Query Optimization
Exercise Session 4

Andrey Gubichev

May 12, 2014

Homework: Selectivity estimations

The selectivity of $\sigma_{R1.x=c}$ is...

- if x is the key: $\frac{1}{|R1|}$
- if x is not the key: $\frac{1}{|R1.x|}$

The selectivity of $\bowtie_{R1.x=R2.y}$ is...

- if both x and y are the keys: $\frac{1}{max(|R1|,|R2|)}$
- if only x is the key: $\frac{1}{|R1|}$
- if both x and y are not the keys: $\frac{1}{max(|R1.x|,|R2.y|)}$

When our selectivity estimations are bad?

Homework: Selectivity estimations

Setup: due Florian Walch
Description: Consider online shop selling women's clothing
Schema: Customers, Countries, Orders (FK: Customers, Countries)

100 customers (99 female), 50 countries, 1000 orders
Query 1: find all the orders and countries of male customers

Homework: Selectivity estimations

Estimations	**Reality**
σ Male customers: 50 tuples	Male customers: 1 tuple
Sel.($\sigma \bowtie$Orders): 0.02	Join with Orders: ca.10 tuples
Sel.(Orders$\bowtie$Countries):	Join with Countries: ca.10 tuples

Another example

Tables: Person (ID, Name, City), Friends(FK: Person, Person)
Query: Find all people called John from NYC who are friends with a person called Mary from Beijing.

Homework

When is a bushy tree with a crossproduct optimal?

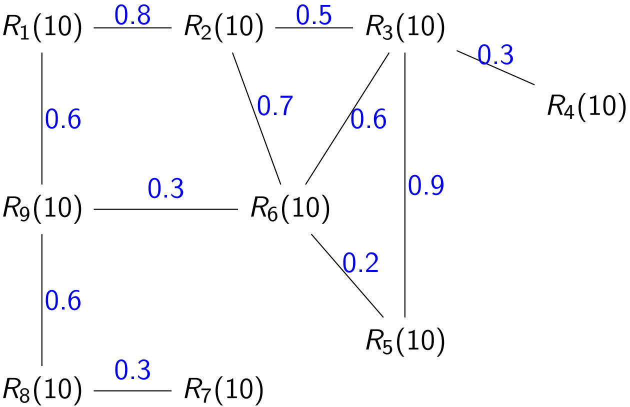
- When is a bushy tree optimal?
- When use of a crossproduct is beneficial?

Greedy operator ordering

- take the query graph
- find relations R_1, R_2 such that $|R_1 \bowtie R_2|$ is minimal and merge them into one node
- repeat until the query graph has more than one node

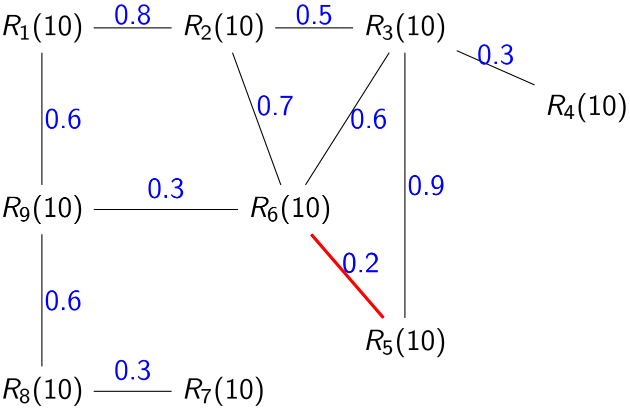
Generates bushy trees!

