Junyu Wang

jw6602

1.

Let’s start. Below we have a table with the current **Old** and the proposed **New**.

|  |  |
| --- | --- |
| **Old** | **New** |
| AC → E  **AC → CB**  **E → DE**  D → B | AC → E  **AC → B**  **E → D**  D → B |

**New** was obtained by removing the “defective” parts of **Old** and therefore there is nothing to do for proving equivalence. **New** now becomes **Old**.

|  |  |
| --- | --- |
| **Old** | **New** |
| AC → E  **AC → B**  E → D  D → B | AC → E  E → D  D → B |

We compute , and as it contains , we prove equivalence.

Therefore, our minimal cover is

AC → E

E → D

D → B

2.

Let’s start. Below we have a table with the current **Old** and the proposed **New**.

|  |  |
| --- | --- |
| **Old** | **New** |
| AB **→** C  A **→** E  DE → F  **E → EF**  F → B | AB **→** C  A **→** E  DE → F  **E → F**  F → B |

**New** was obtained by removing the “defective” parts of **Old** and therefore there is nothing to do for proving equivalence. **New** now becomes **Old**.

|  |  |
| --- | --- |
| **Old** | **New** |
| AB **→** C  A **→** E  **DE → F**  E → F  F → B | AB **→** C  A **→** E  E → F  F → B |

We compute , and as it contains , we prove equivalence. **New** becomes **Old**.

|  |  |
| --- | --- |
| **Old** | **New** |
| **AB → C**  A **→** E  E → F  F → B | **A → C**  A **→** E  E → F  F → B |

We compute , and as it contains , we prove equivalence. **New** becomes **Old**.

|  |  |
| --- | --- |
| **Old** | **New** |
| **A → C**  **A → E**  E → F  F → B | **A → CE**  E → F  F → B |

This is an application of the union rule, which always produces an equivalent set.

Therefore, our minimal cover is

A → CE

E → F

F → B

3.

(a)

1. Let’s try with trivial removal and union rule. We can see that both operations are unavailable.
2. Let’s try to simplify the LHS or RHS.

Attempt 1.

|  |  |
| --- | --- |
| **Old** | **New** |
| **AB → CD**  C → EF  E → C  F → G | **AB → C**  C → EF  E → C  F → G |

We compute , and as it does not contain , we prove non-equivalence.

Attempt 2.

|  |  |
| --- | --- |
| **Old** | **New** |
| **AB → CD**  C → EF  E → C  F → G | **AB → D**  C → EF  E → C  F → G |

We compute , and as it does not contain , we prove non-equivalence.

Attempt 3.

|  |  |
| --- | --- |
| **Old** | **New** |
| **AB → CD**  C → EF  E → C  F → G | **A → CD**  C → EF  E → C  F → G |

We compute , and as it does not contain , we prove non-equivalence.

Attempt 4.

|  |  |
| --- | --- |
| **Old** | **New** |
| **AB → CD**  C → EF  E → C  F → G | **B → CD**  C → EF  E → C  F → G |

We compute , and as it does not contain , we prove non-equivalence.

Attempt 5.

|  |  |
| --- | --- |
| **Old** | **New** |
| AB → CD  **C → EF**  E → C  F → G | AB → CD  **C → E**  E → C  F → G |

We compute , and as it does not contain , we prove non-equivalence.

Attempt 6.

|  |  |
| --- | --- |
| **Old** | **New** |
| AB → CD  **C → EF**  E → C  F → G | AB → CD  **C → F**  E → C  F → G |

We compute , and as it does not contain , we prove non-equivalence.

We prove that we cannot simplify the LHS or RHS. Therefore, the given set of FDs is already a minimal cover.

(b)

We classify the attributes based on where they appear in the FDs:

1. On both sides:
2. On left side only:
3. On right side only:
4. Nowhere:

appear in every key. does not appear in any key. may appear in any key. We start with , which must contain a key. We cannot remove but may try to remove .

We attempt to remove . We compute . We can remove and we continue with .

We attempt to remove . We compute . We can remove and we continue with .

We attempt to remove . We compute . We can remove and we continue with .

As nothing else can be done, is a global key. We can remove because it’s a subset of .

Therefore, our final decomposition is

ABCD

CEF

FG

ABH