Single-Round Multi-Join Evaluation

Bas Ketsman

Outline

1. Introduction

2. Parallel-Correctness

3. Transferability

4. Special Cases

Motivation

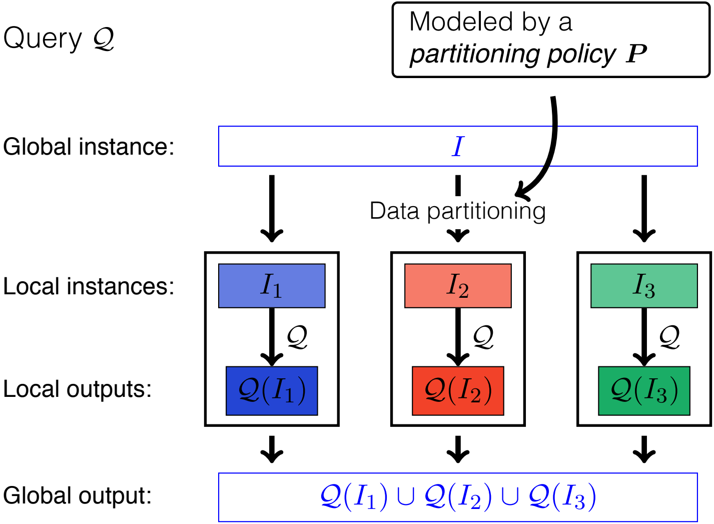
Single-round Multi-joins

- Less rounds / barriers

Formal framework for reasoning about distributed query evaluation and optimization

Building Block

1-Round MPC model [Koutris & Suciu 2011]



Main Questions: Question 1

Given target query and a distribution policy:

Does the simple algorithm work?

Parallel-Correctness

"Is query parallel-correct for current distribution policy?"

- If yes:
  No data reshuffling needed!
- If no:
  Choose one that works and reshuffle.

  future work: Which one is cheapest to obtain?

Main Questions: Question 2

It may be unpractical to reason about distribution policies

- Sometimes complex to reason about
- May be hidden behind abstraction layer
- May not have been chosen yet

Given target query and previously computed query:

Do we need to reshuffle?

Parallel-Correctness $\boxed{\text{Transferability}}$

"Given $\mathcal{Q}_1, \mathcal{Q}_2$: in which order to compute?"

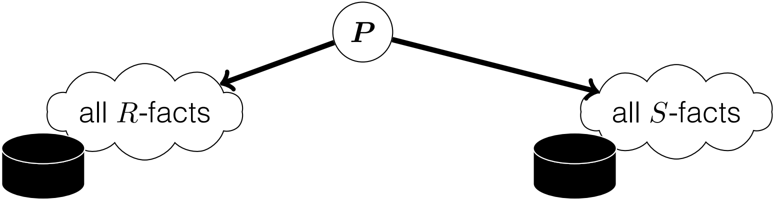
► **If transferability from $\mathcal{Q}_1$ to $\mathcal{Q}_2$:**
Compute $\mathcal{Q}_1$ first, then $\mathcal{Q}_2$ for free!

Outline

1. Introduction
2. Parallel-Correctness
3. Transferability
4. Special Cases

Distribution Policies

Network $\mathcal{N}$ is a finite set of machines [Zinn et all. 2013]



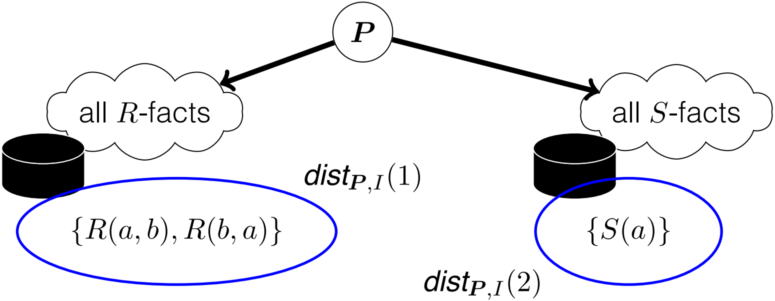
Definition

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

- Based on granularity of facts
- No context
- Obtainable in distributed fashion

Distribution Policies

Network $\mathcal{N}$ is a finite set of machines [Zinn et all. 2013]



