An Equivalence between Dependencies in Nested Databases and a Fragment of Propositional Logic

Sven Hartmann, Sebastian Link

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

1. Examples of Data Dependencies and Objective
2. Nested Databases generated by Record and List Constructor
3. Functional and Multivalued Dependencies
4. The Equivalence
5. Extension to Set and Multiset Constructor

1.1 Sightseeing in Brazil

- database schema **Travel(Tourist,Tour[Venue(Sight,Souvenir)])**

- a typical snapshot r is

  (Jack, [ ])
  (Hans, [(Sugar Loaf, Wallpaper), (Corcovado, Magnet)]),
  (Hans, [(Iguassu, Wallpaper), (Tijuca, Magnet)]),
  (Musashi, [(Tijuca, Postcard), (Casa das Rosas, Postcard),
  (Corcovado, Magnet)])
  (Musashi, [(Corcovado, Postcard), (Sugar Loaf, Postcard),
  (Canasvieiras, Magnet)])

- $\models_r$ Travel(Tourist) $\rightarrow$ Travel(Tour[Venue(Souvenir)])

1.2 More Sightseeing in Brazil

- same schema **Travel(Tourist,Tour[Venue(Sight,Souvenir)])**
- an extended snapshot r:

(Jack, [ ])
(Hans, [(Sugar Loaf, Wallpaper), (Corcovado, Magnet)]),
(Hans, [(Iguassu, Wallpaper), (Tijuca, Magnet)]),
(Musashi, [(Tijuca, Postcard), (Casa das Rosas, Postcard), (Corcovado, Magnet)])
(Musashi, [(Corcovado, Postcard), (Sugar Loaf, Postcard), (Canasvieiras, Magnet)])
(Musashi, [(Tijuca, DVD), (Casa des Rosas, DVD), (Corcovado, Wallpaper)])
(Musashi, [(Corcovado, DVD), (Sugar Loaf, DVD), (Canasvieiras, Wallpaper)])

- $\models_r$ Travel(Tourist) $\twoheadrightarrow$ Travel(Tour[Venue(Souvenir)])

1.3 The Role of Data Dependencies

- dependencies specified on the database schema (time-invariant) (by database designer)
- they constrain the possible database instances (varying over time) to those which are considered meaningful
- crucial for data modeling capabilities
- achieve consistency and allow to assess quality of database
- fundamental is the implication problem:
  What other data dependencies are implied by those specified?
- objective here:
  describe the semantic implication of FDs and MVDs in logical terms

2.1 Database Schemata: Nested Attributes

- capture characteristics of objects in target database syntactically
- 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** $\mathcal{N}\mathrm{A}(\mathcal{U}, \mathcal{L})$:
  - *flat attributes* $\mathcal{U} \subseteq \mathcal{N}\mathrm{A}$,
  - *null attribute* $\lambda \in \mathcal{N}\mathrm{A}$,
  - *record-valued attributes* $L(N_1, \dots, N_k) \in \mathcal{N}\mathrm{A}$, if $L \in \mathcal{L}$ and $N_1, \dots, N_k \in \mathcal{N}\mathrm{A}$ with $k \geq 1$
  - *list-valued attributes* $L[N] \in \mathcal{N}\mathrm{A}$, if $L \in \mathcal{L}$ and $N \in \mathcal{N}\mathrm{A}$
- example: Travel(Tourist,Tour[Venue(Sight,Souvenir)])

2.2 Nested Databases: 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 [ ]
- relational database instance:
  finite set of k-tuples, i.e., instance over $R(A_1, \dots, A_k)$

2.3 Subschemata

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

2.4 Database Transformations: Projection Function

- subattributes represent at most as much info as their superattributes
- for $M \leq N$ define $\pi^N_M : dom(N) \to dom(M)$ by:
  - $\pi^N_N : v \mapsto v$,
  - $\pi^N_\lambda : v \mapsto ok$,
  - $\pi^{L(N_1,\dots,N_k)}_{L(M_1,\dots,M_k)} : (v_1, \dots, v_k) \mapsto (\pi^{N_1}_{M_1}(v_1), \dots, \pi^{N_k}_{M_k}(v_k))$,
  - $\pi^{L[N']}_{L[M']} : [v_1, \dots, v_n] \mapsto [\pi^{N'}_{M'}(v_1), \dots, \pi^{N'}_{M'}(v_n)]$
- [ ] mapped to itself, except when projected on λ

