

A Membership Algorithm for Functional and Multi-valued Dependencies in the Presence of Lists

Sven Hartmann, Sebastian Link

Information Science Research Centre,
Massey University, Palmerston North,
New Zealand

- 1. Motivation & Revision of the RDM**
- 2. The Brouwerian Algebra of Nested Attributes**
- 3. Axiomatisation of FDs and MVDs**
- 4. The Membership Algorithm**
- 5. Extensions**

1.1 A Pubcrawl through Dunedin

- consider **PubCrawl(Person, Visit[Drink(Beer, Pub)])**
- a typical snapshot r is

(Sven, [(Speights, Cook), (Steinlager, Bennu)]),
 (Sven, [(Steinlager, Cook), (Speights, Bennu)]),
 (Klaus-Dieter, [(London Porter, Diva), (Guinness, Perculator), (London Porter, Diva)]),
 (Klaus-Dieter, [(India Pale Ale, Diva), (Tui, Perculator), (London Porter, Diva)]),
 (Klaus-Dieter, [(London Porter, Bennu), (Guinness, Cook), (London Porter, Perculator)]),
 (Klaus-Dieter, [(India Pale Ale, Bennu), (Tui, Cook), (London Porter, Perculator)]),
 (Sebastian, [])

- $\not\models_r$ Pubcrawl(Person) \rightarrow Pubcrawl(Visit[Drink(Pub)])
- \models_r Pubcrawl(Person) \rightarrow Pubcrawl(Visit[Drink(Pub)])
- What other constraints are implied?

