不完全知識を扱う高水準論理型データベースの問合せのための関係代数表現

大原剛三† 馬場口 登† 北橋忠宏†

本論文では，例外を伴う不完全知識を許容する高水準論理型データベース（ALDB：Advanced Logical DataBase）における知識の関係代数による表現法を提案し，その評価法を議論する．ALDBは，完全知識と不完全知識の両方をもつことにより，従来の演繹データベース（DDB：Deductive DataBase）が例外を適切に扱えないという欠点を解消したデータベースである．ここでは，既存の関係データベースの基本演算である関係代数を用いた問合せを実現するための知識の関係代数表現を与える．提案手法においては，例外関係，および例外間に生じる階層関係を明確に表す知識表現，*exc* 表現を新たに導入することにより，例外を伴う知識を関係代数に変換する．*exc* 表現を用いて変換した関係代数式は，DDBの一種である層状データベースで扱い得る形式となるので，代数式の評価，およびデータベースの意味論は層状データベースの理論が適用可能となる．また，他の例外表現として circumscription における abnormal 述語をとりあげ，*exc* 表現と比較検討する．

Relational Algebraic Representation for Query Procedure in Advanced Logical DataBase Handling Incomplete Knowledge

KOUZOU OHARA,† NOBORU BABAGUCHI † and TADAHIRO KITAHASHI †

In this paper, we discuss a relational algebraic representation of knowledge in Advanced Logical DataBase, ALDB. ALDB is an extension of the deductive database, which is capable of handling both the complete knowledge and the incomplete knowledge containing exceptions. We introduce a new predicate, *exc-A*, representing the exception of a predicate *A*, to distinguish predicates expressing exceptions from others. This representation enables us to translate the logical formulas representing knowledge into relational algebraic formulas. A query procedure allowing exceptions can be realized by means of the relational algebraic formulas translated in terms of *exc-representation*. The evaluation process and semantics of ALDB can be defined by the theory of the stratified database. Furthermore we compare *exc-representation* with *abnormal predicate* in *circumscription*.

1. まえがき

近年，盛んに議論されている新しいデータモデルの1つに演繹データベース（DDB：Deductive DataBase）[1]がある．DDBは形式論理に数学的基礎を置く論理型データベースの1つであり，従来のデータベースが事実しか扱わないのに対し，それらの事実間に成立する規則を一階述語論理（もしくはそれに立脚する言語）によって表現したルールも扱う．また，問合せに関しては，データベース中の事実に加え，自身のもつ事実とルールから論理的に帰結される事実も応答とするので，より柔軟な問合せ応答が可能である．

このようなDDBの大規模化を図る際，問題となるのは例外となるデータの混入である．例外がデータ中に混入した場合，DDB中に矛盾が発生する．一階述語論理では矛盾のある理論からは任意の事実が証明されるため，DDBがデータベースとしての機能を十分に果たさなくなる可能性がある．しかしながら，このような例外の混入は，現実データを対象にデータベースを大規模化する際には不可避な事柄であり，例外によって生じた矛盾の回復には多大な労力が必要とされている[2]．

以上の背景から，筆者らは知識集合中に完全な知識と不完全な知識の混在を許容する，高水準論理型データベース（ALDB：Advanced Logical DataBase）を提案し，基本的な枠組や反駁証明を用いた問合せを検討してきた[3),4)]．ここでいう完全な知識とは，常に正しい知識であり，不完全な知識とは，例外を許容する常に正しいとは限らない知識である．このような不完全

† 大阪大学産業科学研究所
The Institute of Scientific and Industrial Research, Osaka University

