# データベース特論 I （情報科学専攻）

# 複雑計算論（複雑理工学専攻）

**Homepage** `http://www.gi.k.u-tokyo.ac.jp/~moris/lecture/database-theory/index.htm`

**目標**　データベース問合せ言語の設計方法と実行最適化技法の解説

理論的な側面が中心

**講義時間と場所**　毎週金曜日　午前 10 時 15 分から午前 11 時 45 分まで

講義室　理学部 7 号館 102 講義室

**講義予定日**

2000/10/06, 10/13, 10/20, 11/10, 11/17, 12/1, 12/8, 12/22,

2001/1/12, 1/19, 1/26 計 11 回の予定

10/27, 11/24, 12/15 は出張中にて休講

# 問合せ言語の意味論 1 — 再帰的問合せ

## 例題 — 生き物の食物連鎖

chain(X,Y) :- eats(X,Y)

chain(X,Y) :- chain(X,Z) & chain(Z,Y)

- eats(X,Y): X が Y の食料である.
- chain(X,Y): X は間接的に Y の食物である.

## 述語 原子式 基礎原子式

- 述語は小文字から始まる名前 (alpha-numeric) で表現
- 原子式 = 述語 + 引数
  例：eats(X,Y), eats(X,fish), eats(pelican, bear)
- 引数 = 変数 または 定数
  (関数記号を含む項については後に定義)
  変数は大文字から始まる名前で表現：X,Y,Z
  定数は小文字から始まる名前で表現：fish, pelican
- 基礎原子式 (ground atom) = 全ての引数が定数の原子式
  例：eats(pelican, bear)

**関係**　食物連鎖の関係 $EATS$

| FOOD | EATER |
|---|---|
| bug | fish |
| fish | fish |
| fish | bear |
| fish | pelican |
| pelican | fox |

- 関係データベースの関係 (relation) とは含まれるタップルのリスト (論理学での外延 extension) なので, 特に EDB (Extensional DataBase) 関係と呼ぶ

### EDB 関係を基礎原子式の集合で表現

- 述語 $eats$ を用意
  $eats(X, Y)$ が真 iff $(X, Y)$ が 関係 $EATS$ のタップル

  eats(bug,fish)
  eats(fish,fish)
  eats(fish,bear)
  eats(fish,pelican)
  eats(pelican,fox)

- EDB 関係に対して用意された述語を EDB 述語と呼ぶ
- EDB 関係と, 対応する EDB 述語の基礎原子式の集合を同一視
- EDB 述語の基礎原子式は略して EDB ファクトと呼ぶ

**ルール**　頭部 :- 本体

- 頭部は原子式
- 本体は 0 個以上の原子式の論理積
- 頭部の変数は本体に出現すると仮定する.
- 本体の原子式をサブゴールと呼ぶ
- 本体が空でないルールの頭部の述語は外延が明示されない, 内包的 (intensional) に定義される述語なので特に IDB(Intensional DataBase) 述語と呼ぶ
- 本体のサブゴールは IDB 述語および EDB 述語を含みうる
- ルールの集合を論理プログラムと呼ぶ
  chain(X,Y) :- eats(X,Y)
  chain(X,Y) :- chain(X,Z) & chain(Z,Y)
- 本体が空のルールだけで表現される原子式は, 基礎原子式であり, EDB ファクトと考える.

**IDB ファクト / IDB 関係**

- IDB 述語の基礎原子式を略して IDB ファクトと呼ぶ
  例：chain(bug,fish), chain(bug,bear)
- ルール中の全ての変数に定数を代入した結果を基礎代入例 (ground instance) と呼ぶ
  chain(bug,fish) :- eats(bug,fish)
  chain(bug,bear) :- chain(bug,fish) & chain(fish,bear)
- ルールにより推論される IDB ファクトの集合を IDB 関係と呼ぶ

## IDB 関係の計算 (Naive Evaluation)

- 入力: EDB 関係 $E$
- 出力: IDB 関係 $I$

1. 初期化
   IDB 関係 $I_0 := \phi$, $i := 0$
   全てのルールの全ての基礎代入例を生成する.
2. 反復ステップ
   基礎原子式は $E \cup I_i$ に含まれるとき真と考える.
   本体のすべての基礎原子式が $E \cup I_i$ で真のとき,
   頭部の IDB ファクトを真と推論し, IDB 関係 $I_{i+1}$ に追加する.
   $i := i + 1$
3. 新しい IDB ファクトを推論できなくなるまで反復ステップを繰返す.

