Rückblick

Membership-Test mit Armstrong-Axiomen

Sei $V = \{A, B, C, D, E, F, G, H, I\}$ und
$\mathcal{F} = \{AB \to E, BE \to I, E \to G, GI \to H\}$.
Es soll getestet werden, ob $AB \to GH \in \mathcal{F}^+$.

1. $AB \to E \overset{A2}{\Rightarrow} AB \to ABE$
2. $BE \to I \overset{A2}{\Rightarrow} ABE \to ABEI$
3. $\overset{A3}{\Rightarrow} AB \to ABEI$
4. $E \to G \overset{A2}{\Rightarrow} ABEI \to ABEGI$
5. $\overset{A3}{\Rightarrow} AB \to ABEGI$
6. $GI \to H \overset{A2}{\Rightarrow} ABEGI \to ABEGHI$
7. $\overset{A3}{\Rightarrow} AB \to ABEGHI$
8. $\overset{A1}{\Rightarrow} ABEGHI \to GH$
9. $\overset{A3}{\Rightarrow} AB \to GH$

Rückblick

Geben Sie einen Schlüssel für die Menge der FAs $\mathcal{F}$ an (1/3)

Sei $V = \{A, B, C, D, E, F, G, H, I\}$ und
$\mathcal{F} = \{AB \to E, BE \to I, E \to G, GI \to H\}$.
Initialisierung: $X := V$

(1) $(X \setminus \{A\})^+ \neq V$, da A nie auf der rechten Seite einer FA erscheint
(2) $(X \setminus \{B\})^+ \neq V$, da B nie auf der rechten Seite einer FA erscheint
(3) $(X \setminus \{C\})^+ \neq V$, da C nie auf der rechten Seite einer FA erscheint
(4) $(X \setminus \{D\})^+ \neq V$, da D nie auf der rechten Seite einer FA erscheint
(5) $(X \setminus \{F\})^+ \neq V$, da F nie auf der rechten Seite einer FA erscheint

Rückblick

Geben Sie einen Schlüssel für die Menge der FAs $\mathcal{F}$ an (2/3)

(6) Test auf $(X \setminus \{E\})^+ = V$ mit $\texttt{XPlus}(X \setminus \{E\}, V, \mathcal{F})$:
- $\texttt{result} = \{A, B, C, D, F, G, H, I\}$
- Runde 1: $AB \to E \Rightarrow \texttt{result} := \texttt{result} \cup E$
- Keine Veränderung von `result` in Runde 2
- $\Rightarrow (X \setminus \{E\})^+ = V$, somit $X := X \setminus \{E\}$.

(7) Test auf $(X \setminus \{G\})^+ = V$ mit $\texttt{XPlus}(X \setminus \{G\}, V, \mathcal{F})$:
- $\texttt{result} = \{A, B, C, D, F, H, I\}$
- Runde 1: $AB \to E \Rightarrow \texttt{result} := \texttt{result} \cup E$
- Runde 2: $E \to G \Rightarrow \texttt{result} := \texttt{result} \cup G$
- Keine Veränderung von `result` in Runde 3
- $\Rightarrow (X \setminus \{G\})^+ = V$, somit $X := X \setminus \{G\}$.

Rückblick

Geben Sie einen Schlüssel für die Menge der FAs $\mathcal{F}$ an (3/3)

(8) Test auf $(X \setminus \{H\})^+ = V$ mit $\texttt{XPlus}(X \setminus \{H\}, V, \mathcal{F})$:
- $\texttt{result} = \{A, B, C, D, F, I\}$
- Runde 1: $AB \to E \Rightarrow \texttt{result} := \texttt{result} \cup E$
- Runde 2: $E \to G \Rightarrow \texttt{result} := \texttt{result} \cup G$
- Runde 3: $GI \to H \Rightarrow \texttt{result} := \texttt{result} \cup H$
- Keine Veränderung von `result` in Runde 4
- $\Rightarrow (X \setminus \{H\})^+ = V$, somit $X := X \setminus \{H\}$.