な知識を適切に処理することにより，例外に伴う問題を解消することが可能となる．

本論文では，ALDBにおける問合せ手続きの1つとして，関係データベースの基本演算である関係代数[5]を用いた問合せ手続き[6),7)]を実現するために，知識の関係代数表現について検討する．データベースにおける問合せを想定した場合，問合せが一般に変数を用いて複数の解答を要求するものであるという条件のもとでは，筆者らの従来手続き[4)]は，個々の事実の証明を試みるため，必ずしも効率的とはいえない．また，集合的に扱いうる事実を個別にしか扱わないという面で，操作性にも問題をもつ．

これに対し関係代数を用いると，事実を関係という集合的概念で扱うため，操作性に優れ，また，複数の同種の事実の証明に伴うバックトラックも伴わないので，効率面でも性能向上が期待できる．さらに関係代数による問合せ手続きは，DDBの研究において数多くの知見が得られており，基本部分にはその技術を有効利用することが可能である．

本論文では主に，ALDBにおいて関係代数を用いる際に主要な問題となる不完全知識の取り扱いについて論ずる．提案手法では，例外関係や例外間に生じる階層関係に着目し，*exc* 表現という新たな知識表現を導入することにより，不完全な知識の関係代数への変換を実現している．*exc* 表現を用いて変換された不完全知識に対応する関係代数式は，条件部に負リテラルを許容するルールに解釈できる形式となる．従って，関係代数式の評価には，条件部に負リテラルを含むルールを扱う層状データベース (SDB: Stratified DataBase) における関係代数式の評価法が適用可能となり，ALDBの意味論の定義にもSDBの理論を適用する．ただし，評価法に関しては，ALDBにおける冗長な処理を削除するため，評価に必要な階層化に伴う諸定義を修正する．

また，他の例外表現として circumscription における abnormal 述語[8)]をとりあげ，*exc* 表現との対比を，関係代数を用いてルールを評価するという観点から検討する．

2. ALDBの概要

2.1 ALDBの基本構成

ALDBの基本構成は図1のとおりである。構成の基本はDDBと同様で，事実の集合である外延データベース (EDB: Extensional DataBase) と知識の集合である内包データベース (IDB: Intensional DataBase)，およびそれらを操作する推論エンジン

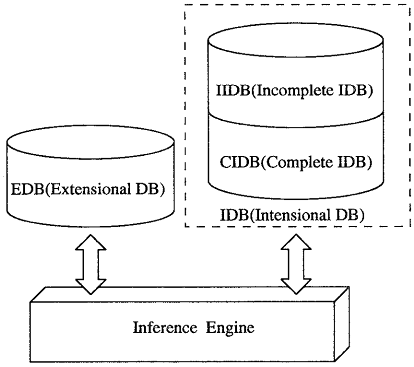


図1 ALDBの基本構成
Fig. 1 Basic configuration of ALDB.

(Inference Engine) から構成される．ただし，ALDBでは知識の集合であるIDBは，完全な知識の集合である完全IDB (CIDB: Complete IDB) と不完全な知識の集合である不完全IDB (IIDB: Incomplete IDB) の2つから構成される．

2.2 ALDBの知識表現

ALDBでは，DDBで用いられている事実と完全な知識に加え，更に不完全な知識と制約という知識を扱い，それぞれを次のように表現する．

［定義1］(ALDBの知識表現)　ALDBのもつ知識・データは以下のように表現する．

- 完全ルール (完全な知識)：$r \leftarrow s_1, \cdots, s_m$
  「$s_1, \cdots, s_m$ ならば**必ず** r である」
- 不完全ルール (不完全な知識)：$r \Leftarrow s_1, \cdots, s_m$
  「$s_1, \cdots, s_m$ ならば**通常** r である」
- ファクト (事実)：$p(t_1, \cdots, t_m)$
  「$t_1, \cdots, t_m$ は p である」
- 制約：$\perp \leftarrow s_1, \cdots, s_m$
  「$s_1, \cdots, s_m$ が**同時に成り立つ**ことは**矛盾**する」

ここで，$r, s_i\,(1 \leq i \leq m)$ は正リテラル，p は m 項述語記号，$t_j\,(1 \leq j \leq m)$ は定数とし，これらのルールは次の条件を満たすとする．

