The Chase

Stéphane Bressan

Let Σ be a set of functional and multi-valued dependencies on a relation schema R. The Chase is an algorithm that solves the decision problem of whether a functional or multi-valued dependency (or join dependency) σ is satisfied by R with a set of functional and mutli-valued (and join) dependencies Σ.

$$(R \text{ with } \Sigma) \models \sigma?$$

Example 1

$$\{\{A\} \twoheadrightarrow \{B, C\}, \{D\} \rightarrow \{C\}\} \models \{A\} \rightarrow \{C\}?$$

Example 2

$$R = \{A, B, C, D\}$$

$$\{\{A\} \twoheadrightarrow \{B\}, \{B\} \twoheadrightarrow \{C\}\} \models \{A\} \twoheadrightarrow \{C\}?$$

Example 3

$$\{\{A\} \twoheadrightarrow \{B, C\}, \{C, D\} \rightarrow \{B\}\} \models \{A\} \rightarrow \{B\}?$$

Example 1

$$\{\{A\} \twoheadrightarrow \{B, C\}, \{D\} \rightarrow \{C\}\} \models \{A\} \rightarrow \{C\}?$$

Create an instance r on the schema $\{A, B, C, D\}$ with two t-uples and distinct values for all attributes.

A	B	C	D
a_1	b_1	c_1	d_1
a_2	b_2	c_2	d_2

Example 1 (Cont.)

$$\{\{A\} \twoheadrightarrow \{B, C\}, \{D\} \rightarrow \{C\}\} \models \{A\} \rightarrow \{C\}?$$

We want to chase $\{A\} \rightarrow \{C\}$.
Make the A-values the same.

$$a_1 = a_2$$

A	B	C	D
a_1	b_1	c_1	d_1
a_1	b_2	c_2	d_2

Example 1 (Cont.)

$$\{\{A\} \twoheadrightarrow \{B,C\}, \{D\} \rightarrow \{C\}\} \models \{A\} \rightarrow \{C\}?$$

Use $\{A\} \twoheadrightarrow \{B,C\}$. Create (two) new t-uples by copying the (two) t-uples that have the same A-value but swapping their B- and C-values. The multi-valued dependency generates t-uples. It is a t-uple generating dependency.

A	B	C	D
a_1	b_1	c_1	d_1
a_1	b_2	c_2	d_2
a_1	b_2	c_2	d_1
a_1	b_1	c_1	d_2

Example 1 (Cont.)

$$\{\{A\} \twoheadrightarrow \{B, C\}, \{D\} \rightarrow \{C\}\} \models \{A\} \rightarrow \{C\}?$$

Use $\{D\} \rightarrow \{C\}$. For each pair of t-uple with the same D-value, make their C-value the same.

$$c_1 = c_2$$

The functional dependency generates values. It is a value generating dependency.

A	B	C	D
a_1	b_1	c_1	d_1
a_1	b_2	c_1	d_2
a_1	b_2	c_1	d_1
a_1	b_1	c_1	d_2

Example 1 (Cont.)

$$\{\{A\} \twoheadrightarrow \{B, C\}, \{D\} \rightarrow \{C\}\} \models \{A\} \rightarrow \{C\}?$$

There is nothing else to do. We observe that r satisfies $\{A\} \rightarrow \{C\}$. Therefore the answer is yes

$$r \models \{A\} \rightarrow \{C\}$$

A	B	C	D
a_1	b_1	c_1	d_1
a_1	b_2	c_1	d_2
a_1	b_2	c_1	d_1
a_1	b_1	c_1	d_2

A	B	C	D
a_1	b_1	c_1	d_1
a_1	b_2	c_1	d_2
a_1	b_2	c_1	d_1
a_1	b_1	c_1	d_2

Example 1 (Cont.)

r also satisfies $\{D\} \rightarrow \{A\}$ but this is a coincidence. We can only answer the question about $\{A\} \rightarrow \{C\}$.
Another chase is needed for $\{D\} \rightarrow \{A\}$. Do it!

