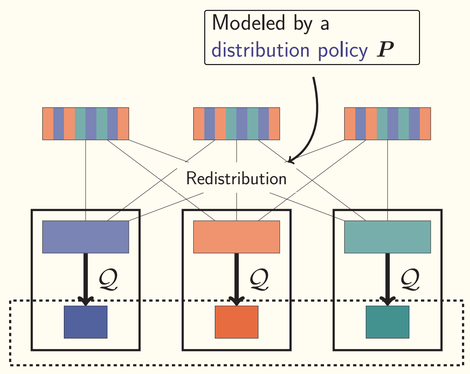
= distribution of I based on $\boldsymbol{P}$

Definition
A distribution policy $\boldsymbol{P}$ is a total function mapping facts (over **dom**) to sets of nodes in $\mathcal{N}$

Simple Evaluation Algorithm

1-Round MPC model [Koutris & Suciu 2011]

Input = query $\mathcal{Q}$



Step 1:

Step 2:

Output = union of output at each server

When is the simple algorithm correct on a distribution policy?

Parallel-Correctness

Definition
$\mathcal{Q}$ is parallel-correct on I w.r.t. $\boldsymbol{P}$, iff

$$\mathcal{Q}(I) = \bigcup_{\kappa \in \mathcal{N}} \mathcal{Q}(dist_{\boldsymbol{P},I}(\kappa))$$

$\supseteq$ by monotonicity

Definition (w.r.t. all instances)
$\mathcal{Q}$ is parallel-correct w.r.t. $\boldsymbol{P}$ iff
$\mathcal{Q}$ is parallel-correct w.r.t. $\boldsymbol{P}$ on every I

Sufficient Condition

(C0) for every valuation V for $\mathcal{Q}$,

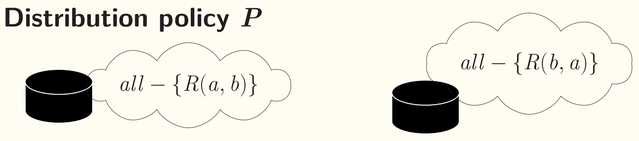
$$\bigcap_{\boldsymbol{f} \in V(\text{body}_\mathcal{Q})} \boldsymbol{P}(\boldsymbol{f}) \neq \emptyset.$$

Intuition: Facts required by a valuation meet at some node

Lemma
(C0) implies $\mathcal{Q}$ parallel-correct w.r.t. $\boldsymbol{P}$.

Not necessary

Distribution policy $\boldsymbol{P}$



Query $\mathcal{Q}$: $T(x,z) \leftarrow R(x,y), R(y,z), R(x,x)$

$V = \{x, z \to a, y \to b\}$ $\quad$ $V' = \{x, y, z \to a\}$

Requires: $\quad$ Requires:

$R(a,b)$ $R(b,a)$ $R(a,a)$ $\supsetneq$ $R(a,a)$

Derives: $\quad$ Derives:

$T(a,a)$ $=$ $T(a,a)$

Do not meet

Notice: $\mathcal{Q}$ is minimal CQ

- CQ is minimal iff injective valuations are minimal

Proposition
Testing whether a valuation is minimal is coNP-complete.

Characterization

Lemma
$\mathcal{Q}$ is parallel-correct w.r.t. $\boldsymbol{P}$ iff
(C1) for every minimal valuation V for $\mathcal{Q}$,

$$\bigcap_{\boldsymbol{f} \in V(\text{body}_\mathcal{Q})} \boldsymbol{P}(\boldsymbol{f}) \neq \emptyset.$$

Definition
V is minimal if no V' exists, where
$V'(\text{head}_\mathcal{Q}) = V(\text{head}_\mathcal{Q})$, $V'(\text{body}_\mathcal{Q}) \subsetneq V(\text{body}_\mathcal{Q})$.

Complexity

Theorem
Deciding whether $\mathcal{Q}$ is parallel-correct w.r.t. $\boldsymbol{P}$ is Π_2^P-complete.

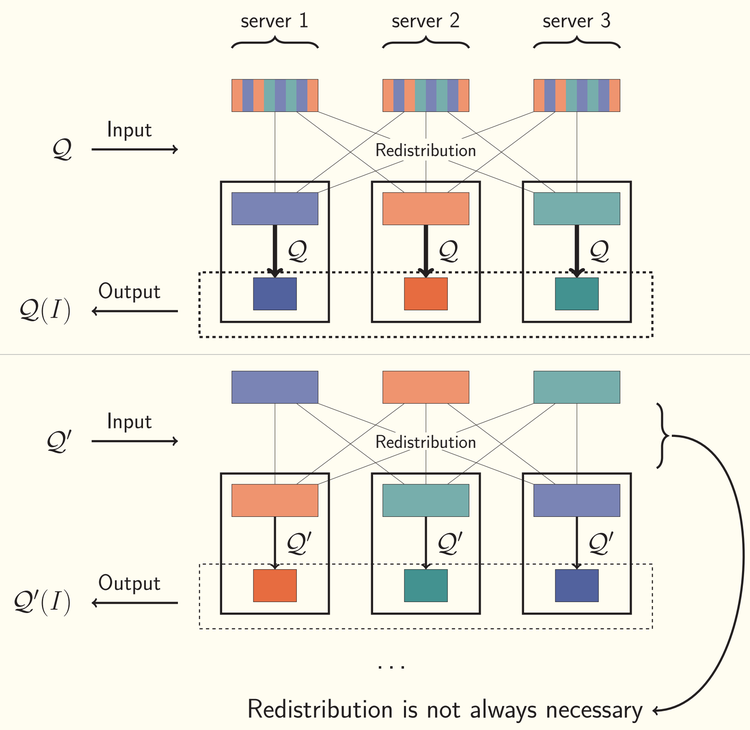
Proof:

- Lower bound: Reduction from Π_2-QBF
- Upper bound: Characterization

but, requires proper formalization of $\boldsymbol{P}$

Multi-Query Optimization

Computing a set of queries: $\{\mathcal{Q}, \mathcal{Q}', \ldots\}$



Which queries allow to reuse the distribution obtained for another query?