1. ルールの帰結部 r に現れる変数は，条件部 $s_1, \cdots, s_m$ のいずれかのリテラル中に少なくとも1度は現れなければならない．
2. 制約の条件部 $s_1, \cdots, s_m$ の各リテラルに現れる変数は，他のリテラル中に少なくとも1度は現れなければならない．
3. 算術述語 (算術比較を表す述語) に現れる変数はルール条件部の他の算術述語以外の述語に現れなければならない．
4. ルール中のリテラルは定数記号以外の関数記号を

含まない．　□

ここで挙げた条件は，ルールを関係代数を用いて評価した際，得られる関係が無限集合にならないように設けられた条件であり，条件 1.，2. は DDB における値域制限[9]と同様のものである．条件 2. に関しては後述の制約の変換の際に，条件部のリテラルを帰結部へ移行する等価変換を行うために必要となる．

また，定義 1 の知識表現では，不完全ルール以外は確定ホーン節で記述される．不完全ルールにおける記号 $\Leftarrow$ は例外許容型オペレータであり，不完全ルールの論理的意味を，「**不完全ルールは，ルールの条件部が真となり，かつルールの帰結部の否定が導かれないとき，ルールの帰結部を結論する**」と定義する．

例えば，‘鳥は通常飛ぶ’という知識を表す不完全ルール

$$fly(x) \Leftarrow bird(x)$$

を用いることにより，次のような

$$fly(x) \leftarrow bird(x), \neg ostrich(x), \neg penguin(x), \cdots$$

という条件部に負リテラルを用いて例外（*ostrich*, *penguin*, …）を羅列する表現より，簡潔で，より直観的に意味の捉えやすい表現が可能となる．また，新たな飛ばない鳥に関する知識が追加されても，前述の不完全ルールは，変更する必要はなく，有効である．

なお，ALDB の扱うリテラルを正リテラルに限定するのは，文献 4) の手法において確定ホーン節による反駁証明を用いるためである．このことから述語 r の否定は *not-r* という r とは別の述語によって表現される．

さらに，上の定義の制約というルールは，ALDB の矛盾を検出するためのものである．制約は完全ルールの特別な形式と捉えることができ，帰結部の記号 $\perp$ は矛盾（恒偽命題）を表す記号である．

以下，EDB 中に存在する述語（ファクトを表す述語）を EDB 述語，IDB 中のルールで定義されている述語（ルール帰結部の述語）を IDB 述語という．

3．関係代数の導入

前述のように，問合せにおける効率，操作性の向上を実現するために本論文では関係代数を用いた問合せの実現を目的とする．DDB においても関係代数による問合せは提案されているが，その対象とする知識は ALDB におけるファクトと完全ルールだけで表現されうるものに過ぎない．しかし，DDB の問合せ技法面における知見を ALDB にも有効利用することが望ましく，そのためには不完全ルール，および制約の取り扱いが問題となる．ここでは，主に不完全ルール，制約の関係代数式への変換を議論する．

不完全ルール，制約の変換に先立ち DDB におけるファクト，および完全ルールの関係代数への変換[5]を簡単に述べる．ただし，その詳細は本論文の本質ではないので割愛し，完全ルールに関しては *Rel* というマクロ的演算子を新たに導入して以下のように定義する．

［定義 2］（ファクト，完全ルールの関係代数表現）

- ファクト

  同一の述語名をもつファクトの集合をその述語名を関係名とした 1 つの関係とする．

- 完全ルール

  完全ルール

  $$r \leftarrow s_1, \cdots, s_m$$

  に対する関係代数式を次のように表す．

  $$R = Rel(S_1, \cdots, S_m)$$

  ただし，$R, S_1, \cdots, S_m$ はそれぞれ述語 $r, s_1, \cdots, s_m$ に対応する関係とし，*Rel* はルールの条件部の述語に対応する関係を引数としてルールの条件部に対応する関係を計算する演算子とする☆．　□

