Decomposition of Functional Relations

Examples

Closure Algorithm

Example 1. Consider a relation \( R(A, B, C, D, E, F) \) with asserted FDs

\[
\begin{align*}
(1) & \quad A \rightarrow B, C \\
(2) & \quad F \rightarrow E \\
(3) & \quad B, F, E \rightarrow D \\
(4) & \quad A, D \rightarrow E \\
(5) & \quad C, E, \rightarrow D, F
\end{align*}
\]

Problem 1: Find \( \{A\}^+ \).

Solution. We start with set \( X_0 = \{A\} \).

The left side of FD (1) is \( A \subseteq X_0 \). We set \( X_1 = \{A\} \cup \{B, C\} = \{A, B, C\} \).

The left sides of FDs (2),(3),(4),(5) are not proper subsets of \( X_1 \). Therefore, \( \{A\}^+ = X_1 = \{A, B, C\} \).

Problem 2: Find \( \{A, D\}^+ \).

Solution. We start with set \( X_0 = \{A, D\} \).

The left side of FD (1) is \( \{A\} \subseteq X_0 \). We set \( X_1 = \{A\} \cup \{B, C\} = \{A, B, C\} \).

The left side of FD (4) is \( \{A, D\} \subseteq X_1 \). We set \( X_2 = X_1 \cup \{E\} = \{A, B, C, D, E\} \).

The left side of FD (5) is \( \{C, E\} \subseteq X_2 \). We set \( X_3 = X_2 \cup \{D, F\} = \{A, B, C, D, E, F\} \).

\( X_3 \) contains all attributes from \( R \), therefore \( \{A, D\}^+ = X_3 = \{A, B, C, D, E, F\} \).

FD Projection Algorithm

Example 2. Consider a relation \( R(A, B, C, D, E, F) \) with asserted FDs

\[
\begin{align*}
(1) & \quad A \rightarrow B, C \\
(2) & \quad F \rightarrow E \\
(3) & \quad B, F, E \rightarrow D \\
(4) & \quad A, D \rightarrow E \\
(5) & \quad C, E, \rightarrow D, F
\end{align*}
\]

Problem 1: Find FDs asserted on \( R_1 = \pi_{A,B,C,D}(R) \)
Solution. Start with $S_0 = \emptyset$. We consider all subsets of $\{A, B, C, D\}$ in turn.

- $\{A\}$: left side of FD (1) is $A$. Right side of FD (1) is $\{B, C\} \subseteq \{A, B, C, D\}$. $S_1 = S_0 \cup \{A \rightarrow B, C\}$. No other FD can be matched.
  - $\{B\}$. No FD has left side $B$.
  - $\{C\}$. No FD has left side $C$.
  - $\{D\}$. No FD has left side $D$.
- $\{A, B\}$. FD (1) qualifies, but we have already dealt with it. No new FDs emerge.
- $\{A, C\}$. No new FDs emerge.
- $\{A, D\}$. FD (4) has left side $A, D$. It’s right side is $\{E\} \cap \{A, B, C, D\} = \emptyset$, hence no new FDs are added to $S_1$.
- $\{B, C\}, \{B, D\}, \{C, D\}$: no FDs emerge.
- $\{A, B, C\}, \{B, C, D\}, \{A, C, D\}, \{A, B, D\}$: no new FDs emerge.

Therefore, there is only one FD asserted on $R_1$: $A \rightarrow B, C$.

BCNF Decomposition

Example 3. Consider a relation $R(A, B, C, D, E, F)$ with asserted FDs

1. $A \rightarrow C, D$
2. $B \rightarrow E$
3. $A, E \rightarrow F$

Let us decompose $R$ into BCNF.

Step 1. Establish all keys of $R$. Using FD-Closure algorithm, we can establish that $A, B$ is the only key of $R$.

Step 2. Check if $R$ is in BCNF. We observe that none of the FDs above satisfy BCNF condition, as neither $A$, nor $B$, nor $A, E$ are superkeys.

Step 3. Decompose $R$. We pick FD (1). $\{A\}^+ = \{A, C, D\}$. We decompose $R$ into $R_1(A, C, D)$, and $R_2(A, B, E, F)$.

Step 3. Decompose $R_1$. Using FD-Project we assert $A \rightarrow C, D$ FD on $R_1$, and verify that $A$ is a key. $R_1$ is in BCNF.


Using FD-Closure we establish that $A, B$ is the only key. Then FD (2) violates BCNF.

$\{B\}^+ = \{B, E\}$. We decompose $R_2$ into $R_3(B, E)$ and $R_4(A, B, F)$.

Step 5. Decompose $R_3$. Using FD-Project we assert $B \rightarrow E$ on $R_3$. $B$ is the key, and $R_3$ is in BCNF.

Step 6. Decompose $R_4$. Using FD-Project we assert $A, B \rightarrow F$ on $R_4$. $A, B$ is the key, and $R$ is in BCNF.

Therefore, the decomposition of $R(A, B, C, D, E, F)$ into a BCNF schema is: $R_1(A, C, D), R_3(B, E), R_4(A, B, F)$.

Example 4. BCNF decomposition gone awry. Consider a relation $R(A, B, C)$ with the following FDs asserted:

1. $A \rightarrow B$
2. $B, C \rightarrow A$

Task: Decompose $R$ into BCNF.