2.5 The Brouwerian Algebra of Subattributes

- $\leq$ induces operations of join $\sqcup_N$, meet $\sqcap_N$, and pseudo-difference $\dot{-}_N$
- $(Sub(N), \leq, \sqcup_N, \sqcap_N, \dot{-}_N, N)$ is a **Brouwerian Algebra**
  - $(Sub(N) = \{X \in \mathcal{N}A \mid X \leq N\}, \leq, \sqcup_N, \sqcap_N)$ is a lattice
  - N is top element
  - pseudo-difference $Z \dot{-} Y$ of Z and Y in $Sub(N)$ satisfies

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

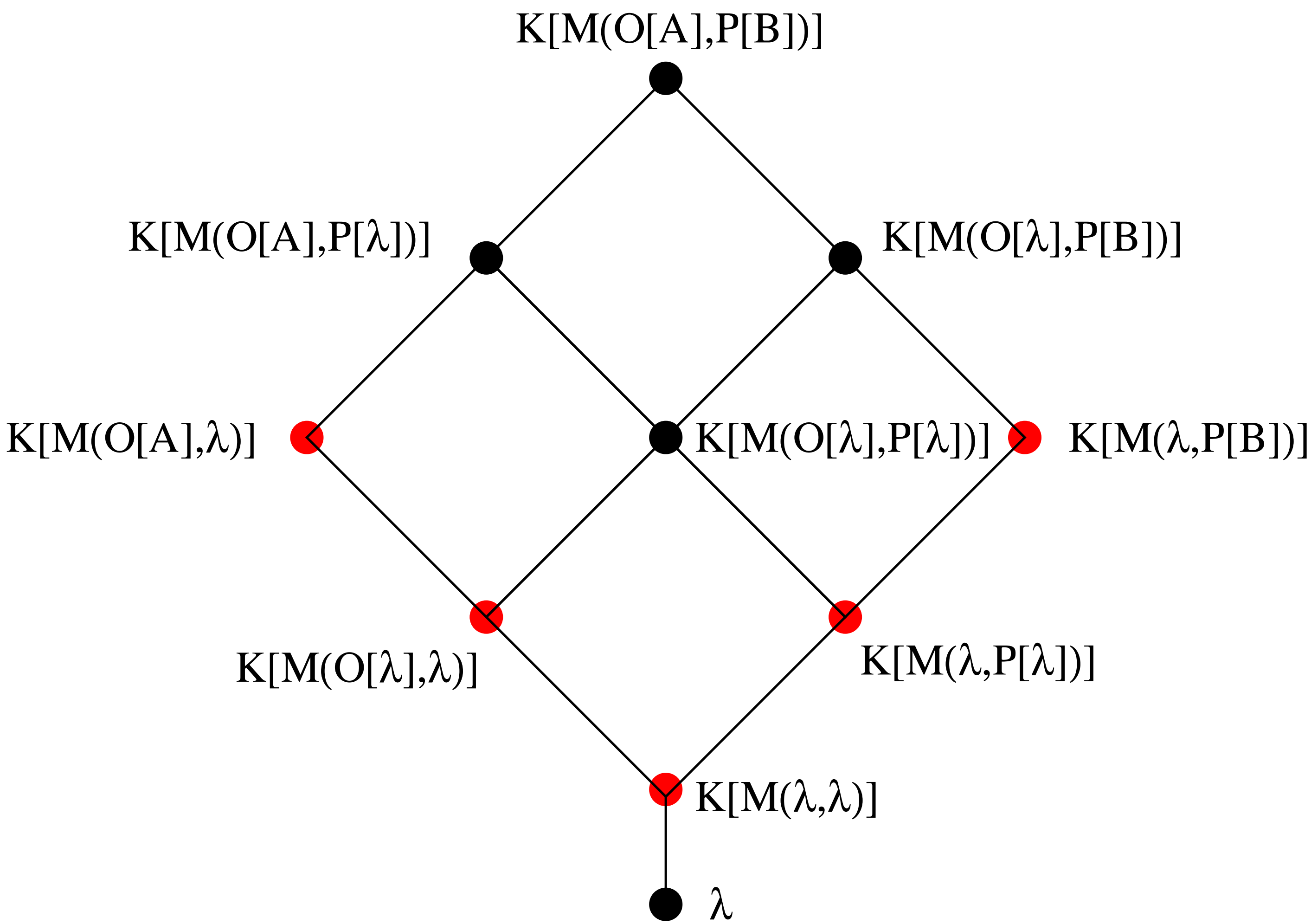
  for all $X \in Sub(N)$

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

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

- $(Sub(N), \leq, \sqcup_N, \sqcap_N, (\cdot)_N^{\mathcal{C}}, \lambda_N, N)$ is not a **Boolean Algebra**

2.6 The Algebra of Nested Attributes: An Example



3.1 Data Dependencies between Nested Data Elements

- **FD** on N is: $X \rightarrow Y$ with $X, Y \in Sub(N)$
- $r \subseteq dom(N)$ **satisfies** $X \rightarrow Y$ on N ($\models_r X \rightarrow Y$) iff

$$\forall t_1, t_2 \in r\text{: if } \pi^N_X(t_1) = \pi^N_X(t_2), \text{ then } \pi^N_Y(t_1) = \pi^N_Y(t_2)$$

- example on Travel(Tourist,Tour[Venue(Sight,Souvenir)]):
  - Travel(Tourist,λ) $\rightarrow$ Travel(λ,Tour[Venue(λ,Souvenir)])
- **MVD** on N is: $X \twoheadrightarrow Y$ with $X, Y \in Sub(N)$
- $r \subseteq dom(N)$ **satisfies** $X \twoheadrightarrow Y$ on N ($\models_r X \twoheadrightarrow Y$) iff