以下，EDB 述語，IDB 述語に対する関係をそれぞれ EDB 関係，IDB 関係といい，ルールにおける小文字記号を述語名，関係代数式における大文字記号を関係名として表現する．

制約，不完全ルールの関係代数への変換については，例外が関与するため，ここでは，例外の階層性に着目した変換を提案する．

3.1 例外関係の表現

本節では，まず不完全ルールの直観的意味を考察する．不完全ルールの結論集合は，不完全ルールの論理的意味（2.2 節参照）より，ルールの条件部を真にする要素の集合のうち，ルールの帰結部の否定を満たさない要素の集合である．これに対し，ルールの条件部を満たし，かつルールの帰結部の否定を満たす要素は，ルールの条件部により定義されるクラスにおける例外といえる．不完全ルールの条件部を満たす定数の組の集合は，不完全ルールを完全ルールとして評価すれば得られるため，不完全ルールの直観的意味を「**不完全ルールを完全ルールとみなした結果から例外となるものを除いたもの**」と定義する．

この定義に沿って不完全ルールを変換するには，例

☆ Rel は実際には次のような関係代数式で記述される．

$$Rel(S_1, \cdots, S_m) \triangleq \pi_\alpha(\sigma_{\theta_1}(S_1) \bowtie \cdots \bowtie \sigma_{\theta_m}(S_m))$$

ここで，α は R の属性集合であり，π，σ，$\bowtie$ はそれぞれ関係代数演算の射影，選択，自然結合を表し，θ_i ($1 \leq i \leq m$) は選択の条件を表す．

外の特定が必要となる．以下では，ALDB の知識表現が問合せ評価の際に抱える問題を考察する．

まず，次のようなルールを考えよう．

$notfly(x) \Leftarrow animal(x)$ (1)

$fly(x) \Leftarrow bird(x)$ (2)

$notfly(x) \leftarrow ostrich(x)$ (3)

$animal(x) \leftarrow bird(x)$ (4)

$bird(x) \leftarrow ostrich(x)$ (5)

ここで，式(1)を評価するには，取り除くべき例外を特定する必要があるが，上のルール集合には式(1)の例外を示す明示的判断基準は存在しない．実際には式(2)の評価結果がその例外にあたる．同様に，式(2)に対する例外は式(3)の評価結果となる．従って，式(3)の評価結果は式(1)の例外の例外となる．「例外の例外」という関係は，例外間に生じる階層といえる．一般的な概念の集合を得るには，より特殊な概念の集合を先に確定する必要があり，例外階層はその確定順序，すなわち不完全ルールの評価順序を表すものといえる．この例外階層に関する情報はこれらのルールから何も得られない．

以上の議論から定義1で示したALDBの知識表現について，関係代数を導入するにあたり次の問題を解決する必要がある．

- 例外の特定を可能にする
- 例外の階層関係を明確にする

そこで我々は，この問題を解決するために，次のような新たな知識表現を導入する．その知識表現は，例外という意味の'*exception*'という語を省略した *exc* という記号を用いて，述語 r の例外，もしくは例外を含む述語を $exc\text{-}r$ と表現するものである．さらに述語 r の例外の例外は $exc^2\text{-}r$ と表現する．この場合，2のように *exc* の右肩に記述する数字を次数と呼び，次数により例外の階層を表現する．*exc* の次数に関する一般的な表記規則を次のように定義する．

[定義3]（*exc* の表記規則） *exc* の表記規則は次のように定義される．

$exc^0\text{-}r \triangleq r$

$exc^1\text{-}r \triangleq exc\text{-}r$

$exc^n\text{-}r \triangleq exc(exc^{n-1}\text{-}r) \quad (n=1, 2, \cdots)$ □

このように *exc* を用いた表現を *exc* 表現と呼び，$exc^n\text{-}r$ は1つの述語として扱う．

式(1)～(3)は次のように *exc* 表現で記述でき，例外関係，例外階層が明確に表されていることがわかる．

$notfly(x) \Leftarrow animal(x)$ (6)

