Minimalizácia konjunktínych dotazov

Ján Šturc

Jar, 2013

Motivačný príklad 1 – fp

- Funny path
  r_1: fp(x, y) ← e(x, y)
  r_2: fp(x, y) ← e(y, x), fp(z, x)
- Po rozvinutí rekurzie z pravidla r_2 dostaneme
  - r_{21}: fp(x,y) ← e(y, x),e(z_1,x)
    • • •
    r_{2k}: fp(x,y) ← e(y, x), e(z_1,x), e(z_2,x), …, e(z_k,x)
    • • •
    $r_{2k} \subseteq r_{21}$ prvé dva podciele sa mapujú identicky, na ďalšie sa nemapuje nič.
    $r_{21} \subseteq r_{2k}$ pre každé i > 1, z_i sa mapuje na z_1.
- Všetky rozvoje sú ekvivalentné a druhé pravidlo môžeme nahradiť nerekurzívnym pravidlom:
  r_2': fp(x,y) ← e(y, x), e(z, x)

Motivačný príklad 2 – EDM

- Daná je extenzionálna databáza ed(z,o), dm(o,v) a view edm = ed $\bowtie$ dm. Dotaz $Q_1 = \prod_o \sigma_{z='m'} edm$. Otázka je, či môžeme tento dotaz optimalizovať na $Q_2 = \prod_o \sigma_{z='m'} ed$?
- Preklad do rektifikovaného datalógu:
  Q_1: a(o) $\leftarrow$ ed(z, o), dm(o,v), z = 'm'
  Q_2: a(o) $\leftarrow$ ed(z, o), z = 'm'
- Zrejme $Q_1 \subseteq Q_2$. Opačné pohltenie neplatí.
- $Q_1 \not\equiv Q_2$. Ale v prípade, že ed a dm sa spája bezstrátove t.j. relácie ed a dm vznikli normalizáciou relácie edm napr. vzhľadom na funkčné závislosti o $\leftrightarrow$ v (oddelenie má práve jedného vedúceho), dávajú oba dotazy „vždy“ tú istú odpoveď.

Slabé pohltenie – weak containment

- Pohltenie bolo definované, že pre každú databázu D platí $Q_1(D) \subseteq Q_2(D)$.
- Oslabíme požiadavku pre každú len na databázy, ktoré sa spájajú bezstrátovo.

Definícia: Nech Q_1 a Q_2 sú dva dotazy na databázu s relačnou schémou $R_1, \ldots, R_n$. Hovoríme, že Q_2 slabo pohlcuje Q_1, ak pre každú databázu D so schémou $R_1, \ldots, R_n$ takú, že splňuje podmienku $\forall i(R_i = \prod_{R_i}(R_1 \bowtie \ldots \bowtie R_n))$ platí $Q_1(D) \subseteq Q_2(D)$. Píšeme $Q_1 \subseteq_w Q_2$.

- Predpoklad $\forall i(R_i = \prod_{R_i}(R_1 \bowtie \ldots \bowtie R_n))$ bezstrátovosti spojenia relácií D sa nazýva aj predpokladom „univerzálnej relácie / reprezentatívnej inštancie“. Tiež globálna konzistencia. Vzťahuje sa aj na podciele dotazu.

Poznámka

- Predpoklad „univerzálnej relácie“ je o schéme a dotazoch
  - Môžeme predpokladať že schéma vznikla normalizáciou (dekompozíciou) jedinej relácie
  - a dotazy sa týkajú len „spätných“ spojení dekomponovaných relácii podľa rovnako nazvaných atribútov.
- Relácie sú jednoznačne určené množinou svojich atribútov, ich mená sú zbytočné. Presne ako sme to robili pri normalizácii v minulom semestri.

Slabá ekvivalencia dotazov

Definícia: Hovoríme, že dotazy Q_1 a Q_2 sú slabo ekvivalentné, (Píšeme $Q_1 \cong_w Q_2$.) práve vtedy, ak $Q_1 \subseteq_w Q_2$ a $Q_2 \subseteq_w Q_1$.