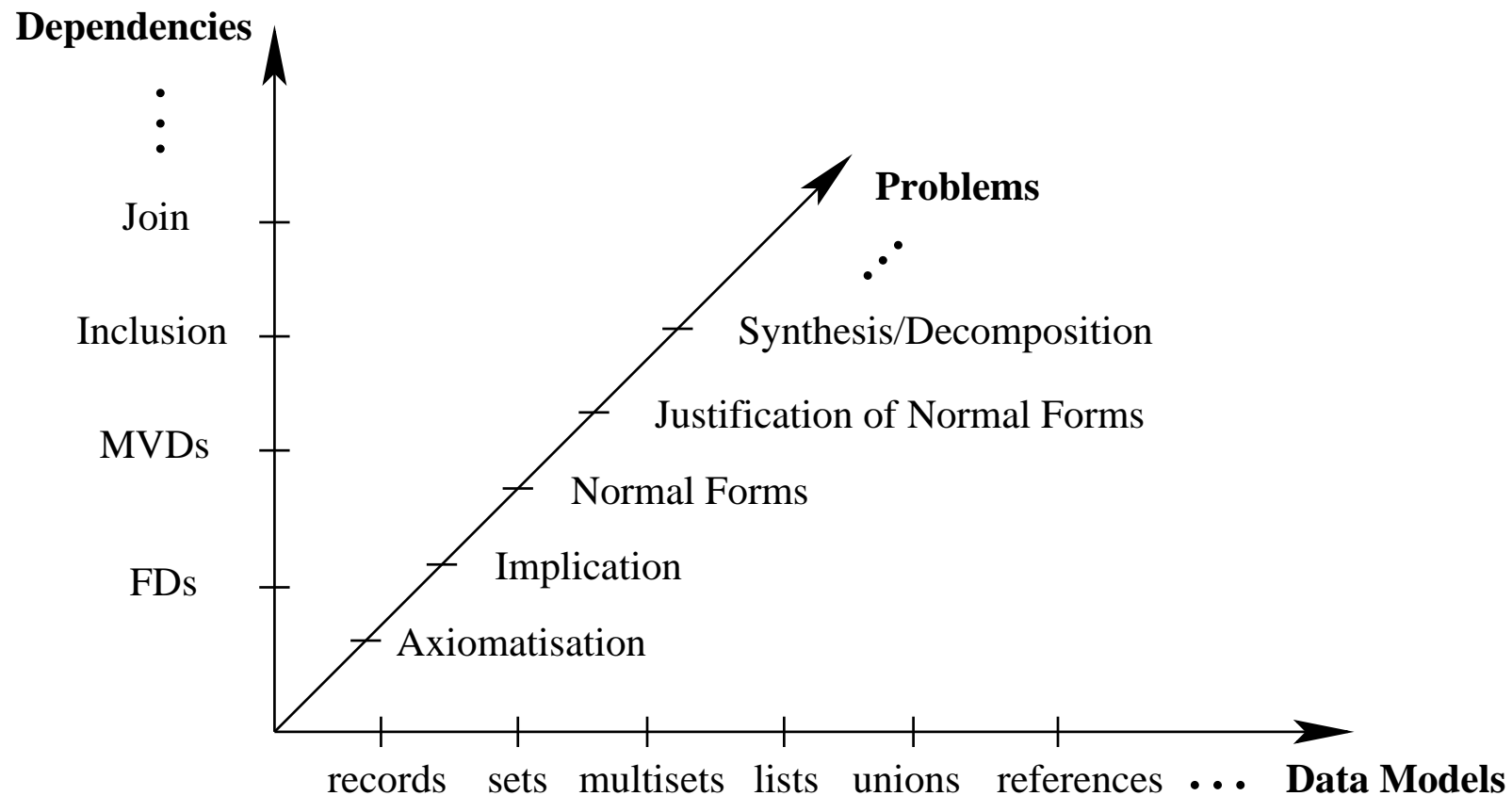
1.2 FDs and MVDs in the RDM

- FDs introduced in context of RDM by E.F. Codd in 1972
 - expression $X \rightarrow Y$ with $X, Y \subseteq R$
 - $\models_r X \rightarrow Y$ iff $t_1 \upharpoonright_Y = t_2 \upharpoonright_Y$, if $t_1 \upharpoonright_X = t_2 \upharpoonright_X$ for any $t_1, t_2 \in r$
- MVDs introduced in context of RDM by R. Fagin, C. Zaniolo in 1976
 - expression $X \twoheadrightarrow Y$ with $X, Y \subseteq R$
 - $\models_r X \twoheadrightarrow Y$ iff

$$\forall t_1, t_2 \in r. t_1 \upharpoonright_X = t_2 \upharpoonright_X \Rightarrow \exists t \in r. t \upharpoonright_{XY} = t_1 \upharpoonright_{XY}, t \upharpoonright_{X(R-Y)} = t_2 \upharpoonright_{X(R-Y)}$$
- axiomatisation for FDs, MVDs, and FDs + MVDs (1974,1977)
- important results for automated design tools:
 - **(Finite) Implication Problem** decidable in almost linear time
 - efficiently deciding equiv of sets of FDs, computing minimal covers
 - **Normal Forms** (BCNF, 3NF, 4NF)
 - no redundancies and update anomalies, simple integrity checking

1.3 Challenges with Advanced Data Models

- find unifying framework, extend achievements to complex object types
- classify data models according to the types they support



1.4 The Need for Lists

- here: **values, records of values and lists of values**
- ordered relations
- time-series data, meteorological and astronomical data streams, runs of experimental data, multidimensional arrays, textual information, voices, sound, images, video, etc.
- subject to studies in deductive and temporal database community
- occur naturally in object-oriented databases (XML)
- bioinformatics: lists occur naturally in genomic sequence databases

2.1 Syntax: Nested Attributes

- capture characteristics of objects in target database by attributes
- finite set \mathcal{U} of flat attributes and $dom(A)$ for all $A \in \mathcal{U}$
- use set \mathcal{L} of labels with $\mathcal{U} \cap \mathcal{L} = \emptyset$ and $\lambda \notin \mathcal{U} \cup \mathcal{L}$
- **nested attributes** $\mathcal{NA}(\mathcal{U}, \mathcal{L})$:
 - *flat attributes* $\mathcal{U} \subseteq \mathcal{NA}$,
 - *null attribute* $\lambda \in \mathcal{NA}$,
 - *record-valued attributes* $L(N_1, \dots, N_k) \in \mathcal{NA}$, if $L \in \mathcal{L}$ and $N_1, \dots, N_k \in \mathcal{NA}$ with $k \geq 1$
 - *list-valued attributes* $L[N] \in \mathcal{NA}$, if $L \in \mathcal{L}$ and $N \in \mathcal{NA}$

2.2 Semantics: Domain Assignment

- extend mapping dom from flat attributes to nested attributes by:
 - $dom(\lambda) = \{ok\}$,
 - $dom(L(N_1, \dots, N_k)) = \{(v_1, \dots, v_k) \mid v_i \in dom(N_i)\}$,
 - $dom(L[N]) = \{[v_1, \dots, v_n] \mid v_i \in dom(N)\}$
- empty list denoted by $[]$
- RDM: record-valued attributes only