$exc\text{-}notfly(x) \Leftarrow bird(x)$ (7)

$exc^2\text{-}notfly(x) \leftarrow ostrich(x)$ (8)

exc 表現は，ルールの帰結部の述語に付されるラベルと考えることができる．すなわち，あるルールの例外となる帰結部の述語にのみ，対応する *exc* 表現をラベル付けする．筆者らは，このようなラベル付けのメカニズム，つまり *exc* 表現の生成法[10]に関しても検討しており，機会を改めて論ずる．

ここで，以上の *exc* 表現においては $fly(x)$ に関する問合せが不可能ではないかという危惧が生じるが，これは前述のように *exc* 表現があくまでもラベルであり，本来の述語の情報を消失させるものではないということから解消される．つまり，$fly(x)$ と $exc\text{-}notfly(x)$ の対応をシステムが保持することにより，$fly(x)$ に関する問合せは可能となる．

また，*exc* 表現では次数の大きな述語ほど特殊な概念を表すことになるので，次数の大きい述語から順次評価することが，例外階層を反映した評価となる．従って，次数は評価順序において重要な基準となる．

3.2 不完全ルール・制約の関係代数式への変換

定義1，および定義3より，直ちに次のような不完全ルールの直観的意味に沿った関係代数式への変換が定義される．ただし，不完全ルールの変換においては，例外となるべきものはすべて *exc* 表現されていると仮定する．

[定義4]（不完全ルールの変換） 不完全ルール

$r \Leftarrow s_1, \cdots, s_m$

は，次の関係代数式に変換できる．

$R = Rel(S_1, \cdots, S_m) - exc\text{-}R$ □

同様に，例外を定義する不完全ルール（例外ルール）は次のように変換できる．

[定義5]（例外ルールの変換） 例外ルール

$exc^n\text{-}r \Leftarrow s_1, \cdots, s_m$

は，次の関係代数式に変換できる．

$exc^n\text{-}R = Rel(S_1, \cdots, S_m) - exc^{n+1}\text{-}R$ □

ここで，制約と例外の関連を簡単に述べる．ALDBにおいては，制約の条件部が真になると矛盾が発生したと判断し，矛盾を引き起こすファクトを例外として扱う．そして，制約の条件部に現れるリテラルを帰結部にもつ完全ルールのいずれかを不完全ルールに変換することにより矛盾を解消する．従って，矛盾を引き起こした例外は不完全化されたルールによって許容されることになる．

以上の議論から次の定理が得られる．なお，以降のすべての定理の証明は付録に記す．

[定理1]（制約の変換） 制約

$\perp \leftarrow s_1, \cdots, s_m$

は，ある $s_i \, (1 \leq i \leq m)$ について s_i が不完全ルールの帰

結部に現れる場合，次のように関係代数式に変換できる．

$$exc\text{-}S_i = Rel(S_1, \cdots, S_{i-1}, S_{i+1}, \cdots, S_m) \qquad \square$$

s_i の選択は，不完全ルールの帰結部に現れるかどうかに依存し，不完全ルールの帰結部に現れる s_i が複数存在しても，各 s_i について，上のような関係代数式が一意に決定される．

以上の定義から，例えば前述の式(4)～(8)に対する関係代数式は次のようになる．

$$ANIMAL(x) = Rel(BIRD(x)) \qquad (9)$$
$$BIRD(x) = Rel(OSTRICH(x)) \qquad (10)$$
$$NOTFLY(x) = Rel(ANIMAL(x)) - exc\text{-}NOTFLY(x) \qquad (11)$$
$$exc\text{-}NOTFLY(x) = Rel(BIRD(x)) - exc^2\text{-}NOTFLY(x) \qquad (12)$$
$$exc^2\text{-}NOTFLY(x) = Rel(OSTRICH(x)) \qquad (13)$$

