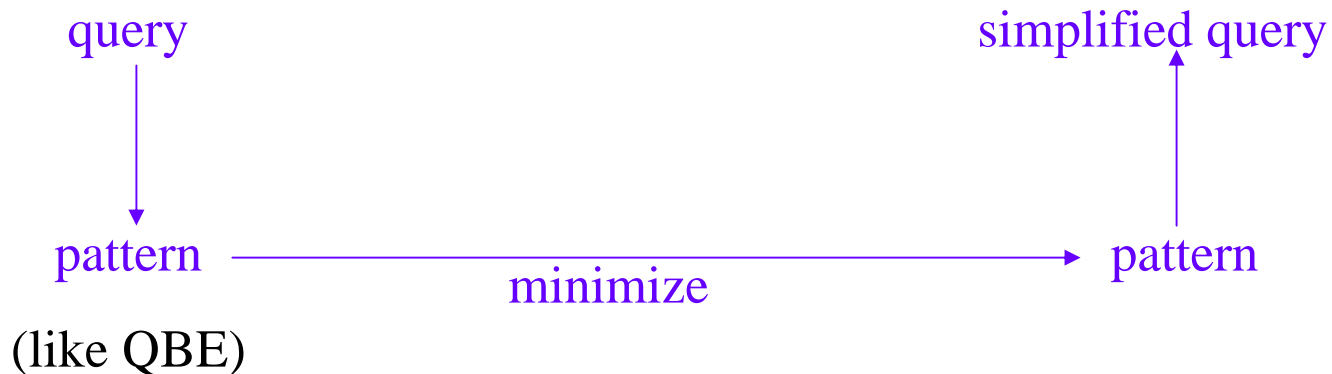


Exact Minimization of # of Joins

Possible for large set of queries:

- Basic, unnested SQL queries
- One-stage QBE queries
- Relational algebra with π , σ , $\triangleright\triangleleft$
- Relational calculus with \exists , $=$, \wedge

- Basic idea:



Minimization for a simple case

- Relational algebra with:
 - one input relation (can be extended to several)
 - $\pi, \triangleright \triangleleft$
 - σ_{cond} where cond is $A = \text{const}$

↓
rspj query

- First step: write query in QBE style, using a pattern \rightarrow tableau
 - e.g. tableau for $\pi_{AB}(R) \triangleright \triangleleft \pi_{BC}(R)$, where $R: ABC$, is:

A	B	C		
a	b	c ₁		
a ₁	b	c		
a	b	c		

|————— pattern
————— answer

Answer to a tableau query: example

The answer to the tableau query

A	B	C
a	b	c ₁
a ₁	b	c
a	b	c

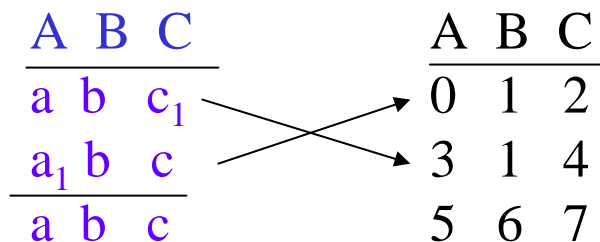
applied to

A	B	C
0	1	2
3	1	4
5	6	7

is:

A	B	C
0	1	4
3	1	2
0	1	2
3	1	4
5	6	7

For instance, the tuple $\langle 3,1,2 \rangle$ is obtained by the following mapping:



- a → 3
- b → 1
- c₁ → 4
- a₁ → 0
- c → 2

Tableaux

- Tableau over R: $\langle S, t \rangle$ where
 - S is the set of rows over R, with variables or constants
 - t is the “answer” rowt is over a subset of $\text{att}(R)$
t(A) is a variable or a constant
- Terminology
 - variables in t: distinguished (free)
 - denoted a, b, c
 - other variables: non-distinguished (quantified)
 - denoted $a_1 a_2 \dots b_1 b_2 \dots c_1 c_2$