$$\forall t_1, t_2 \in r \text{ with } \pi^N_X(t_1) = \pi^N_X(t_2)$$

$$\exists t \in r \text{ with } \pi^N_{X \sqcup Y}(t) = \pi^N_{X \sqcup Y}(t_1) \text{ and } \pi^N_{X \sqcup Y^C}(t) = \pi^N_{X \sqcup Y^C}(t_2)$$

- example on Travel(Tourist,Tour[Venue(Sight,Souvenir)]):
  - Travel(Tourist,λ) $\twoheadrightarrow$ Travel(λ,Tour[Venue(λ,Souvenir)])

Massey University

3.2 Semantic Implication

- (finite) implication:
  $\Sigma \models_{(f)} \tau$ iff $\models_r \tau$ if $\models_r \sigma$ for all $\sigma \in \Sigma$ and any (finite) r
- implication in two-element instances:
  $\Sigma \models_2 \tau$ iff $\models_r \tau$ if $\models_r \sigma$ for all $\sigma \in \Sigma$ and any $r = \{t_1, t_2\}$
- finite and unrestricted implication coincide for FDs and MVDs
- trivial FDs: $\models_r X \twoheadrightarrow Y$ for all $r \subseteq dom(N)$ iff $Y \leq X$
- trivial MVDs:
  $\models_r X \twoheadrightarrow Y$ for all $r \subseteq dom(N)$ iff $Y \leq X$ or $X \sqcup Y = N$
- $\models_r X \twoheadrightarrow Y$ iff $r = \pi_{X \sqcup Y}(r) \bowtie \pi_{X \sqcup Y^C}(r)$

3.3 Lossless Decomposition

$\models_r$ Travel(Tourist) $\twoheadrightarrow$ Travel(Tour[Venue(Souvenir)]) suggests:

(Jack, [ ])
(Hans, [(ok, Wallpaper), (ok, Magnet)]),
(Musashi, [(ok, Postcard), (ok, Postcard), (ok, Magnet)]),
(Musashi, [(ok, DVD), (ok, DVD), (ok, Wallpaper)])