(9) Test auf $(X \setminus \{I\})^+ = V$ mit $\texttt{XPlus}(X \setminus \{I\}, V, \mathcal{F})$:
- $\texttt{result} = \{A, B, C, D, F\}$
- Runde 1: $AB \to E \Rightarrow \texttt{result} := \texttt{result} \cup E$
- Runde 2: $E \to G \Rightarrow \texttt{result} := \texttt{result} \cup G$
- Runde 3: $BE \to I \Rightarrow \texttt{result} := \texttt{result} \cup I$
- Runde 4: $GI \to H \Rightarrow \texttt{result} := \texttt{result} \cup H$
- Keine Veränderung von `result` in Runde 5
- $\Rightarrow (X \setminus \{I\})^+ = V$, somit $X := X \setminus \{I\}$.

- $\Rightarrow X = \{A, B, C, D, F\}$ ist ein Schlüssel.

6.2.4 Minimale Überdeckung

Äquivalenz zweier Mengen von funktionalen Abhängigkeiten

- Seien $\mathcal{F}, \mathcal{G}$ Mengen von funktionalen Abhängigkeiten.
- $\mathcal{F}, \mathcal{G}$ nennen wir *äquivalent*, $\mathcal{F} \equiv \mathcal{G}$, wenn $\mathcal{F}^+ = \mathcal{G}^+$.
- Wir suchen eine *minimale Überdeckung* von $\mathcal{F}$.

Beispiel Rechtsreduktion

- $\mathcal{F}_1 = \{A \to B, B \to A, B \to C, A \to C, C \to A\}$.

  Kann die FA $B \to C$ zu $B \to \emptyset$ reduziert werden, d.h. gestrichen werden?

  Sei $\mathcal{F}_1' = \{A \to B, B \to A, A \to C, C \to A\}$.

  Gilt $\mathcal{F}_1^+ = \mathcal{F}_1'^+$? Ja, denn

  (a) $\mathcal{F}_1^+ \subseteq \mathcal{F}_1'^+$ wegen $XPlus(B, C, \mathcal{F}_1')$.
  (b) $\mathcal{F}_1^+ \supseteq \mathcal{F}_1'^+$ wegen $\mathcal{F}_1 \supseteq \mathcal{F}_1'$.

Beispiel Linksreduktion

- $\mathcal{F}_2 = \{AB \to C, A \to B, B \to A\}$.

  Kann die FA $AB \to C$ zu $B \to C$ reduziert werden, d.h. $AB \to C$ durch $B \to C$ ersetzt werden?

  Sei $\mathcal{F}_2' = \{B \to C, A \to B, B \to A\}$.

  Gilt $\mathcal{F}_2^+ = \mathcal{F}_2'^+$? Ja, denn

  (a) $\mathcal{F}_2^+ \subseteq \mathcal{F}_2'^+$ wegen (A2) und (A6) angewendet auf $B \to C$.
  (b) $\mathcal{F}_2^+ \supseteq \mathcal{F}_2'^+$ wegen $XPlus(B, C, \mathcal{F}_2)$.

Links- und Rechtsreduktion

- Eine Menge $\mathcal{F}$ funktionaler Abhängigkeiten heißt *linksreduziert*, wenn sie die folgende Eigenschaft erfüllt.

  Wenn $X \to Y \in \mathcal{F}, Z \subseteq X$, dann $\mathcal{F}' = (\mathcal{F} \setminus \{X \to Y\}) \cup \{Z \to Y\}$ nicht äquivalent zu $\mathcal{F}$.

  Linksreduktion: ersetze $X \to Y$ in $\mathcal{F}$ durch $Z \to Y$.
- Sie heißt *rechtsreduziert*, wenn $X \to Y \in \mathcal{F}, Z \subseteq Y$, dann $\mathcal{F}' = (\mathcal{F} \setminus \{X \to Y\}) \cup \{X \to Z\}$ nicht äquivalent zu $\mathcal{F}$.

  Rechtsreduktion: ersetze $X \to Y$ in $\mathcal{F}$ durch $X \to Z$.