Example

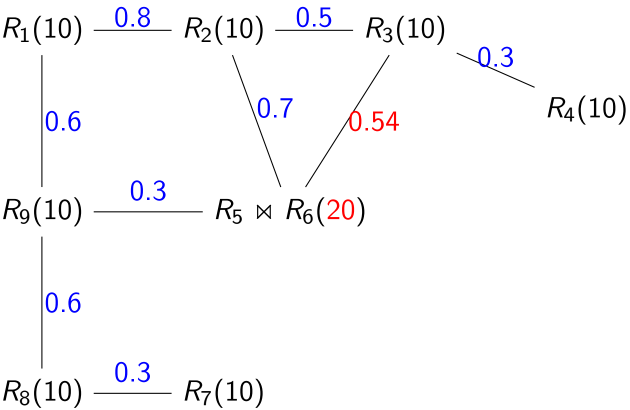


$R_1(10)$
0.8
$R_2(10)$
0.5
$R_3(10)$
0.3
$R_4(10)$
0.6
0.7
0.6
0.9
$R_9(10)$
0.3
$R_6(10)$
0.2
0.6
$R_5(10)$
$R_8(10)$
0.3
$R_7(10)$

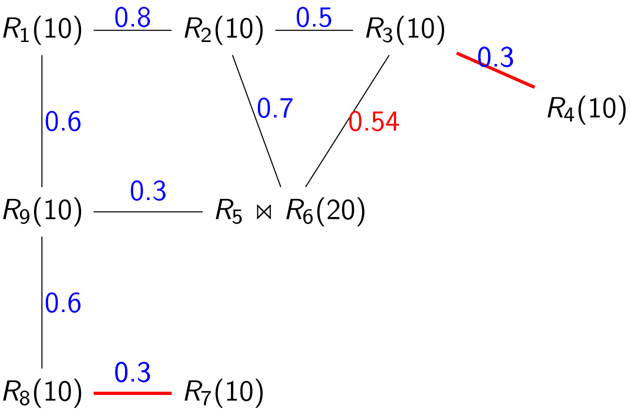
Example - step 1



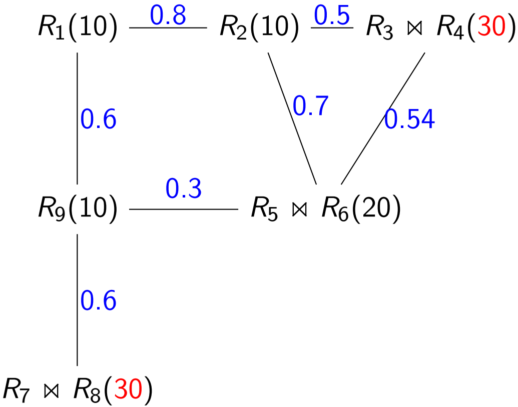
Example - after step 1



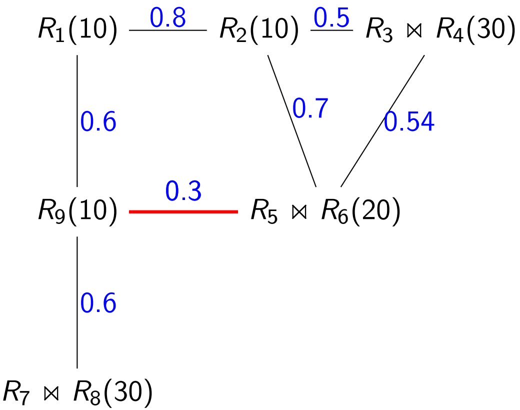
Example - step 2



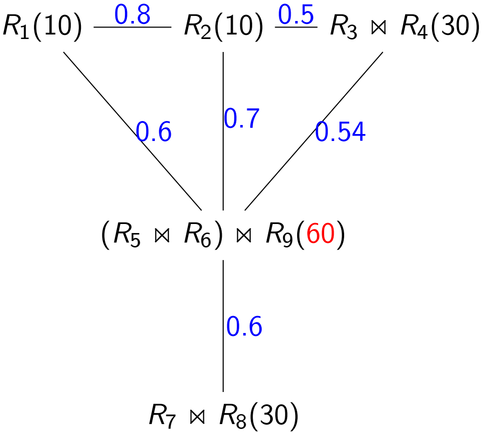
Example- after step 2



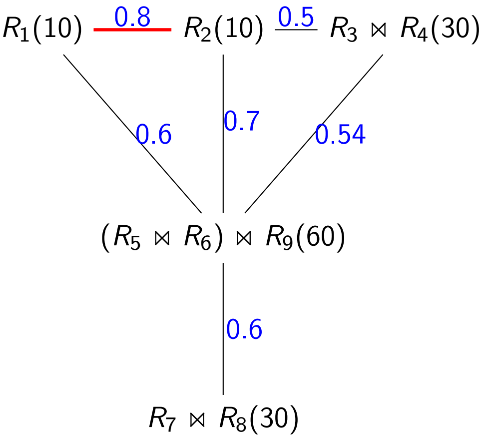
Example - step 3



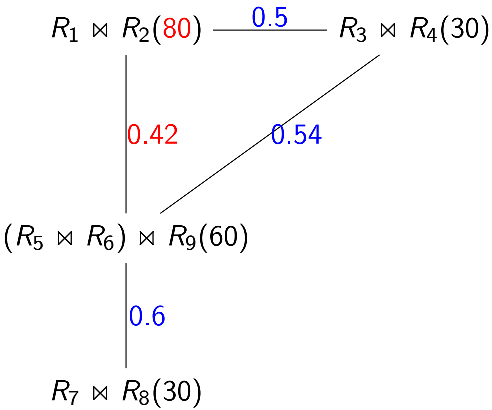
Example - after step 3



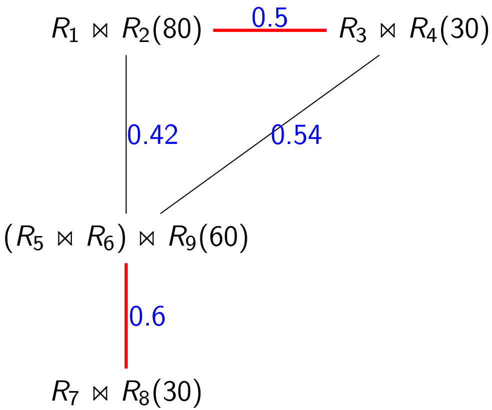
Example - step 4



Example - after step 4



Example - step 5