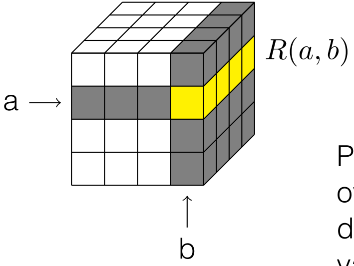
= *distribution of* I *based on* $\boldsymbol{P}$

Instance $I = \{R(a,b), R(b,a), S(a)\}$

Example Policy: Hypercube

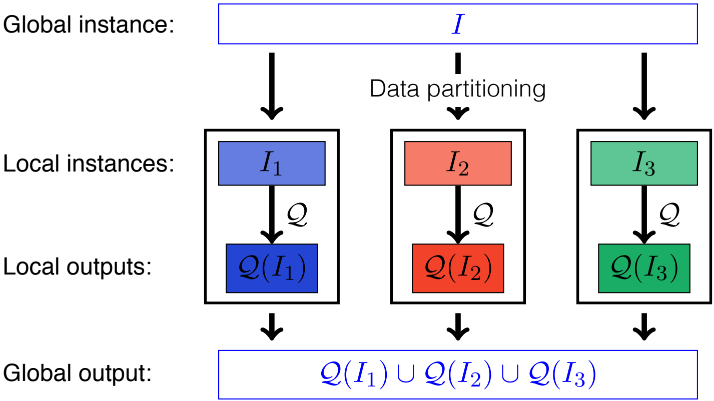
[Afrati & Ullman 2010, Beame, Koutris & Suciu 2014]

$$(x, y, z) \leftarrow R(x, y), S(y, z), T(z, x)$$



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

Simple Evaluation Algorithm



Notation

$$[\mathcal{Q}, \boldsymbol{P}](I) = \bigcup_{\kappa \in \mathcal{N}} \mathcal{Q}(\mathit{dist}_{\boldsymbol{P},I}(\kappa))$$

Parallel-Correctness

Definition

$\mathcal{Q}$ is ***parallel-correct on*** I ***w.r.t.*** $\boldsymbol{P}$, iff $[\mathcal{Q}, \boldsymbol{P}](I) = \mathcal{Q}(I)$

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

Conjunctive Queries

Conjunctive Query:
Existentially quantified conjunction of relational atoms

$$\underbrace{T(\bar{x})}_{\text{head}_{\mathcal{Q}}} \leftarrow \underbrace{R_1(\bar{y}_1), \ldots, R_m(\bar{y}_m)}_{\text{body}_{\mathcal{Q}}}$$

Valuations: V = mapping from variables to domain elements

If $V(\text{body}_{\mathcal{Q}}) \subseteq I$ then output $V(\text{head}_{\mathcal{Q}})$.

CQs are monotone ($\mathcal{Q}(I) \subseteq \mathcal{Q}(I \cup J) \forall I, J$):

- CQs are parallel-sound on every $\boldsymbol{P}$
- parallel-correct iff parallel-complete

$$[\mathcal{Q}, \boldsymbol{P}](I) = \mathcal{Q}(I), \forall I \text{ iff } \mathcal{Q}(I) \subseteq [\mathcal{Q}, \boldsymbol{P}](I), \forall I$$

Parallel-Correctness
Sufficient Condition

(PC0) 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 machine

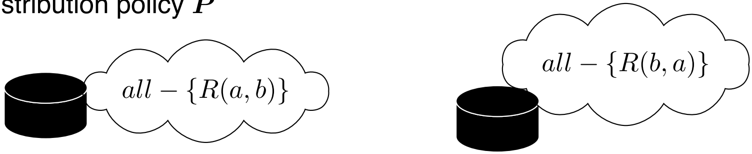
Lemma

(PC0) implies $\mathcal{Q}$ parallel-correct w.r.t. $\boldsymbol{P}$.

Not necessary

(PC0) not Necessary
Example

Distribution policy $\boldsymbol{P}$



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

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

Requires:

$\boxed{R(a,b) \quad R(b,a) \quad R(a,a)} \supsetneq \boxed{R(a,a)}$

Derives: Do not meet

$\boxed{T(a,a)} = \boxed{T(a,a)}$

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

Requires:

$\boxed{R(a,a)}$

Derives:

$\boxed{T(a,a)}$

Lemma

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

(PC1) 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
$V'(\text{head}_{\mathcal{Q}}) = V(\text{head}_{\mathcal{Q}})$, $V'(\text{body}_{\mathcal{Q}}) \subsetneq V(\text{body}_{\mathcal{Q}})$.