Entscheidung mittels XPlus-Algorithmus

- Sei $X \to Y$ eine Abhängigkeit in $\mathcal{F}$ und sei $Z \to Y$, wobei $Z \subseteq X$.
  Wir führen die entsprechende Linksreduktion durch, wenn *XPlus*$(Z, Y, \mathcal{F})$ das Ergebnis `true` liefert.
- Sei $X \to Y$ eine Abhängigkeit in $\mathcal{F}$ und sei $X \to Z$, wobei $Z \subseteq Y$.
  Wir führen die entsprechende Rechtsreduktion durch, wenn *XPlus*$(X, Y, \mathcal{F}')$ das Ergebnis `true` liefert.

Satz

Sei eine Menge funktionaler Abhängigkeiten $\mathcal{F}$ gegeben und sei $\mathcal{F}'$ aus $\mathcal{F}$ durch eine Links- oder Rechtsreduktion hervorgegangen.

Dann $\mathcal{F} \equiv \mathcal{F}'$.

Minimale Überdeckung

Eine Menge funktionaler Abhängigkeiten $\mathcal{F}^{min}$ ist eine *minimale Überdeckung* zu $\mathcal{F}$, wenn wir sie durch Anwendung der folgenden Schritte erzeugen können:

- Führe alle möglichen Linksreduktionen durch.
- Führe alle möglichen Rechtsreduktionen durch.
- Streiche alle trivialen funktionalen Abhängigkeiten der Form $X \to \emptyset$.
- Vereinige alle funktionalen Abhängigkeiten mit gleicher linker Seite $X \to Y_1, \ldots, X \to Y_n$ zu einer einzigen FA der Form $X \to Y_1 \ldots Y_n$[1].

[1] Durch Anwendung von (A4).

Beispiel: Minimale Überdeckung

Sei $V = \{A, B, C, D\}$ und $\mathcal{F} = \{AB \to CD, B \to D, D \to C\}$.

1. Linksreduktion(en):
   - Prüfe, ob $\mathcal{F}' = (F \setminus \{AB \to CD\}) \cup \{B \to CD\}$ äquivalent zu $\mathcal{F}$ ist.
     $\Rightarrow$ *XPlus*$(B, CD, \mathcal{F})$: `true`
   - $\mathcal{F} := \{B \to CD, B \to D, D \to C\}$.
2. Rechtsreduktion(en):
   - Prüfe, ob $\mathcal{F}' = (F \setminus \{B \to CD\}) \cup \{B \to D\}$ äquivalent zu $\mathcal{F}$ ist.
     $\Rightarrow$ *XPlus*$(B, CD, \mathcal{F}')$: `true`
   - $\mathcal{F} := \{B \to D, D \to C\}$.
3. $F^{min} = \{B \to D, D \to C\}$

6.3 Verlustfreie und abhängigkeitsbewahrende Zerlegungen

- Sei ein Relationsschema gegeben durch eine Attributmenge V und eine Menge funktionaler Abhängigkeiten $\mathcal{F}$.
- Sei $\rho = \{X_1, \ldots, X_k\}$ eine *Zerlegung* von V.
- *Verlustfreiheit:* Sei $r \in \mathrm{Sat}(V, \mathcal{F})$ und seien $r_i = \pi[X_i]r$, $1 \leq i \leq k$ die Projektionen von r auf die einzelnen Elemente der Zerlegung. Jede Relation zu dem Ausgangsschema bleibt mittels $\bowtie$ aus den einzelnen Relationen der Zerlegung ρ exakt rekonstruierbar.

  Abhängigkeitsbewahrung: Die funktionalen Abhängigkeiten in $\mathcal{F}$ können auch über den Schemata der Zerlegung ρ ausgedrückt werden.

6.3.1 Verlustfreiheit

Sei $\rho = \{X_1, \ldots, X_k\}$ eine *Zerlegung* von V.