## 例

| | |
|---|---|
| eats(bug,fish) | chain(X,Y) :- eats(X,Y) |
| eats(fish,fish) | chain(X,Y) :- chain(X,Z) & chain(Z,Y) |
| eats(fish,bear) | |
| eats(fish,pelican) | |
| eats(pelican,fox) | |

- $I_0 = \phi$
- $I_1 =$ {chain(bug,fish), chain(fish,fish), chain(fish,bear), chain(fish,pelican), chain(pelican,fox)}
- $I_2 = I_1 \cup$ {chain(bug,bear), chain(bug,pelican), chain(fish,fox)}
- $I_3 = I_2 \cup$ {chain(bug,fox)}
- 重複した無駄な推論が多い. 改善できないか？

**Naive Evaluation の改良**

- $A \in I_{i+1} \Leftrightarrow I_i$ $(i \geq 1)$ のとき $A$ を推論したルールの基礎代入例を

$$A \text{ :- } B_1 \& \ldots \& B_n$$

  $B_i \in I_i \Leftrightarrow I_{i-1}$ となる $B_i$ が必ず存在することに注意.

- $i$ 回目の反復ステップで新たに推論された集合 $I_i \Leftrightarrow I_{i-1}$ を $\Delta I_i$ と記述.

**Seminaive Evaluation**

1. 初期化
   全てのルールの全ての基礎代入例を生成する.
   $E$ の元だけから推論された IDB ファクトの集合を $I_1$ と $\Delta I_1$ に代入.
2. 反復ステップ
   (a) 本体中の１つの IDB ファクトは $\Delta I_i$ の元であり, 他の基礎原子式は $E \cup I_i$ の元となるルール $A \text{ :- } B_1 \& \ldots \& B_n$ により推論される IDB ファクトの集合を $J_{i+1}$.
   (b) $\Delta I_{i+1} := J_{i+1} \Leftrightarrow I_i, \qquad I_{i+1} := I_i \cup \Delta I_{i+1}$
3. 新しい IDB ファクトを推論できなくなる $(\Delta I_{i+1} = \phi)$ まで反復ステップを繰返す.

**例**

1. $I_1 = \Delta I_1 = \{$chain(bug,fish), chain(fish,fish), chain(fish,bear), chain(fish,pelican), chain(pelican,fox)$\}$
2. $J_2 = I_1 \cup$ $\{$chain(bug,bear), chain(bug,pelican), chain(fish,fox)$\}$
   $\Delta I_2 = \{$chain(bug,bear), chain(bug,pelican), chain(fish,fox)$\}$
3. $J_3 = \{$ $\underline{\text{chain(fish,fox)}}$, chain(bug,fox)$\}$
   $\Delta I_3 = \{$chain(bug,fox)$\}$
   chain(fish, fox) :- chain(fish, fish) & chain(fish, fox) に注意.

### エルブランモデルによる論理プログラムの解釈

$M$ を基礎原子式 (EDB ファクトと IDB ファクト) の集合.

$M$ が次の性質を満たすとき, 論理プログラム $P$ のエルブランモデルと呼ぶ.

- $P$ のルールの任意の基礎代入例 $A$ :- $B_1 \& \ldots \& B_n$ について、
  本体のすべての基礎原子式が $M$ の元ならば, 頭部の $A$ も $M$ の元.