2.3 Subattributes

- define $\leq \subseteq \mathcal{NA} \times \mathcal{NA}$ by:
 - $N \leq N$ for all nested attributes $N \in \mathcal{NA}$,
 - $\lambda \leq A$ for all flat attributes $A \in \mathcal{U}$,
 - $\lambda \leq N$ for all list-valued attributes $N \in \mathcal{NA}$,
 - $L(N_1, \dots, N_k) \leq L(M_1, \dots, M_k)$, if $N_i \leq M_i$ for all $i = 1, \dots, k$,
 - $L[N] \leq L[M]$, if $N \leq M$
- subattribute relation \leq on nested attributes is partial order

2.4 Semantics on Subattributes: Projection Function

- for $M \leq N$ define $\pi_M^N : Dom(N) \rightarrow Dom(M)$ by:
 - $\pi_N^N : v \mapsto v,$
 - $\pi_\lambda^N : v \mapsto ok,$
 - $\pi_{L(M_1, \dots, M_k)}^{L(N_1, \dots, N_k)} : (v_1, \dots, v_k) \mapsto (\pi_{M_1}^{N_1}(v_1), \dots, \pi_{M_k}^{N_k}(v_k)),$
 - $\pi_{L[M']}]^{L[N']}] : [v_1, \dots, v_n] \mapsto [\pi_{M'}^{N'}(v_1), \dots, \pi_{M'}^{N'}(v_n)]$
- $[]$ mapped to itself, except when projected on λ

2.5 Operations on Subattributes

- $Sub(N) = \{X \in \mathcal{NA} \mid X \leq N\}$: $\lambda_N, Y \sqcup_N Z, Y \sqcap_N Z, Y \dot{-}_N Z$:
 - $\lambda_N = L(\lambda_{N_1}, \dots, \lambda_{N_k})$, if $N = L(N_1, \dots, N_k)$, and $\lambda_N = \lambda$ else,
 - $X \leq Y$: $X \sqcup_N Y = Y$, $X \sqcap_N Y = X$, and $X \dot{-}_N Y = \lambda_N$,
 - $X \dot{-}_N \lambda_N = X$,
 - $N = L(N_1, \dots, N_k), X = L(X_1, \dots, X_k), Y = L(Y_1, \dots, Y_k)$:

$$X \circ_N Y = L(X_1 \circ_{N_1} Y_1, \dots, X_k \circ_{N_k} Y_k)$$
 for $\circ \in \{\sqcup, \sqcap, \dot{-}\}$
 - $N = L[M], X = L[X'], Y = L[Y'], \circ \in \{\sqcup, \sqcap\}$:

$$X \circ_N Y = L[X' \circ_M Y'], \quad X \not\leq Y : X \dot{-}_N Y = L[X' \dot{-}_M Y']$$

2.6 The Brouwerian Algebra of Subattributes

- $(Sub(N), \leq, \sqcup_N, \sqcap_N, \dot{-}_N, N)$ is a **Brouwerian Algebra**

- $(Sub(N), \leq, \sqcup_N, \sqcap_N)$ is a lattice
- N is top element
- pseudo difference $Z \dot{-} Y$ of Z and Y in $Sub(N)$ satisfies