ρ heißt *verlustfrei*, wenn für jede Relation $r \in \mathrm{Sat}(V, \mathcal{F})$ gilt:

$$r = \pi[X_1]r \bowtie \ldots \bowtie \pi[X_k]r.$$

Beispiel

- Sei $V = \{A, B, C\}$ und $\mathcal{F} = \{A \to B, A \to C\})$.
- Sei $r \in Sat(V, \mathcal{F})$ wie folgt:

$$r = \begin{array}{ccc} A & B & C \\ \hline a_1 & b_1 & c_1 \\ a_2 & b_1 & c_2 \end{array}$$

- Seien $\rho_1 = \{AB, BC\}$ und $\rho_2 = \{AB, AC\}$.
- $r \subset \pi[AB]r \bowtie \pi[BC]r$,
  ρ_1 ist nicht verlustfrei.
- $r = \pi[AB]r \bowtie \pi[AC]r$,
  ρ_2 ist verlustfrei (für r).

Satz

Sei V eine Attributmenge mit einer Menge $\mathcal{F}$ funktionaler Abhängigkeiten. Sei $\rho = (X_1, X_2)$ eine Zerlegung von V.

ρ ist verlustfrei genau dann, wenn

$$(X_1 \cap X_2) \to (X_1 \setminus X_2) \in \mathcal{F}^+, \text{oder } (X_1 \cap X_2) \to (X_2 \setminus X_1) \in \mathcal{F}^+.$$

Korollar

Sei $R = (V, \mathcal{F})$ ein Relationsschema und sei $X \to Y \in \mathcal{F}$, wobei $X \cap Y = \emptyset$.

Dann ist die Zerlegung $\rho = (V \setminus Y, XY)$ verlustfrei.

Beweis: $(V \setminus Y) \cap XY = X$; $XY \setminus (V \setminus Y) = Y$.

6.3.2 Abhängigkeitsbewahrung

Beispiel

Sei $V = \{A, B, C, D\}$ und $\rho = \{AB, BC\}$.

- Betrachte $\mathcal{F} = \{A \to B, B \to C, C \to A\}$.
  Ist ρ abhängigkeitsbewahrend bzgl. $\mathcal{F}$?
- Betrachte die zu $\mathcal{F}$ äquivalente Menge $\mathcal{F}' = \{A \to B, B \to C, C \to B, B \to A\}$.
  Ist ρ abhängigkeitsbewahrend bzgl. $\mathcal{F}'$?

Definition

- Sei $R = (V, \mathcal{F})$ gegeben. Sei weiter $Z \subseteq V$.
- Sei die *Projektion* von $\mathcal{F}$ auf Z definiert zu

$$\pi[Z]\mathcal{F} = \{X \to Y \in \mathcal{F}^+ \mid XY \subseteq Z\}.$$

- Eine Zerlegung $\rho = \{X_1, \ldots, X_k\}$ von V heißt *abhängigkeitsbewahrend* bzgl. $\mathcal{F}$, wenn

$$\bigcup_{i=1}^{k} \pi[X_i]\mathcal{F} \equiv \mathcal{F}.$$

zum vorangehenden Beispiel

Sei $V = \{A, B, C, D\}$, $\rho = \{AB, BC\}$ und $\mathcal{F} = \{A \to B, B \to C, C \to A\}$.
Ist ρ abhängigkeitsbewahrend bzgl. $\mathcal{F}$?
Ja, denn

- $\{A \to B, B \to A\} \subseteq \pi[AB]\mathcal{F}$,
- $\{B \to C, C \to B\} \subseteq \pi[BC]\mathcal{F}$ und
- $\{A \to B, B \to A\} \cup \{B \to C, C \to B\} \equiv \mathcal{F}$.

Beobachtung: Nicht jede verlustfreie Zerlegung ist abhängigkeitsbewahrend!

- $R = (V, \mathcal{F})$, wobei $V = \{$`Stadt, Adresse, PLZ`$\}$,
- $\mathcal{F} = \{$`Stadt Adresse` $\to$ `PLZ, PLZ` $\to$ `Stadt`$\}$.
- $\rho = \{X_1, X_2\}$: $X_1 = \{$`Adresse, PLZ`$\}$ und $X_2 = \{$`Stadt, PLZ`$\}$.
- ρ ist verlustfrei, da $(X_1 \cap X_2) \to (X_2 \setminus X_1) \in \mathcal{F}$.
- ρ ist nicht abhängigkeitsbewahrend.