(Jack, [ ])
(Hans, [(Sugar Loaf, ok), (Corcovado, ok)]),
(Hans, [(Iguassu, ok), (Tijuca, ok)]),
(Musashi, [(Tijuca, ok), (Casa das Rosas, ok), (Corcovado, ok)]),
(Musashi, [(Corcovado, ok), (Sugar Loaf, ok), (Canasvieiras, ok)])

4.1 Join-Irreducible Subattributes

- an element $a \in L$ of a lattice $(L, \sqsubseteq, \sqcup, \sqcap, 0)$ with bottom element 0 is called join-irreducible iff $a \neq 0$ and if $a = b \sqcup c$ holds for any $b, c \in L$, then $a = b$ or $a = c$

- let $\mathcal{B}(N)$ denote the join-irreducibles of $(Sub(N), \leq, \sqcup, \sqcap, \lambda_N)$

- $\mathcal{B}$(Travel(Tourist,Tour[Venue(Sight,Souvenir)])) consists of:

  Travel(Tourist,λ), Travel(λ,Tour[Venue(λ,λ)]),
  Travel(λ,Tour[Venue(Sight,λ)]), Travel(λ,Tour[Venue(λ,Souvenir)])

- for lists it suffices to consider join-irreducibles:

  if $\pi_X^N(t_1) = \pi_X^N(t_2)$ and $\pi_Y^N(t_1) = \pi_Y^N(t_2)$, then $\pi_{X \sqcup Y}^N(t_1) = \pi_{X \sqcup Y}^N(t_2)$

4.2 Data Dependencies and Propositional Formulae

- $X \in Sub(N)$: $\vartheta(X) = \max_{\leq}\{W \mid W \in \mathcal{B}(N) \text{ and } W \leq X\}$
- $\psi : \mathcal{B}(N) \rightarrow \mathcal{V}$ bijection onto propositional variables
- let σ: $X \rightarrow Y$ be an FD on N where $\vartheta(X) = \{X_1, \dots, X_n\}$ and $\vartheta(Y) = \{Y_1, \dots, Y_k\}$
- $\Phi(\sigma)$ denotes propositional formula

$$\psi(X_1) \wedge \cdots \wedge \psi(X_n) \Rightarrow \psi(Y_1) \wedge \cdots \wedge \psi(Y_k)$$

- let σ: $X \twoheadrightarrow Y$ be an MVD on N where $\vartheta(X) = \{X_1, \dots, X_n\}$, $\vartheta(Y) = \{Y_1, \dots, Y_k\}$ and $\vartheta(Y_N^{\mathcal{C}} \dot{-} X) = \{Z_1, \dots, Z_m\}$
- $\Phi(\sigma)$ denotes the propositional formula

$$\psi(X_1) \wedge \cdots \wedge \psi(X_n) \Rightarrow (\psi(Y_1) \wedge \cdots \wedge \psi(Y_k)) \vee (\psi(Z_1) \wedge \cdots \wedge \psi(Z_m))$$

4.3 Example

- bijection ψ:

$$\text{Travel(Tourist,}\lambda) \leftrightarrow V_1,$$
$$\text{Travel(}\lambda\text{,Tour[Venue(Sight,}\lambda)]) \leftrightarrow V_2\ ,$$
$$\text{Travel(}\lambda\text{,Tour[Venue(}\lambda\text{,Souvenir)]) } \leftrightarrow V_3,$$
$$\text{Travel(}\lambda\text{,Tour[Venue(}\lambda,\lambda)]) \leftrightarrow V_4$$

- Travel(Tourist,λ) $\rightarrow$ Travel(λ,Tour[Venue(λ,λ)]) becomes:

$$V_1 \Rightarrow V_4$$

- Travel(Tourist,λ) $\twoheadrightarrow$ Travel(λ,Tour[Venue(Sight,λ)]) becomes:

$$V_1 \Rightarrow V_2 \vee V_3$$

4.4 The Equivalence Theorem

- Σ set of FDs and MVDs on N: let $\Pi = \{\Phi(\sigma) \mid \sigma \in \Sigma\}$
- encode the structure of $\mathcal{B}(N)$ by Horn clauses:

$$\Pi_N = \{\psi(U) \Rightarrow \psi(V) \mid U, V \in \mathcal{B}(N), U \text{ covers } V\}$$