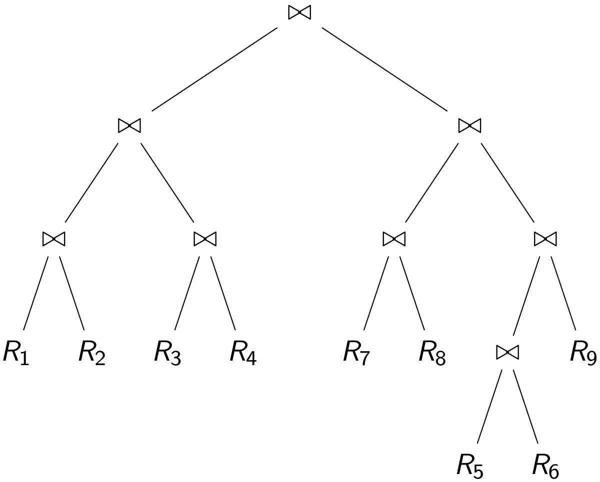
Example - after step 5

$$(R_1 \bowtie R_2) \bowtie (R_3 \bowtie R_4)(1200)$$

0.2268

$$(R_7 \bowtie R_8) \bowtie ((R_5 \bowtie R_6) \bowtie R_9)(1080)$$

Example - result



IKKBZ (informally)

Query graph Q is acyclic. Pick a root node, turn it into a tree. Run the following procedure for every root node, select the cheapest plan

Input: rooted tree Q

1. if the tree is a single chain, stop
2. find the subtree (rooted at r) all of whose children are chains
3. normalize, if $c_1 \rightarrow c_2$, but $rank(c_1) > rank(c_2)$ in the subtree rooted at r
4. merge chains in the subtree rooted at r, rank is ascending
5. repeat 1