Example

- Directors who are also actors

- QBE:

movie	title	dir	actor
		-d	
			-d
answer	dir		
I.	-d		

- Tableau:

	title	dir	actor
S	t ₁	d	a ₁
	t ₂	d ₁	d
T		d	

answer row

From rspj algebra queries to tableaux: an example

R: ABC

q: $\pi_{AC}(\pi_{AB}(R) \triangleright \triangleleft \pi_{BC}(\sigma_{A=5}(\pi_{AB}(R)) \triangleright \triangleleft \pi_{AC}(R)))$

Relational calculus (caution: use a different variables with each quantifier to avoid ambiguity!):

$\exists b_2[\exists c_1(R(ab_2c_1)) \wedge \exists a_1(\exists c_2(R(a_1b_2c_2)) \wedge a_1=5 \wedge \exists b_1(R(a_1b_1c)))]$

Prenex form (move \exists to left):

$\exists a_1b_1b_2c_1c_2[R(ab_2c_1) \wedge R(a_1b_2c_2) \wedge R(a_1b_1c) \wedge a_1=5]$

Replace a_1 by 5:

$\exists b_1b_2c_1c_2[R(ab_2c_1) \wedge R(5,b_2c_2) \wedge R(5,b_1c)]$

Example, continued

Tableau:

A	B	C
a	b_2	c_1
5	b_2	c_2
5	b_1	c
a		c

Note: # rows in tableau =
1 + # joins

Note: like QBE query

R	A	B	C
	-a	$-b_2$	
	5	$-b_2$	
	5		-c
answer	A	C	
I.	-a	-c	

Minimizing Tableau

- $\langle S, t \rangle$: tableau
- Definition
 - Mapping f on variables is a **homomorphism on $\langle S, t \rangle$** iff:
 - $f(t) = t$
 - $f(c) = c$ if c is constant
 - $f(S) \subseteq S$ (every row is mapped to an existing row in S)

Theorem: $\langle f(S), t \rangle$ is equivalent to $\langle S, t \rangle$

- Therefore, a row $r \notin f(S)$ is redundant
- **Minimization algorithm:** eliminate redundant rows until no longer possible

Example

- Tableau $\langle S, t \rangle$

A	B	C	
a	b_1	c_1	
a_1	b	c_1	
a	b_2	c_2	S
a_2	b_2	c	
a_2	b_1	c	
a	b	c	t

- Let f be defined by $c_2 \rightarrow c_1$
 $b_2 \rightarrow b_1$
all other variables stay unchanged

Example, continued

- $f(\langle S, t \rangle)$:

A	B	C
a	b_1	c_1
a_2	b_1	c
a_1	b	c_1
a	b	c

No redundant rows: MINIMAL

- Fact: all minimal equivalent tableaux are isomorphic!
(the same except possibly for the names of variables)
So the algorithm yields the minimum possible number of rows (and joins).

From tableau back to algebra after minimization

- Example:

A	B	C
a	b	c ₁
a ₁	b	c
a	b	c

- Write domain calculus query for the tableau:

$$\exists c_1 \exists a_1 (R(abc_1) \wedge R(a_1bc))$$

- Translate to algebra:

$$\pi_{ABC} [\delta_{C/C_1}(R) \triangleright \triangleleft \delta_{A/A_1}(R)]$$

- Renaming can **always** be avoided by quantifying (and projecting) as early as possible:

$$\exists c_1 \exists a_1 (R(abc_1) \wedge R(a_1bc)) \text{ is the same as}$$

$$\exists c_1 (R(abc_1)) \wedge \exists a_1 (R(a_1bc)) \quad \text{which translates to}$$