Theorem 1. *Let N be a nested attribute, Σ a set of FDs and MVDs and σ a single FD or MVD on N. Then*

1) Σ implies σ,
2) Σ implies σ in the world of two-element instances, and
3) $\Pi \cup \Pi_N$ logically implies $\Phi(\sigma)$

are equivalent.

- this extends a well-known result by *Fagin et al.(1981)*, where
  - only single application of record constructor allowed,
  - join-irreducibles form anti-chain,
  - complement operation is involutional

4.5 Outline of the Proof

- 2) implies 1): by contraposition and the following Lemma

  Lemma 2. *Assume $r \subseteq dom(N)$ is some finite instance over N, Σ a set of FDs and MVDs on N, and σ a single FD or MVD on N. Suppose r satisfies all FDs and MVDs in Σ, but does not satisfy σ. Then there is some $r' = \{t_1, t_2\} \subseteq r$ such that r' satisfies all FDs and MVDs in Σ, but does not satisfy σ.* □

- 2) is equivalent to 3):
  idea is to define $\theta(V) = \mathit{true}$ iff two tuples have the same projection on join-irreducible $\psi^{-1}(V)$

  Lemma 3. *Let σ be an FD or MVD on the nested attribute N, and $r = \{t_1, t_2\} \subseteq dom(N)$. Then $\models_r \sigma$ if and only if $\models_{\theta_r} \Phi(\sigma)$ where*

$$\theta_r(V) = \begin{cases} \mathit{true}, & \text{if } \pi^N_{\psi^{-1}(V)}(t_1) = \pi^N_{\psi^{-1}(V)}(t_2) \\ \mathit{false}, & \text{else} \end{cases}$$

  for all $V \in \psi(\mathcal{B}(N))$. □

4.6 An Example - Reasoning with Dependencies

- Does σ: $\underbrace{\text{Travel(Tourist,}\lambda\text{)}}_{=X} \twoheadrightarrow \underbrace{\text{Travel(}\lambda\text{,Tour[Venue(Sight,}\lambda\text{)])}}_{=Y}$ imply
  - σ_1: Travel(Tourist,λ) $\rightarrow$ Travel(λ,Tour[Venue(Sight,λ)])
  - σ_2: Travel(Tourist,λ) $\rightarrow$ $\underbrace{\text{Travel(}\lambda\text{,Tour[Venue(}\lambda\text{,}\lambda\text{)])}}_{=Z}$?
- consider two-element instance:

  (Musashi, [(Tijuca, DVD), (Casa des Rosas, DVD), (Corcovado, Wallpaper)])
  (Musashi, [(Corcovado, DVD), (Sugar Loaf, DVD), (Canasvieiras, Wallpaper)])

- two-element instance shows that σ does not imply σ_1
- σ does imply σ_2: any r with $\models_r \sigma$ and any $t_1, t_2 \in r$
  $\pi_X^N(t_1) = \pi_X^N(t_2)$ implies $\exists t \in r.\pi_{X \sqcup Y}^N(t) = \pi_{X \sqcup Y}^N(t_1)$ and $\pi_{X \sqcup Y^{\mathcal{C}}}^N(t) = \pi_{X \sqcup Y^{\mathcal{C}}}^N(t_2)$
  consequently $\pi_Z^N(t_1) \overset{Z \leq Y}{\equiv} \pi_Z^N(t) \overset{Z \leq Y^{\mathcal{C}}}{\equiv} \pi_Z^N(t_2)$