`Stadt Adresse` und `Adresse PLZ` sind Schlüssel zu R.

6.4 Normalformen

Sei $R = (V, \mathcal{F})$ ein Schema. Wir wollen eine Zerlegung $\rho = (X_1, \ldots, X_k)$ von R finden, die die folgenden Eigenschaften erfüllt:

- jedes $R_i = (X_i, \pi[X_i]\mathcal{F})$, $1 \leq i \leq k$ ist in einer gewünschten Normalform,
- ρ ist verlustfrei und (möglichst) auch abhängigkeitsbewahrend,
- k minimal.

Begriffe

- Sei X Schlüssel zu R und $X \subseteq Y \subseteq V$, dann nennen wir Y *Superschlüssel* von R.
- Gilt $A \in X$ für irgendeinen Schlüssel X von R, so heißt A *Schlüsselattribut (SA)* in R;
- gilt $A \notin X$ für jeden Schlüssel X, so heißt A *Nicht-Schlüsselattribut (NSA)*.

3. Normalform

Ein Relationsschema $R = (V, \mathcal{F})$ ist in *3. Normalform* (3NF) genau dann, wenn jedes NSA $A \in V$ die folgende Bedingung erfüllt.

Wenn $X \rightarrow A \in \mathcal{F}$, $A \notin X$, dann ist X ein Superschlüssel.

Die Bedingung der 3NF verbietet nichttriviale funktionale Abhängigkeiten $X \rightarrow A$, in denen ein NSA A in der Weise von einem Schlüssel K transitiv funktional abhängt, dass $K \rightarrow X$, $K \not\subseteq X$ gilt und des Weiteren $X \rightarrow A$.

Welche Art von Redundanz wird so vermieden?

Welche funktionale Abhängigkeiten verletzen die 3NF?

Stadt

SNr	SName	LCode	LFläche
7	Freiburg	D	357
9	Berlin	D	357
40	Moscow	RU	17075
43	St.Petersburg	RU	17075

Kontinent

KName	LCode	KFläche	Prozent
Europe	D	3234	100
Europe	RU	3234	20
Asia	RU	44400	80

Was ist hier zu sagen?

Stadt’

SNr	SName	LCode
7	Freiburg	D
9	Berlin	D
40	Moscow	RU
43	St.Petersburg	RU

Land’

LCode	LFläche
D	357
RU	17075

Lage’

LCode	KName	Prozent
D	Europe	100
RU	Europe	20
RU	Asia	80

Kontinent’

KName	KFläche
Europe	3234
Asia	44400

Boyce-Codd-Normalform

Ein Relationsschema $R = (V, \mathcal{F})$ ist in *Boyce-Codd-Normalform* (BCNF) genau dann, wenn die folgende Bedingung erfüllt ist.

Wenn $X \to A \in \mathcal{F}$, $A \notin X$, dann ist X ein Superschlüssel.

Die BCNF verschärft die 3NF.

- Sei $R = (V, \mathcal{F})$, wobei $V = \{$ `Stadt, Adresse, PLZ` $\}$, und $\mathcal{F} = \{$ `Stadt Adresse` $\to$ `PLZ, PLZ` $\to$ `Stadt`$\}$.
- R ist in 3NF, aber nicht in BCNF.
- Sei $\rho = \{$`Adresse PLZ, PLZ Stadt`$\}$ eine Zerlegung, dann erfüllt ρ die BCNF und ist verlustfrei, jedoch nicht abhängigkeitsbewahrend.

6.5 Algorithmen zur Normalisierung

BCNF-Analyse: verlustfrei und nicht abhängigkeitsbewahrend

Sei $R = (V, \mathcal{F})$ ein Relationsschema.

1. Sei $X \subset V$, $A \in V$ und $X \to A \in \mathcal{F}$ eine FA, die die BCNF-Bedingung verletzt.[2] Sei weiter $V' = V \setminus \{A\}$.