4. 関係代数式の評価法

ここでは上で定義した関係代数式の評価法を検討する．不完全ルールに対する次のような関係代数式

$$exc^n\text{-}R = Rel(S_1, \cdots, S_m) - exc^{n+1}\text{-}R \quad (n=0, 1, 2, \cdots) \qquad (14)$$

は，

$$exc^n\text{-}r \leftarrow s_1, \cdots, s_m, \neg exc^{n+1}\text{-}r \qquad (15)$$

というルールで記述できる．集合差は不完全ルールに対応する関係代数式に高々一つしか現れないので，式(15)のように記述した場合，負リテラルも高々1つしか現れない．このことから，前に議論したALDBの知識を表す関係代数式は，実際にはSDBが扱いうる関係代数式の範疇に包含される．SDBとは，DDBにおいてルールの条件部に負リテラルを許容したモデルである．従って，ALDBの問合せにおいてもSDBの関係代数式の評価法[5]が適用できることになるが，処理の冗長性を省くために，以下において定義に若干の修正を加える．

4.1 関係の階層化

SDBにおいては，従来のDDBで用いられてきた単調な関係代数演算から成る関係代数式集合に対する評価法，例えばセミナイーブ評価法[5]のようなものを適用するために，述語集合全体を階層化している．これは，関係代数式に現れる集合差の非単調性を除去するためであり，階層化はルールの条件部のすべての述語の階層が，帰結部の述語の階層以下になるように定義される．特に負リテラルに関しては，集合差において除かれるべき関係に対応することから，必ず帰結部の述語の階層より低くなるように定義される．このように階層を定義し，順次，低い階層から評価することにより全体の評価が可能となる．

一方，ALDBにおいては式(14)における項 $exc^{n+1}\text{-}R$ は，もとの不完全ルール中には現れない．従って，SDBの階層化の定義[5]を適用するには，式(14)のような関係代数式をさらに式(15)のようなルールに変換しなければならない．このような変換は明らかに冗長であるため，ここではSDBの階層化の定義における階層化の対象をルールにおける述語から，変換後の関係代数式に現れる関係に変更する．これにより，関係代数式をさらにルールに変換することなく，関係代数式から直接，階層を定義することが可能となる．

次に関係の階層化の定義を示す．ここで，式(14)における項 $exc^{n+1}\text{-}R$ を不完全ルールの例外項と呼び，以下，式集合とは関係代数式の集合を指すものとする．

［定義6］(関係の階層化)　関係の階層化は次のように定義される．

1. 式集合中のすべての関係の階層を0とする．
2. IDB関係 R に対して以下の操作を行う．
   - R に関する関係代数式のすべての例外項 E に対して
     $$stratum[R] := \max(stratum[R], 1 + stratum[E])$$
   - R に関する関係代数式の例外項でないすべての関係 S に対して
     $$stratum[R] := \max(stratum[R], stratum[S])$$
3. どの層にも変化がなくなるまでステップ2を繰り返す．　□

このように階層化した場合，次の定理が成り立つ．

［定理2］　任意の関係 $exc^i\text{-}R, exc^j\text{-}R$ について $i<j$ ならば，

$$stratum[exc^j\text{-}R] < stratum[exc^i\text{-}R]$$

である．　□

定理2より，階層の低い関係から順次評価することは，次数の高い関係から順次評価することになり，前述の評価順序を反映しているといえる．

各層は前述のセミナイーブ評価法により評価が可能であることに着目し，我々は関係代数による問合せを提案している[6),7)]．

4.2 階層化可能性

本節では，任意の式集合に対する階層化可能性[5]を定義する．

［定義7］(階層化可能)　式集合に現れる関係の集合が有限個の互いに素な全順序集合に分割できるとき，その式集合は階層化可能であるという．　□

本論文では，問合せの対象とする式集合はすべて階層化可能であることを前提とする．次に，階層化可能性を考察していく準備となる関係代数式集合の依存グラフの定義を示す．

[定義 8]（依存グラフ） 次のようなノードと有向リンクから構成されるグラフを依存グラフという．
ノード：関係代数式における関係名
有向リンク：関係代数式における右辺の関係名（例外項を含む）から左辺の関係名へ結んだもの □