$$Z \dot{-} Y \leq X \quad \text{if and only if} \quad Z \leq Y \sqcup X$$

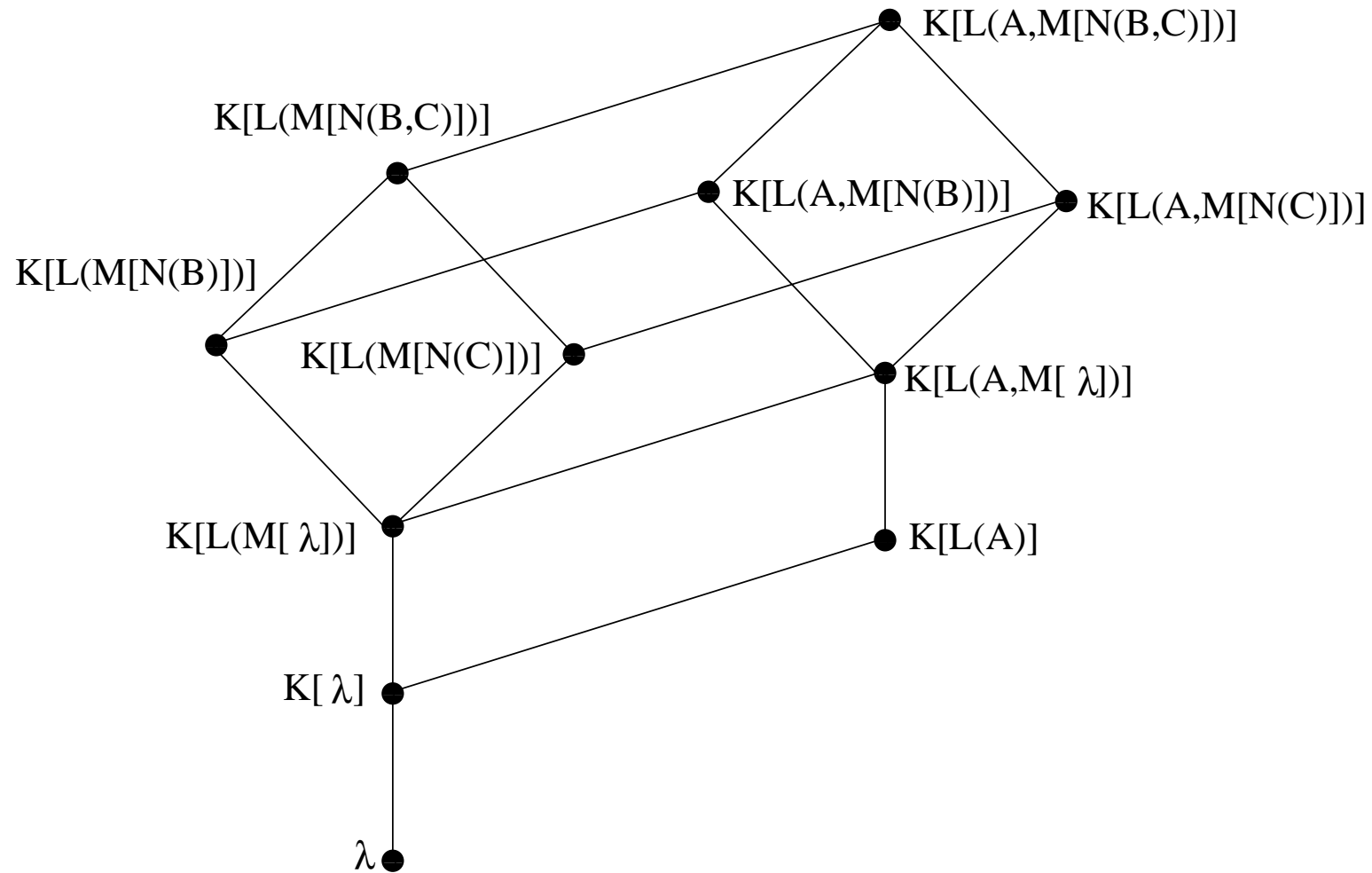
for all $X \in Sub(N)$

- **Brouwerian Complement:** $Y^{\mathcal{C}}_N = N \dot{-}_N Y$ satisfies

$$Y^{\mathcal{C}} \leq X \quad \text{if and only if} \quad X \sqcup Y = N$$

- bottom element $\lambda_N = N \dot{-} N$
- $(Sub(N), \leq, \sqcup_N, \sqcap_N, (\cdot)^{\mathcal{C}}_N, \lambda_N, N)$ is not a **Boolean Algebra**
- every Brouwerian Algebra is distributive

2.7 The Algebra of Nested Attributes: An Example



3.1 FDs and MVDs

- **FD** on N : $X \rightarrow Y$ with $X, Y \in Sub(N)$
- $\models_r X \rightarrow Y$ iff $\pi_X^N(t_1) = \pi_X^N(t_2)$ implies $\pi_Y^N(t_1) = \pi_Y^N(t_2)$
- **MVD** on N : $X \twoheadrightarrow Y$ with $X, Y \in Sub(N)$
- $\models_r X \twoheadrightarrow Y$ iff $\forall t_1, t_2 \in r$ with $\pi_X^N(t_1) = \pi_X^N(t_2) \exists t \in r$ with $\pi_{X \sqcup Y}^N(t) = \pi_{X \sqcup Y}^N(t_1)$ and $\pi_{X \sqcup Y}^N c(t) = \pi_{X \sqcup Y}^N c(t_2)$
- implication: $\Sigma \models \tau$ iff $\models_r \tau$ if $\models_r \sigma$ for all $\sigma \in \Sigma$ and any (finite) r
- **semantic hull**: $\Sigma^* = \{\sigma \mid \Sigma \models \sigma\}$
- **syntactic hull**: $\Sigma^+ = \{\sigma \mid \Sigma \vdash_{\mathfrak{R}} \sigma\}$ for set \mathfrak{R} of inference rules
- goal: find **sound** and **complete** \mathfrak{R} , i.e., $\Sigma^+ \subseteq \Sigma^*$ and $\Sigma^* \subseteq \Sigma^+$