IKKBZ (informally)

For every relation R_i we keep

- cardinality n_i
- selectivity s_i — the selectivity of the incoming edge from the parent of R_i
- cost $C(R_i) = n_i s_i$ (or 0, if R_i is the root)
- rank $r_i = \frac{n_i s_i - 1}{n_i s_i}$

Moreover,

- $C(S_1 S_2) = C(S_1) + T(S_1) C(S_2)$
- $T(S) = \prod_{R_i \in S}(s_i n_i)$
- rank of a sequence $r(S) = \frac{T(S) - 1}{C(S)}$

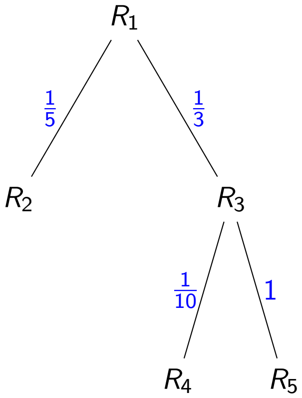
Understanding IKKBZ

- what is the rank?
- when is $(R_1 \bowtie R_2) \bowtie R_3$ cheaper than $(R_1 \bowtie R_3) \bowtie R_2$?

Understanding IKKBZ

- what is the rank?
- when is $(R_1 \bowtie R_2) \bowtie R_3$ cheaper than $(R_1 \bowtie R_3) \bowtie R_2$?
- if $r(R_2) < r(R_3)$!

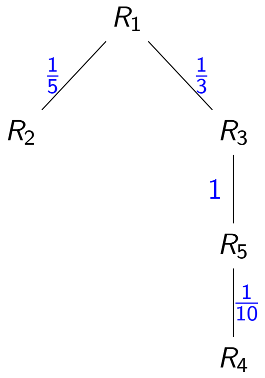
IKKBZ - example



Relation	n	s	C	T	*rank*
2	20	$\frac{1}{5}$	4	4	$\frac{3}{4}$
3	30	$\frac{1}{3}$	10	10	$\frac{9}{10}$
4	50	$\frac{1}{10}$	5	5	$\frac{4}{5}$
5	2	1	2	2	$\frac{1}{2}$

IKKBZ - example

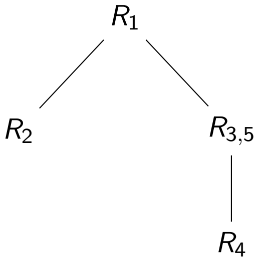
Subtree R_3: merging, $rank(R_5) < rank(R_4)$



Relation	n	s	C	T	*rank*
2	20	$\frac{1}{5}$	4	4	$\frac{3}{4}$
3	30	$\frac{1}{3}$	10	10	$\frac{9}{10}$
4	50	$\frac{1}{10}$	5	5	$\frac{4}{5}$
5	2	1	2	2	$\frac{1}{2}$

IKKBZ - example

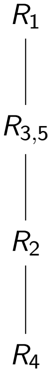
Subtree R_1:
$rank(R_3) > rank(R_5)$,
normalizing



Relation	n	s	C	T	*rank*
2	20	$\frac{1}{5}$	4	4	$\frac{3}{4}$
3	30	$\frac{1}{3}$	10	10	$\frac{9}{10}$
4	50	$\frac{1}{10}$	5	5	$\frac{4}{5}$
5	2	1	2	2	$\frac{1}{2}$
3,5	60	$\frac{1}{3}$	30	20	$\frac{19}{30}$

IKKBZ - example

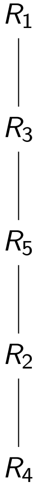
Subtree R_1: merging



Relation	n	s	C	T	*rank*
2	20	$\frac{1}{5}$	4	4	$\frac{3}{4}$
3	30	$\frac{1}{15}$	10	10	$\frac{9}{10}$
4	50	$\frac{1}{10}$	5	5	$\frac{4}{5}$
5	2	1	2	2	$\frac{1}{2}$
3,5	60	$\frac{1}{3}$	30	20	$\frac{19}{30}$