例えば，前述の式(9)〜(13)に関する依存グラフは**図 2**のようになる．

依存グラフを上のように定義したとき，次の定理が成り立つ．

[定理 3] 階層化可能な式集合に対する依存グラフには，不完全ルールの帰結部を表す関係に対応するノードから，その不完全ルールの例外項に対応するノードへの経路が存在しない． □

定理 3，および図 2 から式(9)〜(13)は階層化できることがわかり，その階層は次のようになる．

層 0：*OSTRICH*, *BIRD*, *ANIMAL*, exc^2-*NOTFLY*
層 1：*exc-NOTFLY*
層 2：*NOTFLY*

さて，階層化不可能な式集合は非単調推論における多重拡張という現象を引き起こす．これは，関係代数式の評価順序が一意に決定できないため，複数の評価順序それぞれについて，異なった解を生じる現象である．これは，ALDB における重要課題の 1 つであるが，本論文の問合せは，前述のように対象とする式集合は階層化可能であるという前提より多重拡張は現れない．

5．検　　討

ここでは，ALDB のデータベースとしての意味論，および *exc* 表現と他の例外表現との対比を検討する．

5.1 ALDB の意味論

通常，論理型データベースにおいては，その意味論として演繹的な証明手続きにより真偽を判定する手続き的意味論と，モデルを用いて真偽を判定する宣言的意味論の両者が議論され，両者の一致が焦点となる．ALDB における意味論については，前述のように ALDB の知識表現が *exc* 表現を用いると SDB の知識表現に包含されるので，SDB についての理論[9] がそのまま適用できる．

本論文における ALDB の関係代数による問合せは，SDB との対応より EDB∪CIDB∪IIDB に対する極小モデルを計算するので[5]，SDB 同様，ALDB の宣言的意味論に対応していることになる．

また，ALDB の手続き的意味論については，不完全ルールを 4.1 節の式(15)のようなルールに読み換えることにより，通常の SDB と同じ議論が可能となり，宣言的意味論との対応も与えられる[9]．

5.2 他の例外表現との対比

例外を表現する代表的手法としては，circumscription における abnormal 述語[8] を用いたものがある．circumscription とは，知識集合中に存在する述語 r について，r を満たすべきものは知識集合中でそうであると記述されたものだけであるという言明を与えるものである．この circumscription を例外を表す「異常である」という意味の abnormal 述語（一般に $ab(x)$ と記述）に対して行うことにより，知識集合中に記述されている例外以外の例外は考慮しなくてもよいという仮定を得る．

ここで，3.1 節で用いた例をもう一度考えてみよう．式(1)〜(3)に対する *exc* 表現は式(6)〜(8)であった．ここで，式(1)，(2)は abnormal 述語を用いると次のように表現できる．

$$notfly(x) \leftarrow animal(x), \neg ab1(x) \quad (16)$$
$$ab1(x) \leftarrow fly(x) \quad (17)$$
$$fly(x) \leftarrow bird(x), \neg ab2(x), \quad (18)$$
$$ab2(x) \leftarrow ostrich(x) \quad (19)$$