3.2 Lossless Decomposition

- $\models_r X \twoheadrightarrow Y$ iff $r = \pi_{X \sqcup Y}(r) \bowtie \pi_{X \sqcup Y}c(r)$
- $\models_r \text{Pubcrawl}(\text{Person}) \twoheadrightarrow \text{Pubcrawl}(\text{Visit}[\text{Drink}(\text{Pub})])$ suggests

(Sven, [(ok, Cook), (ok, Bennu)]),
 (Klaus-Dieter, [(ok, Diva), (ok, Perculator), (ok, Diva)]),
 (Klaus-Dieter, [(ok, Bennu), (ok, Cook), (ok, Perculator)]),
 (Sebastian, [])

(Sven, [(Speights, ok), (Steinlager, ok)]),
 (Sven, [(Steinlager, ok), (Speights, ok)]),
 (Klaus-Dieter, [(London Porter, ok), (Guinness, ok), (London Porter, ok)]),
 (Klaus-Dieter, [(India Pale Ale, ok), (Tui, ok), (London Porter, ok)]),
 (Sebastian, [])

3.3 Axiomatisation for FDs and MVDs

$$\frac{}{X \rightarrow Y} \quad Y \leq X$$

$$\frac{X \rightarrow Y}{X \rightarrow X \sqcup Y}$$

$$\frac{X \rightarrow Y, Y \rightarrow Z}{X \rightarrow Z}$$

$$\frac{X \twoheadrightarrow Y}{X \rightarrow Y^c}$$

$$\frac{X \twoheadrightarrow Y \quad Y \twoheadrightarrow Z}{X \twoheadrightarrow Z \dot{-} Y}$$

$$\frac{X \twoheadrightarrow Y \quad X \twoheadrightarrow Z}{X \twoheadrightarrow Y \sqcup Z}$$

$$\frac{X \rightarrow Y}{X \twoheadrightarrow Y}$$

$$\frac{X \twoheadrightarrow Y \quad Y \rightarrow Z}{X \rightarrow Z \dot{-} Y}$$

$$\frac{X \twoheadrightarrow Y}{X \rightarrow Y \sqcap Y^c}$$

$$\frac{X \twoheadrightarrow Y}{W \sqcup X \twoheadrightarrow V \sqcup Y} \quad V \leq W$$

$$\frac{X \twoheadrightarrow Y \quad X \twoheadrightarrow Z}{X \twoheadrightarrow Z \dot{-} Y}$$

$$\frac{X \twoheadrightarrow Y \quad X \twoheadrightarrow Z}{X \twoheadrightarrow Y \sqcap Z}$$

3.4 Basis Attributes and Generalised Subsets

- **subattribute basis** of N is smallest subset $SubB(N) \subseteq Sub(N)$ such that every $X \in Sub(N)$ satisfies $X = \sqcup \mathcal{Z}$ for some $\mathcal{Z} \subseteq SubB(N)$ (set of join-irreducible elements)
- $MaxB(N)$ denotes maximal elements in $(SubB(N), \leq)$
- subattribute basis of **Pubcrawl(Person, Visit[Drink(Beer, Pub)])**:

Pubcrawl(Person, λ), Pubcrawl(λ , Visit[Drink(Beer, λ)]),

Pubcrawl(λ , Visit[Drink(λ , Pub)]) and Pubcrawl(λ , Visit[Drink(λ , λ)])

- $\mathcal{X} \subseteq_{\text{gen}} \mathcal{Y}$ if and only if $\forall X \in \mathcal{X}. \exists Y \in \mathcal{Y}$ with $X \leq Y$

3.5 Dependency Basis

- $X \in Sub(N)$, Σ set of FDs and MVDs on N :