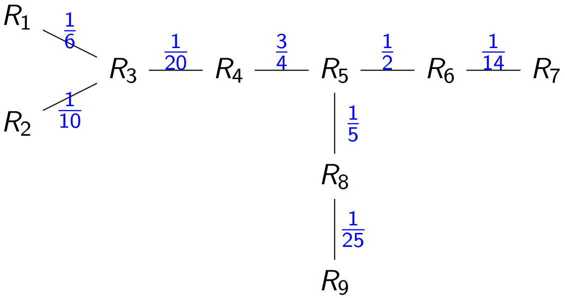
IKKBZ - example

Denormalizing



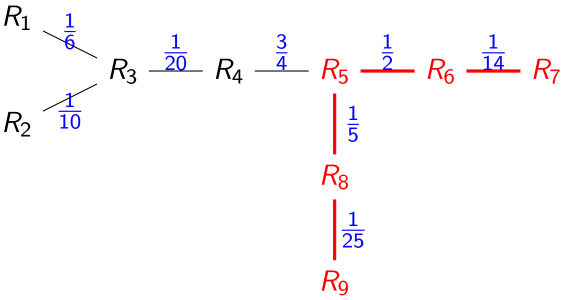
Relation	n	s	C	T	*rank*
2	20	$\frac{1}{5}$	4	4	$\frac{3}{4}$
3	30	$\frac{1}{15}$	10	10	$\frac{9}{10}$
4	50	$\frac{1}{10}$	5	5	$\frac{4}{5}$
5	2	1	2	2	$\frac{1}{2}$
3,5	60	$\frac{1}{3}$	30	20	$\frac{19}{30}$

IKKBZ - another example



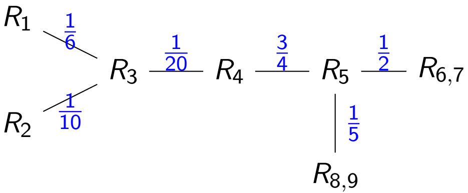
- $|R_1| = 30$
- $|R_2| = 100$
- $|R_3| = 30$
- $|R_4| = 20$
- $|R_5| = 10$
- $|R_6| = 20$
- $|R_7| = 70$
- $|R_8| = 100$
- $|R_9| = 100$

IKKBZ



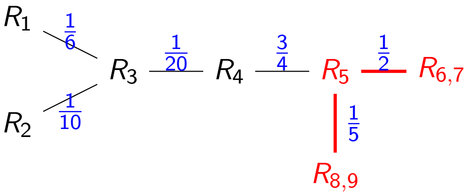
- $r(R_2) = \frac{9}{10}$
- $r(R_3) = \frac{4}{5}$
- $r(R_4) = 0$
- $r(R_5) = \frac{13}{15}$
- $r(R_6) = \frac{9}{10}$
- $r(R_7) = \frac{4}{5}$
- $r(R_8) = \frac{19}{20}$
- $r(R_9) = \frac{3}{4}$

IKKBZ



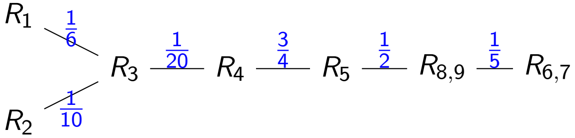
- $C(R_{8,9}) = 100$
- $T(R_{8,9}) = 80$
- $r(R_{8,9}) = \frac{79}{100} = \frac{237}{300}$
- $C(R_{6,7}) = 60$
- $T(R_{6,7}) = 50$
- $r(R_{6,7}) = \frac{49}{60} = \frac{245}{300}$