### 例

```
eats(bug,fish)          chain(X,Y) :- eats(X,Y)
eats(fish,fish)         chain(X,Y) :- chain(X,Z) & chain(Z,Y)
eats(fish,bear)
eats(fish,pelican)
eats(pelican,fox)
```

エルブランモデルの例

- すべての EDB と IDB ファクトの集合: いかなるモデルも包含する.
- 最小のモデル:
  { eats(bug,fish), eats(fish,fish), eats(fish,bear), eats(fish,pelican), eats(pelican,fox),
  chain(bug,fish), chain(fish,fish), chain(fish,bear), chain(fish,pelican), chain(pelican,fox),
  chain(bug,bear), chain(bug,pelican), chain(fish,fox), chain(bug,fox)}
- どのモデルがプログラムの意図を反映しているのか？

**定理**　論理プログラム $P$ のエルブランモデル全体の共通部分は, $P$ のエルブランモデル (最小エルブランモデルと呼ぶ) である.

$$A \text{ :- } B_1 \& \ldots \& B_n$$

- 任意のエルブランモデル $M_i$. 共通部分 を $\cap M_i$.
- 任意の $B_i$ が $\cap M_i$ の元ならば, $B_i \in M_i$. $M_i$ はエルブランモデルなので,
  $A \in M_i$ . $M_i$ は任意なので $A \in \cap M_i$.

## １ステップの推論を表現する写像

- $I$ = 基礎原子式の集合, $P$ = 論理プログラム
- $I$ を仮定して 1 ステップ推論した結果
  $T_P(I) = \{A \mid A$ :- $B_1 \& \ldots \& B_n$ は $P$ のルールの基礎代入例,
  各 $i = 1, \ldots, n$ について $B_i \in I\}$
- naive evaluation は $T_P$ で表現できる
  $T_P{\uparrow}0 = \phi$
  $T_P{\uparrow}(n+1) = T_P(T_P{\uparrow}n) \quad n$ は非負整数
  $T_P{\uparrow}\omega = \cup\{T_P{\uparrow}n \mid n$ は非負整数 $\}$

## 例

$P$

eats(bug,fish)　　chain(X,Y) :- eats(X,Y)

eats(fish,fish)　　chain(X,Y) :- chain(X,Z) & chain(Z,Y)

eats(fish,bear)

eats(fish,pelican)

eats(pelican,fox)

$T_P{\uparrow}0 = \phi$

$T_P{\uparrow}1 = \{$eats(bug,fish), eats(fish,fish), eats(fish,bear), eats(fish,pelican), eats(pelican,fox)$\}$

$T_P{\uparrow}2 = \{$chain(bug,fish), chain(fish,fish), chain(fish,bear), chain(fish,pelican), chain(pelican,fox)$\}$

$T_P{\uparrow}3 = T_P{\uparrow}2 \cup \{$ chain(bug,bear), chain(bug,pelican), chain(fish,fox) $\}$

$T_P{\uparrow}4 = T_P{\uparrow}3 \cup \{$chain(bug,fox)$\}$

## 性質

- $T_P$ は単調: $I \subseteq J \Rightarrow T_P(I) \subseteq T_P(J)$
- $I$ がエルブランモデル $\Leftrightarrow T_P(I) \subseteq I$
- 最小エルブランモデル $= \cap\{I | T_P(I) \subseteq I\}$

**定理**　最小エルブランモデル $= T_P{\uparrow}\omega$

**証明**

$\cap\{I \mid T_P(I) \subseteq I\} = \cap\{I \mid T_P(I) = I\}$

- $\subseteq$-方向は自明.