Example 2

$$R = \{A, B, C, D\}$$

$$\{\{A\} \twoheadrightarrow \{B\}, \{B\} \twoheadrightarrow \{C\}\} \models \{A\} \twoheadrightarrow \{C\}?$$

Create an instance r on the schema $\{A, B, C, D\}$ with two t-uples and distinct values for all attributes.

A	B	C	D
a_1	b_1	c_1	d_1
a_2	b_2	c_2	d_2

Example 2 (Cont.)

$$R = \{A, B, C, D\}$$

$$\{\{A\} \twoheadrightarrow \{B\}, \{B\} \twoheadrightarrow \{C\}\} \models \{A\} \twoheadrightarrow \{C\}?$$

We want to chase $\{A\} \twoheadrightarrow \{C\}$.
Make the A-values the same.

$$a_1 = a_2$$

A	B	C	D
a_1	b_1	c_1	d_1
a_1	b_2	c_2	d_2

Example 2 (Cont.)

$$R = \{A, B, C, D\}$$

$$\{\{A\} \twoheadrightarrow \{B\}, \{B\} \twoheadrightarrow \{C\}\} \models \{A\} \twoheadrightarrow \{C\}?$$

Use $\{A\} \twoheadrightarrow \{B\}$.

A	B	C	D
a_1	b_1	c_1	d_1
a_1	b_2	c_2	d_2
a_1	b_2	c_1	d_1
a_1	b_1	c_2	d_2

Example 2 (Cont.)

$$\{\{A\} \twoheadrightarrow \{B\}, \{B\} \twoheadrightarrow \{C\}\} \models \{A\} \twoheadrightarrow \{C\}?$$

Use $\{B\} \twoheadrightarrow \{C\}$ (twice, for b_1 and for b_2).

A	B	C	D
a_1	b_1	c_1	d_1
a_1	b_2	c_2	d_2
a_1	b_2	c_1	d_1
a_1	b_1	c_2	d_2
a_1	b_1	c_2	d_1
a_1	b_1	c_1	d_2
a_1	b_2	c_1	d_2
a_1	b_2	c_2	d_1

Example 2 (Cont.)

There is nothing else to do. We observe that r satisfies $\{A\} \twoheadrightarrow \{C\}$. Therefore the answer is yes

$$r \models \{A\} \twoheadrightarrow \{C\}$$

A	B	C	D
a_1	b_1	c_1	d_1
a_1	b_2	c_2	d_2
a_1	b_2	c_1	d_1
a_1	b_1	c_2	d_2
a_1	b_1	c_2	d_1
a_1	b_1	c_1	d_2
a_1	b_2	c_1	d_2
a_1	b_2	c_2	d_1

Example 3

$$\{\{A\} \twoheadrightarrow \{B, C\}, \{C, D\} \rightarrow \{B\}\} \models \{A\} \rightarrow \{B\}?$$

A	B	C	D
a_1	b_1	c_1	d_1
a_2	b_2	c_2	d_2

Example 3 (cont.)

$$\{\{A\} \twoheadrightarrow \{B,C\}, \{C,D\} \rightarrow \{B\}\} \models \{A\} \rightarrow \{B\}?$$

Use $\{A\} \twoheadrightarrow \{B,C\}$.

A	B	C	D
a_1	b_1	c_1	d_1
a_1	b_2	c_2	d_2
a_1	b_2	c_2	d_1
a_1	b_1	c_1	d_2

Example 3 (cont.)

There is nothing else to do.

$$r \not\models \{A\} \rightarrow \{B\}$$

Therefore the answer is No

A	B	C	D
a_1	b_1	c_1	d_1
a_1	b_2	c_2	d_2
a_1	b_2	c_2	d_1
a_1	b_1	c_1	d_2

We have built a counter-example.

The Power of The Chase

What is surprising and powerful is that The Chase is complete: it can prove or disprove that a functional or multi-valued dependency is satisfied!

Theorem

The Chase always builds a counter example if it exists and does not if it does not exists.

Setting The Chase

Let Σ be a set of functional and multi-valued dependencies on a relation schema R.
Let σ be a be a functional and multi-valued dependency.

$$\sigma = X \rightarrow Y \text{ or } \sigma = X \twoheadrightarrow Y$$

1. Create a table r with schema R with two tuples with all different values.
2. For each $A \in X$, make the A-values the same.