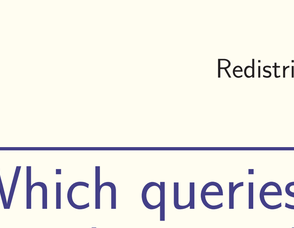
Transferability

Definition
$\mathcal{Q} \to_T \mathcal{Q}'$ iff $\mathcal{Q}'$ is parallel-correct on every $\boldsymbol{P}$
where $\mathcal{Q}$ is parallel-correct on

Example

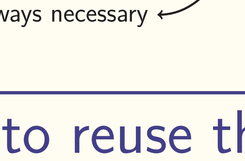
$\mathcal{Q}: T() \leftarrow R(x,y), R(y,z), R(z,w)$

Minimal valuations for $\mathcal{Q}$:



$\mathcal{Q}': N() \leftarrow R(x,y), R(y,x)$

Minimal valuations for $\mathcal{Q}'$:



Facts required by minimal valuations for $\mathcal{Q}'$ are also required by minimal valuations for $\mathcal{Q}$

$\boxed{\mathcal{Q} \to_T \mathcal{Q}'}$

Characterization

Lemma
$\mathcal{Q} \to_T \mathcal{Q}'$ iff
(C2) for every minimal valuation V' for $\mathcal{Q}'$ there is a minimal valuation V for $\mathcal{Q}$, s.t. $V'(\text{body}_\mathcal{Q}) \subseteq V(\text{body}_\mathcal{Q})$.

Based on query structure alone, not on distribution policies

Complexity

Theorem
Deciding $\mathcal{Q} \to_T \mathcal{Q}'$ is Π_3^P-complete.

Proof:

- Lower bound: Reduction from Π_3-QBF
- Upper bound: Characterization

Strongly Minimal CQs

Definition
A CQ is strongly minimal if all its valuations are minimal

- Full-CQs
  $T(x,y) \leftarrow R(x,y), R(x,x)$
- CQs without self-joins
  $T() \leftarrow R(x,y), S(x,x)$
- Hybrids
  $T(y) \leftarrow R(x,y), R(x,x), R(z,x), S(z)$

A minimal CQ is not always strongly minimal

Lemma
Deciding whether $\mathcal{Q}$ is strongly minimal is coNP-complete

Theorem
Deciding $\mathcal{Q} \to_T \mathcal{Q}'$ is NP-complete for strongly minimal $\mathcal{Q}$

Hypercube

- Invented in the context of Datalog evaluation
  [Ganguli, Silberschatz & Tsur 1990]
- Described in Map-Reduce context
  [Afrati & Ullman 2010]
- Intensively studied by many people
  [Beame, Koutris & Suciu 2014]

Algorithm sketch:

- Reshuffling based on structure of $\mathcal{Q}$

Partitioning of complete valuations over servers in instance independent way through hashing of domain values

Let $\mathcal{H}(\mathcal{Q})$ be the family of Hypercube distributions for $\mathcal{Q}$.

Definition
$\mathcal{Q} \to_H \mathcal{Q}'$ iff
$\mathcal{Q}'$ is parallel-correct w.r.t. every $\boldsymbol{P} \in \mathcal{H}(\mathcal{Q})$.

Two properties:

- $\mathcal{Q}$-generous: for every valuation facts meet on some node $(\forall \boldsymbol{P} \in \mathcal{H}(\mathcal{Q}))$
- $\mathcal{Q}$-scattered: there is a policy scattering facts in such a way that no facts meet by coincidence $(\forall I)$

Theorem
Deciding whether $\mathcal{Q} \to_H \mathcal{Q}'$ is NP-complete
(also when $\mathcal{Q}$ or $\mathcal{Q}'$ is acyclic)

Conclusion & Future Work

Summary:
Formal framework for reasoning about correctness of query evaluation and optimization in a distributed setting

Based on two concepts:

- Parallel-correctness
- Transferability

Independent of expression mechanism

Related Concepts

Containment
$\mathcal{Q} \subseteq \mathcal{Q}'$

Lemma
Containment and transferability are incomparable

Determinacy (Data-Integration)
$\mathcal{Q}'(I) = \mathcal{Q}'(J)$ implies $\mathcal{Q}(I) = \mathcal{Q}(J)$, for every I, J

Lemma
Determinacy and transferability are incomparable

Transferability (Distributed)
$\mathcal{Q} \to_T \mathcal{Q}'$

Future Work

Expression Formalism for distribution policies

- Other than Hypercube?

Distribution policy for set of queries

- Given CQ: which distribution policy? Hypercube
- Given set of CQs: which distribution policy? Open question

Tractable Results

- Other classes of queries?
- Other families of distribution policies?

More expressive classes of queries

- This work: CQs
- FO: undecidable
- initial results: UCQs, CQs with negation

[1] <firstname>.<lastname>@uhasselt.be – Hasselt University, Martelarenlaan 42, 3590 Hasselt (Belgium)

[2] <firstname>.<lastname>@udo.edu – TU Dortmund, Fakultät für Informatik, Otto-Hahn-Str. 12, 44227 Dortmund (Germany)