- $Dep(X) = \{Y \leq N \mid X \twoheadrightarrow Y \in \Sigma^+\}$
- $X^+ = \sqcup \{Y \mid X \rightarrow Y \in \Sigma^+\}$

consider $X^M \subseteq Sub(N)$ with:

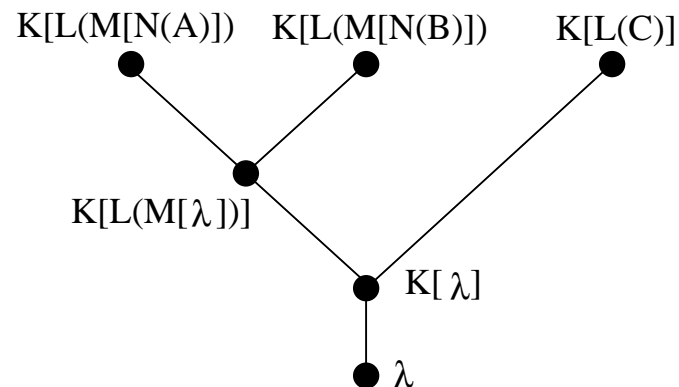
- for all $U \in MaxB(N)$ there is a unique $V \in X^M$ with $U \leq V$
 - for all $U \in X^M$ there is some $W \subseteq MaxB(N)$ with $U = \sqcup W$
 - for all $V \in Dep(X)$ there is some $Z \subseteq X^M$ with $V^{CC} = \sqcup Z$
 - X^M is \subseteq_{gen} -maximal with these properties
- $DepB(X) = SubB(X^+) \cup X^M$ is called **dependency basis**
 - important facts:
 - $X \twoheadrightarrow Y \in \Sigma^+$ if and only if $Y = \sqcup Z$ for some $Z \subseteq DepB(X)$
 - $X \rightarrow Y \in \Sigma^+$ if and only if $Y \leq X^+$

3.6 Examples - Dependency Basis

- $\text{Pubcrawl}(\text{Person}, \text{Visit}[\text{Drink}(\text{Beer}, \text{Pub})])$ with
- $\text{Pubcrawl}(\text{Person}) \twoheadrightarrow \text{Pubcrawl}(\text{Visit}[\text{Drink}(\text{Beer})])$ yields
- X^M contains $\text{Pubcrawl}(\text{Person})$, $\text{Pubcrawl}(\text{Visit}[\text{Drink}(\text{Beer})])$ and $\text{Pubcrawl}(\text{Visit}[\text{Drink}(\text{Pub})])$
- $\text{SubB}(X^+) = \{\text{Pubcrawl}(\text{Visit}[\lambda])\}$ where $X = \text{Pubcrawl}(\text{Person})$
- this gives:
 - $\text{Pubcrawl}(\text{Person}) \twoheadrightarrow \text{Pubcrawl}(\text{Visit}[\text{Drink}(\text{Pub})])$ is implied
 - $\text{Pubcrawl}(\text{Person}) \twoheadrightarrow \text{Pubcrawl}(\text{Visit}[\text{Drink}(\text{Beer}, \text{Pub})])$ is implied
 - $\text{Pubcrawl}(\text{Person}) \rightarrow \text{Pubcrawl}(\text{Visit}[\text{Drink}(\text{Pub})])$ is not implied
 - $\text{Pubcrawl}(\text{Person}) \rightarrow \text{Pubcrawl}(\text{Visit}[\lambda])$ is implied

3.7 Possessed Attributes

- $\mathcal{X} \subseteq \text{Max}B(N)$, $X = \sqcup \mathcal{X}$: $Y \in \text{Sub}B(X)$ is **possessed** by X if and only if every $Z \in \text{Sub}B(N)$ with $Y \leq Z$ satisfies $Z \leq X$
- consider $K[L(M[N(A, B)], C)]$ and $X = K[L(M[N(A, B)], \lambda)]$:
 - X possesses $K[L(M[N(\lambda, \lambda)], \lambda)]$, but does not possess $K[L(\lambda, \lambda)]$



- basis attributes not possessed by any element in X^M are functionally determined by X