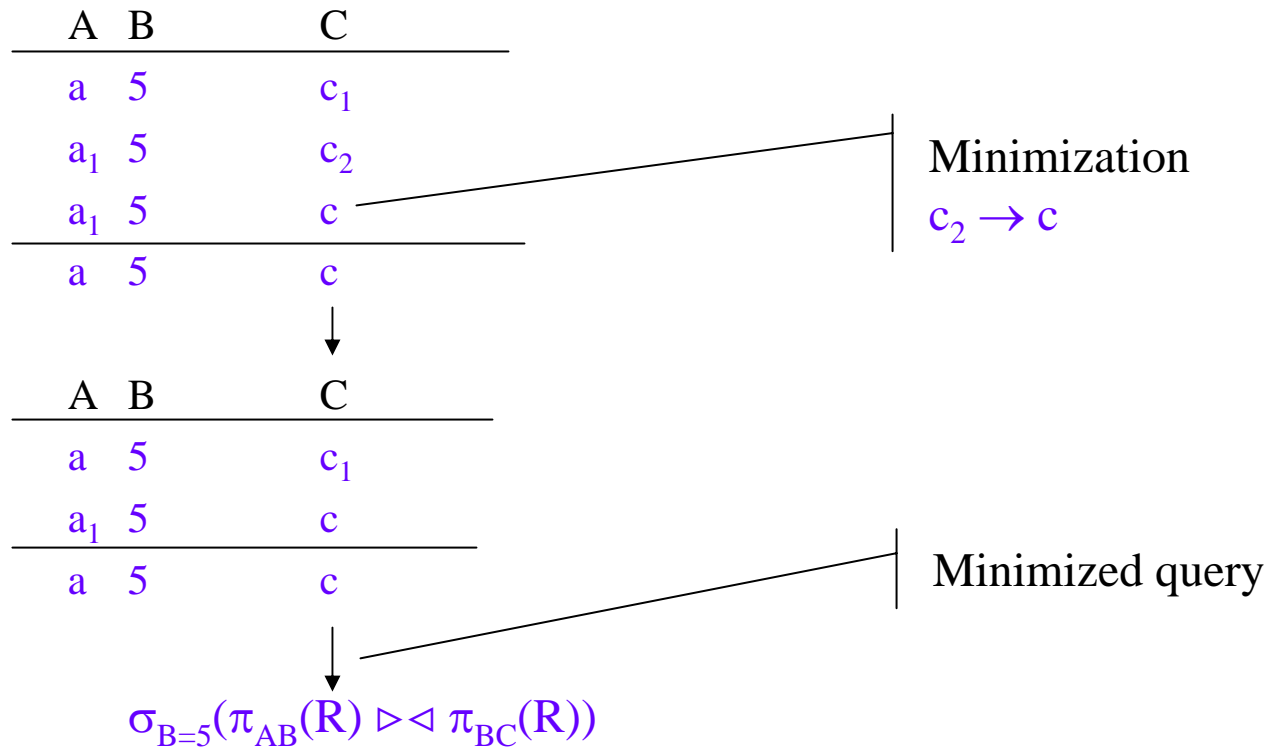
$$\pi_{AB}(R) \triangleright \triangleleft \pi_{BC}(R)$$

Example

R: ABC

q: $\pi_{AB}(\sigma_{B=5}(R)) \bowtie \pi_{BC}(\pi_{AB}(R) \bowtie \pi_{AC}(\sigma_{B=5}(R)))$

Tableau:



Functional Dependencies

- Dependencies: statements about properties of valid data
 - e.g.: “Every student is a person”
 - inclusion dependency
 - “Each employee works in no more than one department”
 - $NAME \rightarrow DEPARTMENT$
 - functional dependency
- Use of dependencies:
 - check data integrity
 - query optimization
 - schema design \rightarrow “normal forms”

Functional Dependencies

- Functional dependency over R:
 - expression $X \rightarrow Y$ where $X, Y \subseteq \text{att}(R)$
- A relation R **satisfies** $X \rightarrow Y$ iff whenever two tuples in R agree on X, they also agree on Y

e.g.