4.7 An Example - Reasoning with Propositional Formulae

- Do $\Phi(\sigma)$: $V_1 \Rightarrow V_2 \vee V_3$ and Π_N formulae $V_2 \Rightarrow V_4$ and $V_3 \Rightarrow V_4$
  - imply $\Phi(\sigma_1) : V_1 \Rightarrow V_2$? or
  - imply $\Phi(\sigma_2) : V_1 \Rightarrow V_4$?
- $\Phi(\sigma_1)$ not implied witnessed by θ with $\theta(V_i) = \mathit{true}$ iff $i \in \{1, 3, 4\}$
  - satisfies $V_1 \Rightarrow V_2 \vee V_3$ and $V_2 \Rightarrow V_4$ and $V_3 \Rightarrow V_4$
  - violates $V_1 \Rightarrow V_2$
- two elements coincide on projections to precisely those join-irreducibles whose corresponding propositional variables are assigned *true* by θ
- $\Phi(\sigma_2)$ is implied:
  - consider any θ with $\theta(V_1) = \mathit{true}$
  - then $\theta(V_2) = \mathit{true}$ or $\theta(V_3) = \mathit{true}$ by $V_1 \Rightarrow V_2 \vee V_3$
  - then $\theta(V_4) = \mathit{true}$ by $V_2 \Rightarrow V_4$ and $V_3 \Rightarrow V_4$

5.1 Extension for FDs to Set and Multiset Constructor

- join-irreducibles are no longer sufficient
- Dance(Day,Class{Duo(Female,Male)},Rating) with FD

$$\text{Dance}(\lambda,\text{Class}\{\text{Duo}(\text{Female},\text{Male})\},\lambda) \rightarrow \text{Dance}(\lambda,\lambda,\text{Rating})$$

- $X, Y \in Sub(N)$ are *reconcilable* iff one of the following is satisfied

(i) $Y \leq X$ or $X \leq Y$,

(ii) $N = L(N_1, \dots, N_k), X = L(X_1, \dots, X_k), Y = L(Y_1, \dots, Y_k)$ where X_i and Y_i are reconcilable for all $i = 1, \dots, k$ or

(iii) $N = L[N'], X = L[X'], Y = L[Y']$ with reconcilable X' and Y'

- *extended join-irreducibles* form smallest $\mathcal{E}(N) \subseteq Sub(N)$ such that

(i) $\mathcal{B}(N) \subseteq \mathcal{E}(N)$, and

(ii) for all $X, Y \in \mathcal{E}(N)$ which are not reconcilable also $X \sqcup Y \in \mathcal{E}(N)$

- FDs are $\mathcal{X} \rightarrow \mathcal{Y}$ with $\leq$-antichains $\mathcal{X}, \mathcal{Y} \subseteq \mathcal{E}(N)$

5.2 An Equivalence for FDs and Horn clauses

- interpret extended join-irreducibles as variables via $\psi : \mathcal{E}(N) \to \mathcal{V}$
- FDs $\{X_1, \ldots, X_n\} \to \{Y_1, \ldots, Y_k\}$ correspond to k Horn clauses

$$\varPhi(\sigma) = \{\bigwedge_{i=1}^{n} \psi(X_i) \Rightarrow \psi(Y_1), \ldots, \bigwedge_{i=1}^{n} \psi(X_i) \Rightarrow \psi(Y_k)\}$$

Theorem 4. *Let N be a nested attribute, Σ a set of FDs and σ a single FD on N. Let $\varPi_N$ denote the Horn clauses which encode the structure of $\mathcal{B}(N)$, and $\varPi$ denote the corresponding set of Horn clauses for Σ. Then*

(i) Σ implies σ,
(ii) Σ implies σ in the world of two-tuple instances, and
(iii) $\varPi \cup \varPi_N$ logically implies π for all $\pi \in \varPhi(\sigma)$

are equivalent.