- Teoreticky je možnosť pojem slabého pohltenia a slabej ekvivalencie zovšeobecniť pre ľubovoľnu množinu $\mathcal{C}$ podmienok (constraints) na databázy.
  Píšeme $Q_1 \subseteq_{\mathcal{C}} Q_2$ a $Q_1 \cong_{\mathcal{C}} Q_2$.
- Zrejme pohltenie, ekvivalencia implikuje slabé pohltenie, slabú ekvivalenciu.

Tablá – tableaux

- Tabló (tableau) je rektangulárne (obdlžníkovité) pole.
- Jeho stĺpce zodpovedajú atribútom („obľúbeným“ premenným), tieto sú zapísané v hlavičke (prvom riadku)
- Druhý (posledný) riadok (sumár) obsahuje hlavu dotazu. Premenné vyskytujúce sa v tomto riadku nazývame významné (distinguished). Zvyšné premenné sa nazývajú nevýznamné (nondistinguished).
- Riadky tabla zodpovedajú jednotlivým podcieľom dotazu.
- Tabló je množina riadkov (idempotencia $\wedge$)
- Konvencia: nevýznamné premenné, ktoré sa v table (dotaze) vyskytujú iba raz (v datalógu podčiarovník) možno nahradiť prazdným symbolom (blank).

Konštrukcia tabla k výrazu relačnej algebry

1. Výraz: $R(x_1, \ldots, x_n)$

   Tabló:

$x_1 \ldots x_n$
$x_1 \ldots x_n$
$x_1 \ldots x_n$

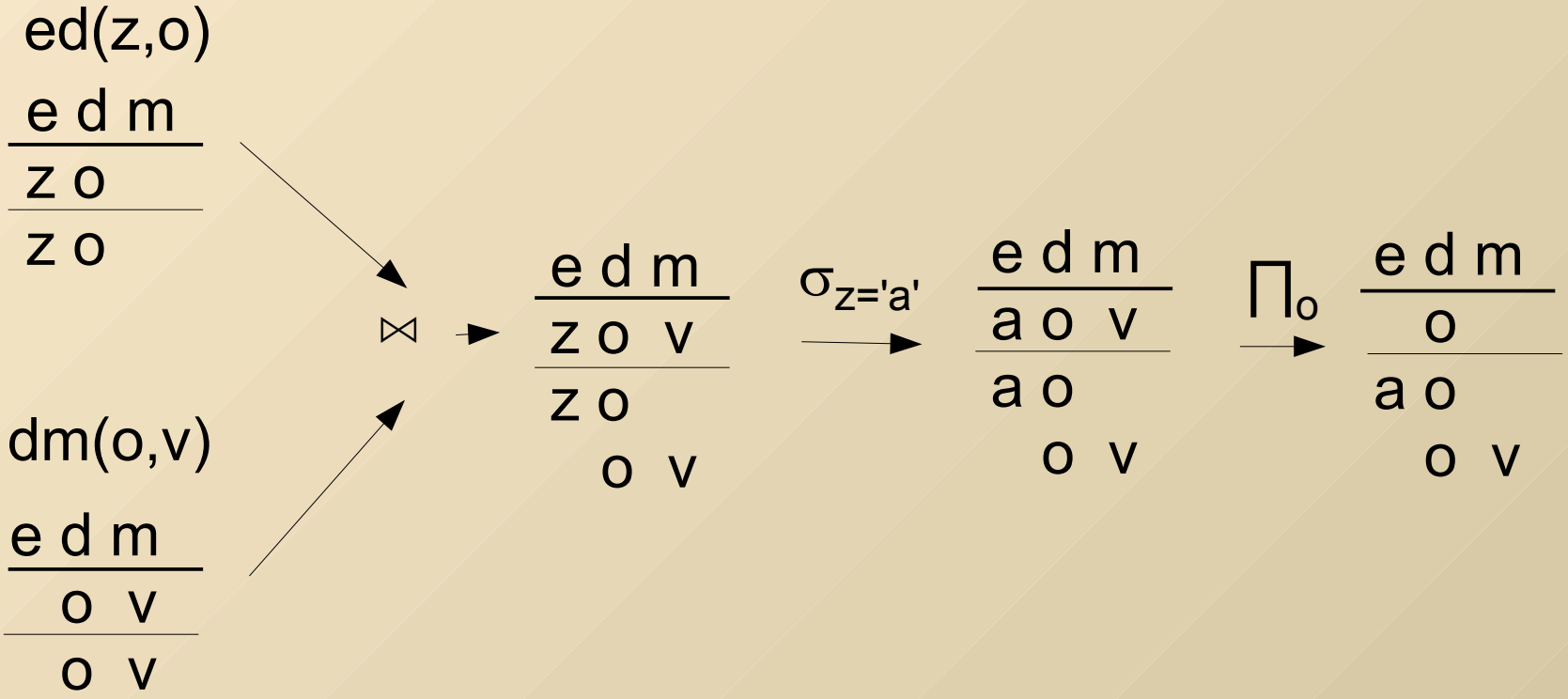
2. Ak T je tabló pre výraz $E(x_1, \ldots, x_n)$. Potom tabló pre $\sigma_{x_i=a}E$ dostaneme z tabla T tak, že v table T nahradíme všetky výskyty premennej x_i konštantou a.
3. Ak T je tabló pre výraz $E(x_1, \ldots, x_n)$. Potom tabló pre $\sigma_{x_i=x_j}E$ dostaneme z tabla T tak, že v table T stotožníme všetky výskyty premennej x_i a x_j.