IKKBZ



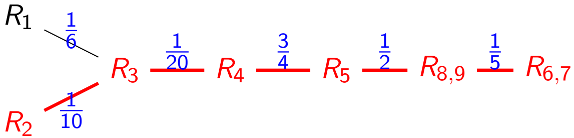
- $C(R_{8,9}) = 100$
- $T(R_{8,9}) = 80$
- $r(R_{8,9}) = \frac{79}{100} = \frac{237}{300}$
- $C(R_{6,7}) = 60$
- $T(R_{6,7}) = 50$
- $r(R_{6,7}) = \frac{49}{60} = \frac{245}{300}$

IKKBZ



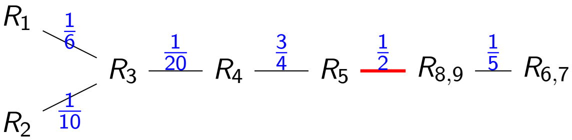
- $r(R_{8,9}) < r(R_{6,7})$

IKKBZ



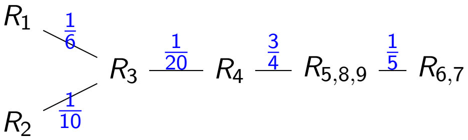
- $r(R_2) = \frac{9}{10}$
- $r(R_3) = \frac{4}{5}$
- $r(R_4) = 0$
- $r(R_6) = \frac{9}{10}$
- $r(R_7) = \frac{4}{5}$
- $r(R_5) = \frac{13}{15} = 0.86..$
- $r(R_{8,9}) = 0.79$

IKKBZ



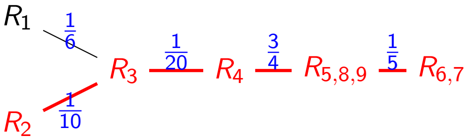
- $r(R_5) = \frac{13}{15} = 0.86..$
- $r(R_{8,9}) = 0.79$

IKKBZ



- $n_{5,8,9} = 800$
- $C_{5,8,9} = \frac{1515}{2}$
- $T_{5,8,9} = 600$
- $r(R_{5,8,9}) = \frac{1198}{1515} = 0.79..$
- $r(R_{6,7}) = 0.816..$

IKKBZ



- $r(R_2) = \frac{9}{10}$
- $r(R_{5,8,9}) = \frac{1198}{1515} = 0.79..$
- $r(R_3) = 0.8$
- $r(R_4) = 0$
- $r(R_{6,7}) = 0.816..$

IKKBZ

$$R_1 \text{ --- } R_3 \text{ --- } R_4 \text{ --- } R_{5,8,9} \text{ --- } R_{6,7} \text{ --- } R_2$$

IKKBZ

$$R_1 — R_3 — R_4 — R_5 — R_8 — R_9 — R_6 — R_7 — R_2$$

IKKBZ-based heuristics

What if Q has cycles?

- Observation 1: the answer of the query, corresponding to any subgraph of the query graph, is a superset of the answer to the original query
- Observation 2: a very selective join is more likely to be influential in choosing the order than a non-selective join

IKKBZ-based heuristics

What if Q has cycles?

- Observation 1: the answer of the query, corresponding to any subgraph of the query graph, is a superset of the answer to the original query
- Observation 2: a very selective join is more likely to be influential in choosing the order than a non-selective join

Choose the minimum spanning tree (minimize the product of the edge weights), compute the total order, compute the original query.

Homework: Task 1 (15 points)

- Give an example query qraph with join selectivities for which the greedy operator ordering (GOO) algorithm does not give the optimal (with regards to C_{out}) join tree. Specify the optimal join tree.
- For that example perform the IKKBZ-based heuristics

Homework: Task 2 (15 points)

- Using the program from the the last exercise as basis, construct the query graph for each connected component.

Info

- Slides and exercises: www3.in.tum.de/teaching/ss14/queryopt
- Send any comments, questions, solutions for the exercises etc. to Andrey.Gubichev@in.tum.de
- Exercises due: 9 AM, May 19, 2014