Parallel-Correctness

Example

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

$V = \{x, z \to a, y \to b\}$		$V' = \{x, y, z \to a\}$
Requires:		Requires:
$\boxed{R(a, b) \quad R(b, a) \quad R(a, a)}$	$\supsetneq$	$\boxed{R(a, a)}$
Derives:		Derives:
$\boxed{T(a, a)}$	$=$	$\boxed{T(a, a)}$

Minimal

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

CQ is minimal iff injective valuations are minimal

Proposition

Testing whether a valuation is minimal is coNP-complete.

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: (PC1)

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

Parallel-Correctness: Complexity

	CQ	$\cdots$	CQ$\{\neq, \cup\}$
$\mathcal{P}_{\text{fin}}$	Π_2^p-c		Π_2^p-c
$\mathcal{P}_{\text{enum}}$	Π_2^p-c		Π_2^p-c
$\mathcal{P}^k_{\text{nondet}}$	Π_2^p-c		Π_2^p-c

Robust under adding inequalities and union

Inequalities:

$$T(\bar{x}) \leftarrow R_1(\bar{y}_1), \ldots, R_m(\bar{y}_m), x \neq y, y \neq z$$

Union:

$$\mathcal{Q} = \{\mathcal{Q}_1, \ldots, \mathcal{Q}_k\},$$

with $\text{head}_{\mathcal{Q}_1}, \ldots, \text{head}_{\mathcal{Q}_k}$ over same relation.

Safe Negation

$$\underbrace{T(\bar{x})}_{\text{head}_{\mathcal{Q}}} \leftarrow \underbrace{R_1(\bar{y}_1), \ldots, R_m(\bar{y}_m)}_{\text{pos}_{\mathcal{Q}}}, \underbrace{\neg S_1(\bar{z}_1), \ldots, \neg S_k(\bar{z}_k)}_{\text{neg}_{\mathcal{Q}}}$$

with $\text{vars}(\text{neg}_{\mathcal{Q}}) \subseteq \text{vars}(\text{pos}_{\mathcal{Q}})$.

In general:

	$\{\neg\}$	$\cdots$	$\{\neg, \cup, \neq\}$
$\mathcal{P}_{\text{enum}}$	coNEXP-c		coNEXP-c
$\mathcal{P}^k_{\text{nondet}}$	coNEXP-c		coNEXP-c

Surprisingly we found this via $\text{CQ}^{\neg}$ containment!!

Containment

We thought Π_p^2 completeness of CQ$^\neg$ containment was folklore

Theorem

In general, containment for CQ$^\neg$ is coNEXPTIME-complete

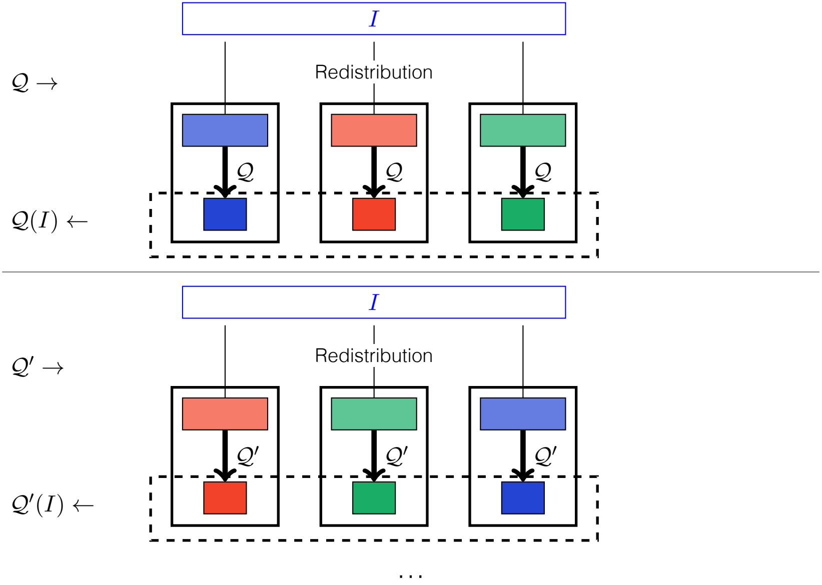
Proof:

- Lower bound: succinct 3-colorability
- Upper bound: guess instances over bounded domain

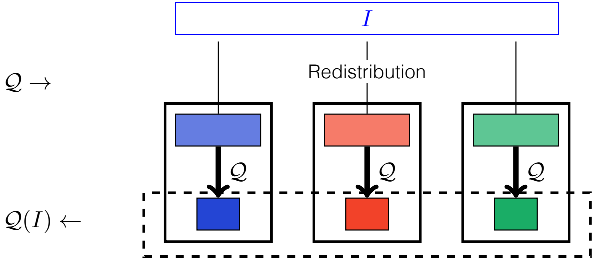
Outline

1. Introduction
2. Parallel-Correctness
3. Transferability
4. Special Cases

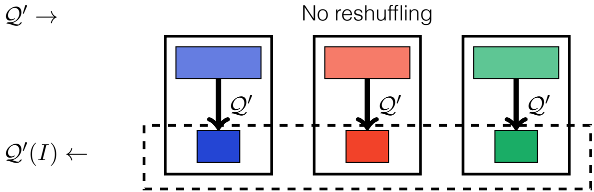
Computing Multiple Queries



Computing Multiple Queries



When can $\mathcal{Q}'$ be evaluated on data partitioning used for $\mathcal{Q}$?



...

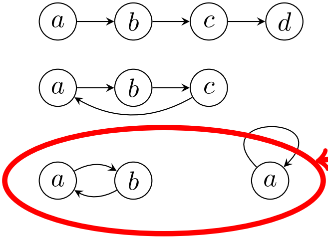
Transferability

Definition

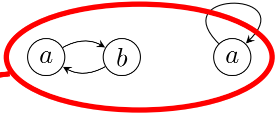
$\mathcal{Q} \rightarrow_T \mathcal{Q}'$ iff $\mathcal{Q}'$ is parallel-correct on every $\boldsymbol{P}$ where $\mathcal{Q}$ is parallel-correct on

Example

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



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



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

Lemma

$\mathcal{Q} \rightarrow_T \mathcal{Q}'$ iff

(C2) for every minimal valuation V' for $\mathcal{Q}'$ there is a minimal valuation V for $\mathcal{Q}$, s.t.

$$V'(\text{body}_{\mathcal{Q}'}) \subseteq V(\text{body}_{\mathcal{Q}}).$$

Transferability
Characterization & Complexity

Lemma

$\mathcal{Q} \rightarrow_T \mathcal{Q}'$ iff

(C2) for every minimal valuation V' for $\mathcal{Q}'$ there is a minimal valuation V for $\mathcal{Q}$, s.t.

$$V'(\text{body}_{\mathcal{Q}'}) \subseteq V(\text{body}_{\mathcal{Q}}).$$

Theorem

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

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

Based on query structure alone, not on distribution policies

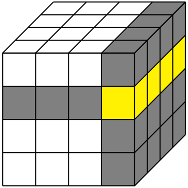
Outline

1. Introduction
2. Parallel-Correctness
3. Transferability
4. Special Cases

Hypercube

Algorithm:

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



$\mathcal{H}(\mathcal{Q})$ = family of Hypercube policies for $\mathcal{Q}$.

Definition

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

Hypercube

Two properties:

- **$\mathcal{Q}$-generous:** for every valuation facts meet on some machine ($\forall \boldsymbol{P} \in \mathcal{H}(\mathcal{Q})$)
- **$\mathcal{Q}$-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)

Tractable results future work

- Queries classes
- Concrete families of distribution policies

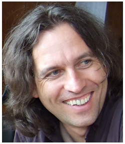
(some other special cases in [AGKNS 2011])

Hybrid techinques / Tradeoffs future work

- Single-round Multi-join vs multi-rounds?
- Combining queries vs sequential distributed evaluation?

Joint work with

Tom Ameloot, Gaetano Geck, Frank Neven and Thomas Schwentick

- Parallel-Correctness and Transferability for Conjunctive Queries, PODS 2015.
- Technical report: http://arxiv.org/abs/1412.4030
- Parallel-Correctness and Containment for Conjunctive Queries with Union and Negation, ICDT 2016.
- Data partitioning for single-round multi-join evaluation in massively parallel systems, Sigmod Record 2016 (not yet published).