SCHEDULE	THEATER	TITLE
	la jolla	killer tomatoes
	hillcrest	tango

Satisfies $\text{THEATER} \rightarrow \text{TITLE}$

SCHEDULE	THEATER	TITLE
	la jolla	killer tomatoes
	hillcrest	tango
	hillcrest	splendor

Violates $\text{THEATER} \rightarrow \text{TITLE}$, satisfies $\text{TITLE} \rightarrow \text{THEATER}$

Using FDs in Query Optimization

- Example: R: ABC with $B \rightarrow C$

- query $\pi_{AB}(R) \triangleright \triangleleft \pi_{BC}(R)$

- Fact: if R satisfies $B \rightarrow C$ then

- $\pi_{AB}(R) \triangleright \triangleleft \pi_{BC}(R) = R$

- Why: tableau of query is

A	B	C
a	b	c_1
a_1	b	c
a	b	c

- If $\langle a, b, c_1 \rangle \in R$ and $\langle a_1, b, c \rangle \in R$ then $c_1 = c$

since R satisfies $B \rightarrow C$

- So, tableau is equivalent to $\frac{A \ B \ C}{a \ b \ c} \equiv R$

- In general: can simplify tableau $\langle S, t \rangle$ over R if R satisfies a set F of FDs.
- Algorithm: **The Chase**
 - Input: tableau $\langle S, t \rangle$, set F of FDs
 - Output: tableau $\text{CHASE}_F \langle S, t \rangle$ on all relations satisfying F
- Note: assume without loss of generality that FDs in F are of the form $X \rightarrow A$ where A is one attribute

The Chase

- Repeat until no change
 - For each $X \rightarrow A$ in F do
 - For each t_1, t_2 in s such that $t_1(X) = t_2(X), t_1(A) \neq t_2(A)$ do
 - if $t_1(A), t_2(A)$ are non-distinguished then replace one by the other in S
 - if $t_1(A)$ distinguished, $t_2(A)$ non-distinguished then replace $t_2(A)$ by $t_1(A)$ in S
 - if $t_1(A)$ is constant, $t_2(A)$ is variable then replace $t_2(A)$ by $t_1(A)$ in S
 - if $t_1(A)$ is constant, $t_2(A)$ is constant then STOP and output \emptyset

Optimization of RSPJ Queries with FDs

- q over R , set of FDs F over R
 - build tableau $\langle S, t \rangle$ of q
 - compute $\text{CHASE}_F \langle S, t \rangle$
 - minimize $\text{CHASE}_F \langle S, t \rangle$
 - construct rspj query from minimal tableau

Example

– R: ABC F = {B → A}

– $q = \pi_{BC}(\sigma_{A=5}(R)) \triangleright \triangleleft \pi_{AB}(R)$

<S, t>:

A	B	C
5	b	c
a	b	c ₁
a	b	c

CHASE<S, t>:A

A	B	C
5	b	c
5	b	c ₁
5	b	c

MIN:

A	B	C
5	b	c
5	b	c

RSPJ: $\sigma_{A=5}(R)$

Example

– R: ABC F = {B → A}, $q = \pi_{BC}(\sigma_{A=5}(R)) \triangleright \triangleleft \pi_{AB}(\sigma_{A=6}(R))$

$\langle S, t \rangle$:

A	B	C
5	b	c
6	b	c ₁
6	b	c

CHASE $\langle S, t \rangle$: \emptyset

QUERY: \emptyset

Example

– R: ABC F = {A → B}, $q = \pi_{AB}(R) \triangleright \triangleleft \pi_A(\sigma_{B=5}(R)) \triangleright \triangleleft \pi_{AB}(\pi_{AC}(R) \triangleright \triangleleft \pi_{BC}(R))$

<S, t>:

A	B	C
a	b	c ₁
a	b ₁	c ₂
a ₁	b	c ₂
a	5	c ₃
a	b	

CHASE<S, t>:

A	B	C
a	5	c ₁
a	5	c ₂
a ₁	5	c ₂
a	5	c ₃
a	5	

MIN:

A	B	C
a	5	c ₁
a	5	

RSPJ: $\pi_{AB}(\sigma_{B=5}(R))$