Pozn.: Operácie 2, 3, sa vykonajú počnúc 2. riadkom. Hlavička zostane nezmenená.

Konštrukcia tabla k výrazu relačnej algebry – pokračovanie

4. Ak T je tabló pre výraz $E(x_1, \ldots, x_n)$. Potom tabló pre $\prod_{x_{i_1} \ldots x_{i_k}} E$, dostaneme z tabla T tak, že v sumári T premenné $x_{i_1}, \ldots, x_{i_k}$ nahradíme prázdnými znakmi. Vo zvyšných riadkoch zostanú, ale stanú sa nevýznamné.
5. Ak T je tabló pre $E(\mathbf{x}, \mathbf{y})$ a S je tabló pre $F(\mathbf{y}, \mathbf{z})$. Potom tabló pre $E(\mathbf{x}, \mathbf{y}) \bowtie F(\mathbf{y}, \mathbf{z})$ dostaneme z tabieľ T a S tak, že zjednotíme ich riadky a „unifikujeme“ ich sumáre.
   - Priorita: konštanta, premenná, blank
   - Unifikačnú substitúciu aplikujeme na všetky riadky
   - Ak sa nedajú unifikovať je sumár prázdny. Tabló mapuje každú EDB do prázdnej relácie. To nastane napr., ak sumáre majú na danej pozícii rôzne konštanty.

Príklad $\prod_{o}\sigma_{z='a'}edm$



Tablá a petrifikované dotazy

- Tabló je grafické znázornenie petrifikovaného dotazu.
  - Sumár je petrifikovaná hlava
  - Riadky sú petrifikované podciele tela
- Premenné v table zodpovedajú petrifikovaným premenným. Rozlíšenie medzi významnými a nevýznamnými premennými je informácia navyše.
- Pohlcujúci homomorfizmus medzi tablami je pohlcujúci homomorfizmus medzi petrifikovanými dotazmi.
- Ekvivalencia tabiel je ekvivalencia petrifikovaných dotazov.
- Predpoklad „univerzálnej relácie“ implikuje, že pohlujúci homomorfizmus nemusí „strážiť“ predikáty podcieľov, ale len ich premenné (atribúty).

Minimalizácia tabiel 1

- Veta o pohlcujúcom homorfizme platí aj pre tablá. Pri konštrukcii pohlcujúceho homomorfizmu sa významné premenné môžu zobraziť podobne ako konštanty len samé na seba.
- $T_1 \subseteq_w T_2$ práve vtedy, keď existuje pohlcujúci homomorfizmus η z T_2 do T_1.

Dôsledok: Nech r_1 a r_2 sú dva riadky tabla T, η je pohlcujúci homomorfizmus z riadku r_2 na riadok r_1, potom pre tabló $T' = T\eta - \{r_1\}$ platí: $T' \subseteq_w T$.