   Zerlege R in

   $$R_1 = (V', \pi[V']\mathcal{F}), \quad R_2 = (XA, \pi[XA]\mathcal{F}).$$

2. Teste die BCNF-Bedingung bzgl. R_1 und R_2 und wende den Algorithmus gegebenenfalls rekursiv an.

[2] Um eine in den gegebenen FAs ausgedrückte Zusammengehörigkeit von Attributen nicht zu verlieren, kann anstatt $X \to A$ die FA $X \to (X^+ \setminus X)$ betrachtet werden. Betrachte hierzu $V = \{A, B, C, D\}$ und $\mathcal{F} = \{A \to B, B \to C\}$.

Beispiel: BCNF-Analyse (1/2)

Sei $V = \{A, B, C, D\}$ und $\mathcal{F} = \{A \to B, B \to C\}$.
Betrachte folgende Zerlegungen mittels BCNF-Analyse:
Zerlegung 1:

(1) Wähle FD $A \to B$:
  (1.1) $R_1 = (\{A, C, D\}, \{A \to C\})$ und
  (1.2) $R_2 = (\{A, B\}, \{A \to B\})$.
(2) Weitere Zerlegung von R_1 mit FD $A \to C$ liefert:
  (2.1) $R_{1_1} = (\{A, D\}, \emptyset)$ und
  (2.2) $R_{1_2} = (\{A, C\}, \{A \to C\})$.
(3) Die Zerlegung $\rho_1 = \{R_{1_1}, R_{1_2}, R_2\}$ ist *nicht* abhängigkeitsbewahrend.
  $\Rightarrow$ Verliere FD: $B \to C$

Beispiel: BCNF-Analyse (2/2)

Sei $V = \{A, B, C, D\}$ und $\mathcal{F} = \{A \to B, B \to C\}$.
Betrachte folgende Zerlegungen mittels BCNF-Analyse:
Zerlegung 2:

(1) Wähle FD $A \to (A^+ \setminus A)$, d.h. $A \to BC$:
   (1.1) $R_1 = (\{A, D\}, \emptyset)$ und
   (1.2) $R_2 = (\{A, B, C\}, \{A \to B, A \to C, B \to C\})$.

(2) Weitere Zerlegung von R_2 mit FD $B \to C$ liefert:
   (2.1) $R_{2_1} = (\{A, B\}, \{A \to B\})$ und
   (2.2) $R_{2_2} = (\{B, C\}, \{B \to C\})$.

(3) Die Zerlegung $\rho_2 = \{R_1, R_{2_1}, R_{2_2}\}$ ist *abhängigkeitsbewahrend*.

3NF-Synthese: verlustfrei und abhängigkeitsbewahrend

Sei $R = (V, \mathcal{F})$ ein Relationsschema.

1. Sei $\mathcal{F}^{min}$ eine minimale Überdeckung zu $\mathcal{F}$.
2. Betrachte jeweils maximale Klassen von funktionalen Abhängigkeiten aus $\mathcal{F}^{min}$ mit derselben linken Seite. Seien $\mathcal{C}_i = \{X_i \to A_{i1}, X_i \to A_{i2}, \ldots\}$ die so gebildeten Klassen, $i \geq 0$.
3. Bilde zu jeder Klasse $\mathcal{C}_i$, $i \geq 0$, ein Schema mit Format $V_{\mathcal{C}_i} = X_i \cup \{A_{i1}, A_{i2}, \ldots\}$.[3]
4. Sofern keines der gebildeten Formate $V_{\mathcal{C}_i}$ einen Schlüssel für R enthält, berechne einen Schlüssel für R. Sei Y ein solcher Schlüssel. Bilde zu Y ein Schema mit Format $V_K = Y$.
5. $\rho = \{V_K, V_{\mathcal{C}_1}, V_{\mathcal{C}_2}, \ldots\}$ ist eine verlustfreie und abhängigkeitsbewahrende Zerlegung von R in 3NF.

[3] Der von uns betrachtete Algorithmus zur Berechnung von $\mathcal{F}^{min}$ hat diese Klassenbildung bereits vorgenommen.

Beispiel: 3NF-Synthese

Sei $V = \{A, B, C, D\}$ und $\mathcal{F} = \{A \to B, B \to C\}$.