Algorithm 0.1 (Attribute Set Closure and Dependency Basis)**Input:** $N \in \mathbf{NA}$, $X \in \mathbf{Sub}(N)$, set Σ of FDs and MVDs on N **Output:** X_{alg}^+ and $\text{DepB}_{\text{alg}}(X)$ **Method:**VAR $DB_{\text{new}}, DB_{\text{old}} \subseteq \mathbf{Sub}(N)$, $X_{\text{new}}, X_{\text{old}}, W, \bar{U}, \tilde{V}, U' \in \mathbf{Sub}(N)$; $X_{\text{new}} := X$; $DB_{\text{new}} := \text{MaxB}(X^{cc}) \cup \{X^c\}$;

REPEAT

 $X_{\text{old}} := X_{\text{new}}$; $DB_{\text{old}} := DB_{\text{new}}$; FOR each $U \rightarrow V \in \Sigma$ DO $\bar{U} := \sqcup \{W \in DB_{\text{new}} \mid \exists U'. U' \text{ is possessed by } W, U' \not\leq X_{\text{new}}, U' \leq U\}$; $\tilde{V} := V \dot{-} \bar{U}$; IF $\tilde{V} \neq \lambda$ THEN BEGIN $X_{\text{new}} := X_{\text{new}} \sqcup \tilde{V}$; $DB_{\text{new}} := \{(W \dot{-} \tilde{V})^{cc} \mid W \in DB_{\text{new}}, (W \dot{-} \tilde{V})^{cc} \neq \lambda\} \cup \text{MaxB}(\tilde{V}^{cc})$;

END;

ENDDO;

FOR each $U \rightarrow V \in \Sigma$ DO
 $\bar{U} := \sqcup \{W \in DB_{\text{new}} \mid \exists U'. U' \text{ is possessed by } W, U' \not\leq X_{\text{new}}, U' \leq U\};$
 $\tilde{V} := V \dot{-} \bar{U};$
 IF $\tilde{V} \neq \lambda$ THEN BEGIN
 $X_{\text{new}} := X_{\text{new}} \sqcup (\tilde{V} \sqcap \tilde{V}^c);$
 FOR each $W \in DB_{\text{new}}$ DO
 IF $(\tilde{V} \sqcap W)^{cc} \neq \lambda$ AND $(\tilde{V} \sqcap W)^{cc} \neq W$ THEN
 $DB_{\text{new}} := (DB_{\text{new}} - \{W\}) \cup \{(\tilde{V} \sqcap W)^{cc}, (W \dot{-} \tilde{V})^{cc}\};$
 ENDDO;
 END;
 ENDDO;
 UNTIL $(X_{\text{new}} = X_{\text{old}})$ AND $(DB_{\text{new}} = DB_{\text{old}});$
 $X_{\text{alg}}^+ := X_{\text{new}};$
 $DepB_{\text{alg}}(X) := SubB(X_{\text{alg}}^+) \cup DB_{\text{new}};$
 RETURN($X_{\text{alg}}^+, DepB_{\text{alg}}(X)$);

□

4.2 Results

- **Theorem:**

The Algorithm works correctly, i.e., $X_{\text{alg}}^+ = X^+$ and $DepB_{\text{alg}}(X) = DepB(X)$.

- $\Sigma_{\text{alg}}^+ \subseteq \Sigma^+$:

- $X \twoheadrightarrow W_j \in \Sigma^+$ for all $W_j \in DepB_{\text{alg}}(X)$ and

- $X \rightarrow X_{\text{alg}}^+ \in \Sigma^+$

- $\Sigma^+ \subseteq \Sigma_{\text{alg}}^+$:

consider each inference rule in turn to justify that it is sufficient to consider FDs and MVDs in Σ

- **Theorem:**

The implication problem is decidable in time $\mathcal{O}(|SubB(N)|^4 \cdot |\Sigma|)$.

5.1 Extensions: More Results

- **Record and List Type:**
 - simple axiomatisation for FDs
 - FD implication decidable in linear time
 - NLNF (weaker than BCNF) proposed and justified
 - MVDs: minimal axiomatisation and finite implication problem
- **Record, List, Set and Multiset Type:**
 - sophisticated axiomatisation for FDs
 - FD implication decidable in $\mathcal{O}(n^3 \cdot s \cdot \min\{s, n\})$
 - CVNF proposed and justified
- **XML**: several classes of FDs, Axiomatisations

5.2 Extensions: Future Research

- normal form for FDs and MVDs
- decomposition/synthesis for NLNF, CVNF
- unions, references
- interactions
- different classes of dependencies: inclusion and join dependencies
- XML: FIP and Normalisation