Dôkaz: Zrejmé. Pohlcujúce zobrazenie riadku na riadok je substitúcia. Pohlcujúci homomorfizmus celého tabla dostane zložením identickej substitúcie a η.

Minimalizácia tabiel 2

- Obrátené tvrdenie vo všeobecnosti neplatí. Ale platí

Veta: Nech Q je konjunktívny dotaz. Potom existuje konjunktívny dotaz Q', ktorého množina podcieľov je podmnožina množiny podcieľov Q a je minimálna.

Dôkaz: Nech T je tabló pre Q a t jeho sumár. Vyberieme riadok r a otestujeme, či existuje homomorfizmus φ taký, že $T\varphi \subseteq_w T - \{r\}$ a $s\varphi = s$. Homomorfizmus môže zobrazovať nevýznamné premenné na iné premenné alebo konštanty, vyskytujúce sa v riadku r.

Jednoduchý príklad

$fp(x,y) \leftarrow e(y, x), e(z_1, x), e(z_2, x), \ldots, e(z_k, x)$

T:

	Z	DO
	x	y
1	y	x
2	z_1	x
3	z_2	x
	. . .	
k	z_k	x

T':

	Z	DO
	x	y
1	y	x
2	z_1	x

Postupnými substitúciami $\varphi_i = [z_i \mapsto z_1]$ eliminujeme riadky 3, …, k. Dostaneme minimalizované tabló T'.

$T' \subseteq_w T; \quad \varphi = \{[z_i \mapsto z_1]\}_{i=2}^{k}$

$T \subseteq_w T'; \quad \iota =$ identická subst.

Teda $T' \cong_w T$.

Ekvivalentné minimálne tabló.

Iný príklad

R	A	B
	x	x
	x	y_1
	y_1	y_2
	$\vdots$	$\vdots$
	y_{n-1}	y_n
	y_n	x
	x	

q_1

R	A	B
	x	x
	x	

q_2

$q_1 \cong q_2$

Žiadná lokálna optimalizácia netransformuje q_1 na q_2.

Francúzi a nemci píšu sumár na spodok tabla.

Mali by sme tiež. Je to účtovnícky zvyk.

Iný príklad – edm

T:	e	d	m
	o	
a	o	v_1
e_1	o	v

T':	e	d	m
	o	
a	o	v_1

$\varphi = [e_1 \mapsto a, v \mapsto v_1]$; $T' \subseteq_w T$

$\iota =$ identická subst. ; $T \subseteq_w T'$

Znovu $T' \cong_w T$.

Napriek tomu tablá T' a T nereprezentujú logicky ekvivalentné dotazy, lebo riadky 1 a 2 zodpovedajú rôzným predikátom.

Dotaz T je ekvivalený tvrdeniu: $T' \wedge (\exists v) dm(o,v)$. V dôsledku bezstrátovosti spojenia ed $\bowtie$ dm to naozaj platí.

Ešte iný príklad

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

$(\mathbf{T}, u) =$

	R	A	B	C
		x	5	z_1
		x_1	5	z_2
		x_1	5	z
	u	x	5	z

$(\mathbf{T}', u) =$

	R	A	B	C
		x	5	z_1
		x_1	5	z
	u	x	5	z

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

Minimalizácia tabiel

- Predošlé tvrdenie umožňuje minimalizovať tablá postupným vynechávaním riadkov.
- Dokonca stačí hladať riadok, ktorý subsumuje (pohlcuje) iný riadok (term matching). $O(m^2n)$, kde m je počet riadkov a n dĺžka riadku.
- Pri blizšom pohľade sa tablá podobajú na petrifikované dotazy. Sumár zodpovedá petrifikovanej hlave a riadky tabla zodpovedajú petrifikovanému telu.
- Petrifikovať môžeme každý dotaz.
- Formálne môžeme aj ku každému dotazu zostaviť tabló, musíme však pre všetky argumenty všetkých podcieľov vytvoriť stĺpec. Je však otazné nakoľko pre takéto dotazy platí predpoklad „univerzálnej relácie“.