6.1 Axiomatisation for FDs and MVDs

$$\frac{}{X \rightarrow Y}\; Y \leq X \qquad \frac{X \rightarrow Y}{X \rightarrow X \sqcup Y} \qquad \frac{X \rightarrow Y,\; Y \rightarrow Z}{X \rightarrow Z}$$

$$\frac{}{\lambda \twoheadrightarrow N} \qquad \frac{X \twoheadrightarrow Y,\; Y \twoheadrightarrow Z}{X \twoheadrightarrow Z \dot{-} Y} \qquad \frac{X \twoheadrightarrow Y,\; X \twoheadrightarrow Z}{X \twoheadrightarrow Y \sqcup Z}$$

$$\frac{X \rightarrow Y}{X \twoheadrightarrow Y} \qquad \frac{X \twoheadrightarrow Y,\; Y \rightarrow Z}{X \rightarrow Z \dot{-} Y} \qquad \frac{X \twoheadrightarrow Y}{X \rightarrow Y \sqcap Y^{\mathcal{C}}}$$

6.2 Axiomatisation for FDs in Complex-Value Databases

- The Armstrong Axioms, i.e.,

$$\frac{}{X \to Y}\, Y \leq X, \quad \frac{X \to Y}{X \to X \sqcup_N Y}, \quad \frac{X \to Y,\ Y \to Z}{X \to Z}$$

  form a minimal, sound and complete set of inference rules for the implication of FDs in the presence of records, and records and lists.

- Let $\mathcal{T}$ be any non-empty subset of {lists, sets, multisets} apart from {lists}. The generalised Armstrong Axioms, i.e.,

$$\frac{}{\mathcal{X} \to \mathcal{Y}}\, \mathcal{Y} \subseteq \mathcal{X}, \quad \frac{}{\{X\} \to \{Y\}}\, Y \leq X, \quad \frac{\mathcal{X} \to \mathcal{Y}}{\mathcal{X} \to \mathcal{X} \cup \mathcal{Y}},$$

$$\frac{}{\{X, Y\} \to \{X \sqcup_N Y\}}\, X, Y \text{ reconcilable}, \quad \frac{\mathcal{X} \to \mathcal{Y},\ \mathcal{Y} \to \mathcal{Z}}{\mathcal{X} \to \mathcal{Z}},$$

  form a minimal, sound and complete set of inference rules for the implication of FDs in the presence of records and $\mathcal{T}$. □

6.3 Real-World Constraints in Digital Halftoning

- convert continuous-tone image into binary one that looks similar
- input matrix A represents digital (gray) image, where a_{ij} represents brightness level of (i,j)-pixel in $N \times N$ pixel grid
- replace A by {0,1}-matrix B that is good approximation of A
- typical family of regions $\mathcal{R}$:

$$\begin{array}{|c|c|}\hline a & b \\ \hline c & d \\ \hline\end{array} \qquad \begin{array}{|c|c|}\hline a & b \\ \hline\end{array} \qquad \begin{array}{|c|}\hline a \\ \hline b \\ \hline\end{array} \quad ,$$

- B to minimise $dist(A,B) = \left| \sum_{(i,j)\in R} a_{i,j} - \sum_{(i,j)\in R} b_{i,j} \right|$ for all $R \in \mathcal{R}$
- simple example: $\{0, \frac{1}{2}, 1\}$-input matrix
- $[0, \frac{1}{2}]$ may be mapped to [0,1], [1,0] or [0,0]

- $[0, 0, 1, \frac{1}{2}]$ can be mapped to any of [0,0,0,1], [0,0,1,0], [0,1,0,0], [1,0,0,0], [0,0,1,1], [0,1,0,1], [1,0,0,1], [0,1,1,0], [1,0,1,0], [1,1,0,0]
- database: **Halftoning(Brightness,Input[Level],Output[Bit])**
- find {0,1}-matrix B that has for every regions of A a corresponding output region in database
- $A = \begin{pmatrix} 0 & 0 \\ \frac{1}{2} & \frac{1}{2} \end{pmatrix}$ has approximation $B = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$ but not $\begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$
- overall brightness and length of input region together determine set of all input regions independently from set of output regions:

$$\text{Halftoning(Brightness,Input}[\lambda]) \twoheadrightarrow \text{Halftoning(Input[Level])}$$

- further FDs:

$$\text{Halftoning(Input}[\lambda]) \longrightarrow \text{Halftoning(Output}[\lambda]), \text{ and}$$
$$\text{Halftoning(Output}[\lambda]) \longrightarrow \text{Halftoning(Input}[\lambda])$$