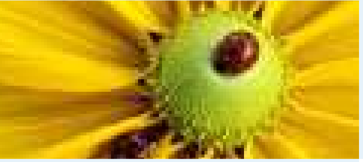
Lossless Decompositions in Complex-Valued Databases

Henning Koehler and Sebastian Link

Massey University, Palmerston North, New Zealand
[h.koehler|s.link]@massey.ac.nz



Relational Data Model

We want to characterize decompositions which are:

- lossless, and
- dependency preserving (for functional dependencies)

Relational Data Model:

Theorem 1. *Dependency-preserving decomposition is lossless iff it contains a key.*

$\Rightarrow$ Basis for synthesis approach to decomposition.

But: Only holds if domains are infinite (or ‘large enough’)!

Example

Let

$$
\begin{aligned}
R &= ABCD \\
\Sigma &= \{A \rightarrow BC, BC \rightarrow A\} \\
\mathcal{D} &= \{ABC, BD, CD\} \\
dom(A) &= \{0, 1\}
\end{aligned}
$$

Then $\mathcal{D}$ contains no key of R, but is lossless (easy to show).

This happens because

- $dom(A)$ is finite, **and**
- A appears in the LHS of $A \rightarrow BC$

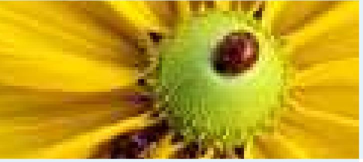
Relational Data Model - Finite Domains

Lemma 2. *If all attributes in LHS of a FD in Σ have infinite domains, then every lossless decomposition contains a key.*

Benefits:

- slightly stronger than Theorem 1
- can be generalized to complex-valued data model

Note: Lemma 2 holds for any cover of Σ.



Complex-Valued Data Model

Based on nesting of type constructors:

- flat attributes: $A, B, C, \ldots$
- record constructor: $(t_1, \ldots, t_n)$
- list constructor: $[t]$
- set constructor: $\{t\}$
- multiset constructor: $\langle t \rangle$

Subtyping by replacing ‘parts’ of type by trivial λ type:

$$(A, \lambda) < (A, [\lambda]) < (A, [B])$$

$\Rightarrow$ Can always project data onto subattributes.

Complex-Valued vs. Relational Model

relational model $\equiv$ flat attributes + record constructor

subschema $\equiv$ subattribute

But: Decompose into *sets* of subattributes:

$$\mathcal{D}_1 := \{ \quad \{\langle A\rangle\}, \quad \{\langle B\rangle\} \quad \}$$
$$\mathcal{D}_2 := \{ \quad \{\langle A\rangle, \langle B\rangle\} \quad \}$$
$$\mathcal{D}_3 := \{ \quad \{\langle A, B\rangle\} \quad \}$$

are all decompositions of $\langle A, B\rangle$, but semantically different!

$\Rightarrow$ extended subattributes (=normalized sets of subattributes)

Lossless Decompositions and Finite Domains

Can have finite domains for subattribute, even if all flat attributes have infinite domains ($dom(\lambda) = \{ok\}$):

$$dom(\{\lambda\}) = \{\emptyset, \{ok\}\}$$

indicating whether a set is empty or not.

$\Rightarrow$ Theorem 1 will never apply if set constructor is used.
$\Rightarrow$ Show Lemma 2 for complex-valued model.

Note: For FD $X \rightarrow Y$, X and Y are *extended* subattributes.

Lemma 2 for Complex-Valued Data Model

Definition 3. We call subattribute S *restricted* if $dom(S)$ is finite. $X \rightarrow Y$ is *LHS-restricted* if some element in X is restricted. Σ is LHS-restricted if any of its FDs are.

Lemma 4. *If Σ is not LHS-restricted, then every lossless decomposition contains a key.*

Theorem 5. *If Σ is not LHS-restricted, then a dependency preserving decomposition is lossless iff it contains a key.*

THE END

Thank You. Questions?