Uzatváracia procedúra – chase

- Môžeme využiť, že databáza splňuje nejaké podmienky (constraints) (napr.)
  - funkčné závislosti (fd)
  - multizávislosti (mvd)
  - spojovacie závislosti (jd)
  - inklúzne závislosti (ind)
- Ako? Pred alebo súčasne s minimalizáciou vynutíme, aby tabló splňovalo podmienky.

Vynutenie fd

- Nech **A** → B, kde **A** je množina atribútov a B atribút
- Vynutenie tejto závislosti na table T, znamená, že pre každé dva riadky r_1, r_2 platí, ak sa zhodujú v atribútoch (stĺpcoch) **A** musia sa zhodovať aj v stĺpci B.
- Vyberieme riadky r_1, r_2 otestujeme rovnosť $r_1.\mathbf{A} = r_2.\mathbf{A}$
- Pre stĺpec B sú nasledujúce možnosti
  - $r_1.B = r_2.B$, netreba nič robiť fd je splnená.
  - $r_1.B \neq r_2.B$ symboly sú rôzne konštanty. Nemôže nastať je to kontrapríklad proti fd, ktorá má platiť.
  - Riadky majú v pozícii B rôzne symboly. Stotožníme ich. Vyberieme jeden z nich, preferujeme najprv konštantu, potom významnú premennú. Všetky výskyty druhej premennej v table T nahradíme vybraným symbolom.

Zložitosť vynútenia fd

- Algoritmus končí, lebo každou aplikáciu fd ubudne jeden symbol z tabla.
- Nech *m* je počet riadkov, *n* počet atribútov a *f* počet funkčných závislostí.
- Existuje $\binom{m}{2}$ dvojíc riadkov
- Test najdenie **A** a porovnanie B je zložitosti *O(n)*.
- Pre nahradenie nezhody v najhoršom preskanujeme celé tabló *O(mn)*
- A máme *f* závislostí
- Celková zložitosť sa dá odhadnúť $O(fm^3n^2)$
- Možno to nie je presné, ale v každom prípade polynomiálna zložitosť.

Vynutenie mvd a jd tuple generating dependencies

- V schéme R(**x**,**y**,**z**) mvd **x** ↠ **y** znamená:
  $R(\mathbf{x},\mathbf{y}_1,\mathbf{z}_1) \wedge R(\mathbf{x},\mathbf{y}_2,\mathbf{z}_2) \Rightarrow R(\mathbf{x},\mathbf{y}_1,\mathbf{z}_2) \wedge R(\mathbf{x},\mathbf{y}_2,\mathbf{z}_1)$
- V tabulárnej forme to znamená doplnenie riadkov
- Podobne jd $\bowtie R_1, ..., R_n$ znamená, že ak existujú riadky pre jednotlivé relácie, existuje aj riadok pre ich spojenie

A	**B**	**C**
x	$\mathbf{y}_1$	$\mathbf{z}_1$
x	$\mathbf{y}_2$	$\mathbf{z}_2$
x	$\mathbf{y}_1$	$\mathbf{z}_2$
x	$\mathbf{y}_2$	$\mathbf{z}_1$

Uvedená mvd je to isté ako jd
⋈ **AB, BC**
Znamená to, že schémy sa spájajú bezstrátovo.
Vynútením mvd a jd môže počet riadkov v table exponenciálne rásť.

Vnorené závislosti – embeded dependencies

- Mvd a jd môžu byť aj vnorené znamená to, že sa nevzťahujú na celú množinu atribútov tabla, ale len na nejakú jej podmnožinu
- V schéme R(**x**,**y**,**z**,**v**) mvd **x** ↠ **y|z** znamená: $\exists(\mathbf{v}_1,\mathbf{v}_2,\mathbf{v}_3)$ $(R(\mathbf{x},\mathbf{y}_1,\mathbf{z}_1,\mathbf{v}_1) \wedge R(\mathbf{x},\mathbf{y}_2,\mathbf{z}_2,\mathbf{v}_2) \Rightarrow R(\mathbf{x},\mathbf{y}_1,\mathbf{z}_2,\mathbf{v}_3))$

Príklad:

A	B	C	D
x	$\mathbf{y}_1$	$\mathbf{z}_1$	$\mathbf{v}_1$
x	$\mathbf{y}_2$	$\mathbf{z}_2$	$\mathbf{v}_2$
x	$\mathbf{y}_1$	$\mathbf{z}_2$	$\mathbf{v}_3$
x	$\mathbf{y}_2$	$\mathbf{z}_1$	$\mathbf{v}_4$

Vynutenie ind

- Inklúzna závislosť znamená $\exists \mathbf{y} R(\mathbf{x},\mathbf{y}) \Rightarrow \exists \mathbf{z} S(\mathbf{x},\mathbf{z})$. Ničmenej premenné x sa môžu v R a S vyskytovať na pozícií rôznych atribútov.
- Inklúzna závislosť eventuálne generuje nejaké riadky do tabla.
- Ind v kombinácii s fd alebo mvd môžu spôsobiť zacyklenie tabló bude stále rásť. Algoritmus vynutenia neskončí. Nerozhodnuteľnosť.

„Silná“ ekvivalencia

Príklad:

$p(x,y) \leftarrow q(x,y), s(u,v)$

Tabló bez hlavičky

x	y	
x	y	
	u	v

x	y		
x	y		q
	u	v	s

Po doplnení anonýmnych premenných a minimalizacii. Prvý riadok pohltí druhý. Dostaneme slabo ekvivalentný dotaz: $p(x,y) \leftarrow q(x,y)$
Ekvivalencia neplatí ak s(u,v) je prázdne.
Aby sme zabránili nesprávnemu stotožneniu riadkov pridáme ku každému riadku tag – reláciu (pocieľ), z ktorého pochádza.

Konštrukcia tabiel pre všeobecné konjunktívne dotazy

- Presná konštrukcia:
  - Očíslujeme predikáty v nejakom poradí
  - Očíslujeme argumenty v rámci predikátov
- Lexikograficky usporiadané argumenty tvoria hlavičku tabla (nemusíme ju explicitne vypisovať).
- Premenné hlavy dotazu tvoria sumár. Na pozíciach a poradí nezáleží.
- Pre jednotlivé podciele zostavime riadok tabla, pričom
  - argumenty podcieľu sa musia nachádzať na správnej pozícii
  - na záver riadku pridáme tag – meno predikátu.
- Pri minimalizácii tag berieme ako konštantu, t.j. nedovolíme subsumciu medzi riadkami s rôznými tagmi.

Príklad: zistite vzájomné pohltenie

Q1: p ← r(X,Y), r(X,X), r(Z,X).

Q2: p ← r(X,Y), r(X,X), r(X,Z).

r(Y, Y)

R_{11}	R_{12}	R_{21}	R_{22}	R_{31}	R_{32}
X		X	X		X
X		X	X	X	

Optimalizácia Q1

R_{11}	R_{12}	
X		r
X	X	r
	X	r

Iné tablá pre Q1

X_1	Y	X_2	tag
X			r
X	X		r
X			r

X_1	Y	X_2	tag
X			r
X	X		r
	X		r

Pre cyklické dotazy konštrukcia alternatívnych tabieľ nie je jednoznačná.

Iný príklad

Q1: p ← r(X, X), r(U, U). X X U U

Q2: p ← r(X, Y), r(Z, X). X Y Z X

Q2 ⊇ Q1opak neplatí. Znovu sa to nedá urobiť lokálne. U sa nemôže súčasne zobraziť na Z aj X.

Alternatívna asi lepšia konštrukcia tabla pre všeobecné CQ.

- Stĺpce konštrujeme pre premenné a pozície v predikátoch
  - Ak sa premenná vyskytuje na rôznych miestách toho istého predikátu má toľko stĺpcov koľko je takýchto výskytov.
- Riadky pre jednotlivé predikáty.

Iný príklad alternatívne

X_1	X_2	U_1	U_2	Y	Z	Tag	Query
X	X					r_1	Q1
		U	U			r_2	Q1
X				Y		r_1	Q2
	X				Z	r_2	Q2

Tie U by som tam radšej nemal resp. mal ich v stĺpcoh pre X.