- Es gilt, dass $\mathcal{F}^{min} = \mathcal{F}$
- Somit $V_{C_1} = \{A, B\}$ und $V_{C_2} = \{B, C\}$
- Da weder V_{C_1} noch V_{C_2} den Schlüssel $K = \{A, D\}$ enthält, muss die Zerlegung auch $V_K = \{A, D\}$ enthalten.
  $\Rightarrow \rho = \{V_{C_1}, V_{C_2}, V_K\}$ ist eine verlustfreie und abhängigkeitsbewahrende Zerlegung.

6.6 Empfohlene Lektüre

On the Structure of Armstrong Relations for Functional Dependencies

CATRIEL BEERI
The Hebrew University, Jerusalem, Israel
MARTIN DOWD
Rutgers University, New Brunswick, New Jersey
RONALD FAGIN
IBM Research Laboratory, San Jose, California
AND
RICHARD STATMAN
Rutgers University, New Brunswick, New Jersey

Abstract. An Armstrong relation for a set of functional dependencies (FDs) is a relation that satisfies each FD implied by the set but no FD that is not implied by it. The structure and size (number of tuples) of Armstrong relations are investigated. Upper and lower bounds on the size of minimal-sized Armstrong relations are derived, and upper and lower bounds on the number of distinct entries that must appear in an Armstrong relation are given. It is shown that the time complexity of finding an Armstrong relation, given a set of functional dependencies, is precisely exponential in the number of attributes. Also shown is the falsity of a natural conjecture which says that almost all relations obeying a given set of FDs are Armstrong relations for that set of FDs. Finally, Armstrong relations are used to generalize a result, obtained by Demetrovics using quite complicated methods, about the possible sets of keys for a relation. [4]

Elements of Relational Database Theory

Paris C. Kanellakis
Department of Computer Science
Brown University, P.O. Box 1910
Providence, RI 02912, USA.

Abstract: The goal of this paper is to provide a systematic and unifying introduction to relational database theory, including some of the recent developments in database logic programming. The first part of the presentation covers the two basic components of the relational data model: its specification component, that is the database scheme with dependencies, and its operational component, that is the relational algebra query language. The choice of basic constructs, for specifying the semantically meaningful databases and for querying them, is justified through an in-depth investigation of their properties. Some important research themes are reviewed in this context: the analysis of the hypergraph syntax of a database scheme and the extensions of the query language using deduction or universal relation assumptions. The subsequent parts of the presentation are structured around the two fundamental concepts illustrated in the first part, namely dependencies and queries. The main themes of dependency theory are implication problems and applications to database scheme design. Queries are classified in a variety of ways, with emphasis on the connections between the expressibility of query languages, finite model theory and logic programming. The theory of queries is very much related to research on database logic programs, which are an elegant formalism for the study of the principles of knowledge base systems. The optimization of such programs involves both techniques developed for the relational data model and new methods for analysing recursive definitions. The exposition closes with a discussion of how relational database theory deals with the problems of complex objects, incomplete information and database updates. [5]

[4] In: Journal of the ACM, Vol. 31, No. 1, 1984. Kann gegoogelt werden.
[5] In: Handbook of Theoretical Computer Science, Volume B: Formal Models and Sematics (B), 1990. Kann als Report gegoogelt werden.

Ausblick

3NF & BCNF

Seien gegeben

$$\begin{aligned} V &= \{SNr, SName, LCode, LFlaeche, KName, KFlaeche, Prozent\}, \\ \mathcal{F} &= \{SNr \to SName\ LCode, LCode \to LFlaeche, \\ &\qquad KName \to KFlaeche, LCode\ KName \to Prozent\}. \end{aligned}$$

(a) Geben Sie eine verlustfreie und abhängigkeitsbewahrende 3NF-Zerlegung an, indem Sie den 3NF-Synthese-Algorithmus anwenden.

(b) Geben Sie eine verlustfreie und möglichst abhängigkeitsbewahrende BCNF-Zerlegung an, indem Sie den BCNF-Analyse-Algorithmus anwenden.

Kennzeichnen Sie jeweils die Schlüssel der Schemata der gefundenen Zerlegungen.