If R is not given, then use the attributes in Σ and σ.

Chasing The Chase

Repeat the following until you reach a fixed point (nothing changes):

1. For each functional dependency $Z \to V \in \Sigma$.
   1.1 If there are tuples in the table with same Z-value, then set their V-values to be the same.
2. For each multi-valued dependency $Z \twoheadrightarrow V \in \Sigma$.
   2.1 If there are two tuples in the table with same Z-value, then add two new tuples with all the same values and except for their V-values that are swapped.

Exit with:

$$r \models \sigma \quad \text{is equivalent to} \quad \Sigma \models \sigma$$

This means that you only need to check whether or not r satisfies the functional or multi-valued depedency σ that you were chasing.

Theorem

The Chase is sound and complete for σ.

$$r \models \sigma \quad \textit{is equivalent to} \quad \Sigma \models \sigma$$

Theorem

The Chase always terminates.

How to use to check to check that a decompostion is lossless?

Testing if a Decomposition is Lossless

Recall that on a relation schema $R = X \cup Y \cup Z$ with X, Y, Z disjoint, a multi-valued dependency $X \twoheadrightarrow Y$ holds in r iff:

$$\pi_{X \cup Y}(r) \bowtie \pi_{X \cup Z}(r) = r$$

This means multi-valued dependencies can be used to test if a relation schema $R = X \cup Y \cup Z$ (satisfying a set of dependencies Σ) can be decomposed into two relations $R_1 = X \cup Y$ and $R_2 = X \cup Z$ in a lossless manner:

Use the chase to decide whether $\Sigma \models X \twoheadrightarrow Y$
(or, equivalently, $\Sigma \models X \twoheadrightarrow Z$)

Setting and Chasing The Chase with Distinguished Variables

Let Σ be a set of functional and multi-valued dependencies on a relation schema R.

1. Create a table r with schema R with two tuples with all different values.
2. Mark some values as "distinguished" (e.g. use a different letter) depending on the task at hand:
   2.1 Distinguish the variables in the left-hand-side for chasing a functional dependency (as seen above). Check the existence of a column of distinguished variables.
   2.2 Distinguish the two fragments for chasing a multi-valued dependency (as seen above). Check the existence of a row of distinguished variables.
   2.3 Distinguish the multiple fragments to test a lossless decomposition (generalisation of what is discussed above to more than two fragments). Check the existence of a row of distinguished variables.
   2.4 Distinguish X to chase all Y such that $X \twoheadrightarrow Y$. Check the existence of distinguished variables (not discussed in the lecture).

Example 1 (Recall)

$\{\{A\} \twoheadrightarrow \{B, C\}, \{D\} \rightarrow \{C\}\} \models \{A\} \rightarrow \{C\}$.

A	B	C	D
a_1	b_1	c_1	d_1
a_1	b_2	c_1	d_2
a_1	b_2	c_1	d_1
a_1	b_1	c_1	d_2

Starting from a row of distinguished variables, we get a column of distinguished variables.

Example 2 (Recall)

$$\{\{A\} \twoheadrightarrow \{B\}, \{B\} \twoheadrightarrow \{C\}\} \models \{A\} \twoheadrightarrow \{C\}$$

A	B	C	D
a_1	b_1	c_1	d_1
a_1	b_2	c_2	d_2
$\cdots$			
a_1	b_1	c_2	d_1
$\cdots$			

Starting from two rows of distinguished variables (chasing the join dependency $R = \{A, B, D\} \bowtie \{A, C\}$), we get a row of distinguished variables.

Example 2 (Recall)

For $R = \{A, B, C, D\}$ with Σ is $\{\{A, B\}, \{B, C\}, \{A, D\}\}$ a lossless decomposition.

A	B	C	D
a_1	b_1	c_1	d_1
a_2	b_1	c_2	d_2
a_1	b_3	c_3	d_3

Chase a row of distinguished variables ((a_1, b_1, c_2, d_3)) from the above table.