  $S := \{I|T_P(I) \subseteq I\}$ として $T_P(\cap S) = \cap S$ を証明

  $\cap S \in \{I \mid T_P(I) = I\}$ なので, $\cap S \supseteq \cap\{I \mid T_P(I) = I\}$

- $I \in S$ に対して $\cap S \subseteq I$

  $T_P(\cap S) \subseteq T_P(I) \subseteq I$

  $T_P(\cap S) \subseteq \cap S$

- $T_P(T_P(\cap S)) \subseteq T_P(\cap S)$

  $T_P(\cap S) \in S$

  $T_P(\cap S) \supseteq \cap S$

$\cap\{I \mid T_P(I) = I\} = T_P{\uparrow}\omega$

- $I$ は $T_P(I) = I$ を満たすとする

  $\phi \subseteq T_P(I)$ の両辺に $T_P$ を適用

  $T_P{\uparrow}1 = T_P(\phi) \subseteq T_P(I) = I$

  $T_P{\uparrow}2 = T_P(T_P{\uparrow}1) \subseteq T_P(I) = I$

  $\vdots$

  $T_P{\uparrow}\omega \subseteq I$

  $T_P{\uparrow}\omega \subseteq \cap\{I \mid T_P(I) = I\}$

- $T_P(T_P{\uparrow}\omega) \subseteq T_P(\cup\{T_P{\uparrow}n \mid n \geq 0\}) \subseteq \cup\{T_P{\uparrow}n \mid n \geq 0\} = T_P{\uparrow}\omega$

  任意の $A \in T_P(T_P{\uparrow}\omega)$ について, $A$ :- $B_1\& \ldots \& B_n$ が存在し, $B_i \in T_P{\uparrow}\omega$.

  ある $m$ が存在して $B_i \in T_P{\uparrow}m$. $A \in T_P{\uparrow}(m+1) \subseteq T_P{\uparrow}\omega$.

  $T_P{\uparrow}\omega \in S$. よって $T_P{\uparrow}\omega \supseteq \cap S = \cap\{I \mid T_P(I) = I\}$.

**不動点**　$T_P(I) = I$ を満たす $I$ を $T_P$ の不動点とよぶ.

**定理**　$T_P{\uparrow}\omega$ は $T_P$ の不動点.

**証明**　$T_P(T_P{\uparrow}\omega) \subseteq T_P{\uparrow}\omega$ は既に証明
$T_P(T_P{\uparrow}\omega) \supseteq T_P{\uparrow}\omega$ も同様に証明可能.

**最小不動点**

- $\cap\{I \mid T_P(I) = I\} = T_P{\uparrow}\omega$ なので $T_P{\uparrow}\omega$ は $T_P$ の不動点の中で最小.
  最小不動点と呼ぶ.

**問題**　同世代（same generation）を表現する次のプログラムの最小エルブランモデルを求めよ.

sg(X,Y) :- flat(X,Y)

sg(X,Y) :- up(X,Z) & sg(Z,U) & down(U,Y)

up(1,2) up(2,3) flat(2,5) flat(3,4) down(4,5) down(5,6)

**問題**　有向グラフの道を表現する次のプログラムの最小エルブランモデルを求めよ.

path(X,Y) :- arc(X,Y)

path(X,Y) :- arc(X,Z) & path(Z,Y)

arc(a,b) arc(b,a) arc(a,c) arc(c,d) arc(d,c)

**問題**　等式を表現する次のプログラムの最小エルブランモデルを求めよ.

eq(X,Y) :- eq(X,Z) & eq(Z,Y)

eq(X,Y) :- eq(Y,X)

eq(a,b) eq(c,b)

**問題**　次のプログラムの最小エルブランモデルを求めよ.

buys(X,Y) :- likes(X,Y)

buys(X,Y) :- trendy(X) & buys(Z,Y)

trendy(a) trendy(c) likes(b,item1)

**問題**　次の 2 つのプログラムが, trendy と likes の任意の EDB ファクトに対して同一の最小エルブランモデルをもつことを示せ.

- buys(X,Y) :- likes(X,Y)
  buys(X,Y) :- trendy(X) & buys(Z,Y)
- buys(X,Y) :- likes(X,Y)
  buys(X,Y) :- trendy(X) & likes(Z,Y)