$R$ is NOT in BCNF. $R$ has two keys: $B, C$ and $A, C$. FD (1) violates BCNF
condition (A is not a superkey). BCNF decomposition yields two relations:

\[ R1(A, B) \]
\[ R2(B, C) \]

Both relations are in BCNF.

Problem: We assert \( A \rightarrow B \) on \( R1 \). However, FD (2) cannot be asserted on either of the tables. This may lead to the following problem.

Consider the following instances of \( R1 \) and \( R2 \):

\[
\begin{array}{ccc}
\text{R1:} & \text{R2:} \\
A & B & B & C \\
a & b & b & a \\
b & b & b & d \\
\end{array}
\]

Let us compute \( R1 \bowtie R2 \):

\[
\begin{array}{ccc}
A & B & C \\
a & b & a \\
a & b & d \\
b & b & a \\
b & b & d \\
\end{array}
\]

We note the the FD \( A \rightarrow B \) holds on \( R1 \bowtie R2 \). However, the FD \( B, C \rightarrow A \) does not. Indeed, tuples \((a,b,a)\) and \((b,b,a)\) agree on values of \( B, C \) but NOT on values of \( A \).

We conclude that the BCNF decomposition of \( R \) does not preserve all functional dependencies.

Example 5. BCNF decomposition gone awry part 2. Consider the relation \( \text{Schedule} (\text{Classroom, Day, Course, Time, Instructor}) \) with with the following FDs asserted:

\[
\begin{align*}
(1) & \quad \text{Classroom, Day, Time} \rightarrow \text{Instructor, Course} \\
(2) & \quad \text{Instructor, Course} \rightarrow \text{Classroom, Time}
\end{align*}
\]

In previous examples we learned that this relation has two keys: \( \text{Classroom, Day, Time, Instructor, Course, Day} \).

Because every attribute of this table is prime, \( \text{Schedule} \) is in 3NF.

Because the left side of the FD (2) does not contain a key, \( \text{Schedule} \) is NOT in BCNF.

A BCNF decomposition would yield two tables:

\[
\begin{align*}
\text{Schedule1}(\text{Instructor, Course, Classroom, Time}) \\
\text{Schedule2}(\text{Instructor, Course, Day})
\end{align*}
\]

FD (2) is preserved on table \( \text{Schedule1} \).

Because the left side of FD (1) is not found in any table, FD (1) "disappears."

Consider the following instances of the tables \( \text{Schedule1} \) and \( \text{Schedule2} \):

\[
\begin{array}{cccc}
\text{Schedule1:} & \text{Schedule2:} \\
\text{Instructor} & \text{Course} & \text{Classroom} & \text{Time} \\
\text{Dekhtyar} & \text{CSC 366} & 14-253 & 12:10 \\
\text{Dekhtyar} & \text{CSC 468} & 14-252 & 12:10 \\
\end{array}
\]

\[
\begin{array}{cc}
\text{Instructor} & \text{Course} \\
\text{Dekhtyar} & \text{CSC 366} \ T \\
\text{Dekhtyar} & \text{CSC 366} \ R \\
\text{Dekhtyar} & \text{CSC 468} \ T \\
\text{Dekhtyar} & \text{CSC 468} \ R \\
\end{array}
\]

This combination of relational tables is consistent, as no FDs are violated. However, \( \text{Schedule1} \bowtie \text{Schedule2} \) yields:
### Schedule

<table>
<thead>
<tr>
<th>Instructor</th>
<th>Course</th>
<th>Classroom</th>
<th>Time</th>
<th>Day</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dekhtyar</td>
<td>CSC 366</td>
<td>14-253</td>
<td>12:10</td>
<td>T</td>
</tr>
<tr>
<td>Dekhtyar</td>
<td>CSC 366</td>
<td>14-253</td>
<td>12:10</td>
<td>R</td>
</tr>
<tr>
<td>Dekhtyar</td>
<td>CSC 468</td>
<td>14-252</td>
<td>12:10</td>
<td>T</td>
</tr>
<tr>
<td>Dekhtyar</td>
<td>CSC 468</td>
<td>14-252</td>
<td>12:10</td>
<td>R</td>
</tr>
</tbody>
</table>

This table violates FD (1) from above.

### 3NF Decomposition

**Example 6. 3NF Decomposition.** Consider a relational table $R(A, B, C, D, E, F)$ with the following FDs asserted:

- $A \rightarrow B$
- $A \rightarrow C$
- $C \rightarrow D$
- $D \rightarrow F$
- $E, B \rightarrow F$

First, let us note that the set of FDs above forms a minimal basis. (try removing any FD, and deriving it from the rest).

Second, we note that $R$ is not in 3NF, due to the presence of transitive FDs.

We use 3NF-Decompose algorithm to create the following tables:

- $R_1(A, B)$
- $R_2(A, C)$
- $R_3(C, D)$
- $R_4(D, F)$
- $R_5(B, E, F)$
- $R_6(A, E)$

Note, that each table $R_1 - R_6$ is in 3NF, and each FD from above is preserved in at least one table.

Note, also, that this is not the most concise decomposition of $R$. In fact, we can combine $R_1$ and $R_2$ into a single relation $R_{12}(A, B, C)$.

Finally, we note that $R_6$ has been created because none of $R_1 - R_5$ contained a full key of $R$. 