これらのルールは確かに例外を特定し，かつ例外の階層も表現している．式(16)〜(19)における $ab1$, $ab2$ に対して circumscription を行うと，「$ab1(x)$ を満たすものは $fly(x)$ を満たすものだけである」，「$ab2(x)$ を満たすものは $ostrich(x)$ を満たすものだけである」という言明が得られ，従って，式(3)〜(5)も考慮すると，「$notfly(x)$ を満たすものは $animal(x)$ を満たし，かつ $fly(x)$ を満たさないものと，$ostrich(x)$ を満たすものである」，「$fly(x)$ を満たすものは $bird(x)$ を満たし，かつ $ostrich(x)$ を満たさないものである」という結論を導くことができる．

また，式(3)〜(5)，式(16)〜(19)は SDB におけるルールと捉えることができる．よって，$notfly(x)$ や $fly(x)$ を満たすものの集合は，直接 SDB に対する評

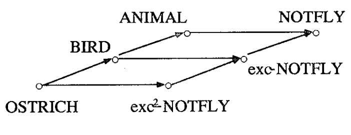


図 2 式(9)〜(13)に関する依存グラフ
Fig. 2 A dependency graph for formulas (9) to (13).

価法を用いることにより定めることが可能である．

しかしながら，これらのルールを関係代数を用いて評価する場合，次のような不都合が生じる．例えばデータベース中に次のようなルールが存在したとする．

$$fly(x) \leftarrow airplane(x) \quad (20)$$

この場合，式(17)を評価する際には式(20)も参照される．従って，$ab1(x)$ に対応する関係，つまり問合せに対する解を求めるまでに生成される中間関係に，冗長な要素が含まれ計算効率の低下につながることになる．このような中間関係の増大を避けるには，$ab1(x)$ のような例外を表す述語には，元となるルールの条件部（$animal(x)$）と交わりをもたないクラス（$airplane(x)$）の要素は含めるべきではない．

そこで，式(17)を次のように書き換えることを考える．

$$ab1(x) \leftarrow bird(x) \quad (21)$$

この場合，同様にして「$notfly(x)$ を満たすものは $animal(x)$ を満たし，かつ $bird(x)$ を満たさないものと，$ostrich(x)$ を満たすものである」,「$fly(x)$ を満たすものは $bird(x)$ を満たし，かつ $ostrich(x)$ を満たさないものである」という結論が導出でき，前述の中間関係の増大は生じない反面，「例外の例外」という例外の階層関係が表現できなくなってしまう．

一方，*exc* 表現においては 3.1 節で述べたように，あるルールの例外となる述語にのみ，対応する *exc* 表現をラベル付けするので，例外の階層を表現しつつ，なおかつ中間関係の増大を抑制することができる．例えば，式(2)の不完全ルールの $fly(x)$ には式(7)のように $exc\text{-}fly(x)$ がラベル付けされるが，式(20)の $fly(x)$ にはラベル付けされない．このラベル付けのメカニズムは，述語のクラス階層を参照することにより実現可能である[10]．

ただし，式(21)を次のような

$$ab1(x) \leftarrow bird(x), \neg ab2(x)$$

というルールで表現するなら，例外の階層を表現しつつ，なおかつ関係代数を用いた評価における中間関係の増大を抑制することが可能であり，*exc* 表現と同様となる．

また，前述のように *exc* 表現はラベル付けと捉えることができることから，本質的なルールの書き換えが不要であり，従って，不完全ルールを abnormal 述語を用いた表現に書き換えるよりも，ALDB 本来のルールを基礎とした簡潔な知識表現が実現できる．

6. む す び

本論文では不完全な知識を伴う論理型データベース ALDB の関係代数を用いた問合せ手続きを実現するための知識の関係代数表現を提案した．提案手法では，例外関係，および例外階層を明確に表現する *exc* 表現を導入することにより，ルール形式の知識の関係代数式への変換法を実現し，SDB で用いられる評価法を利用することを可能とした．また，評価に必要な階層化に伴う諸定義を ALDB において冗長な処理が生じないよう再定義した．さらに，他の例外表現として circumscription における abnormal 述語をとりあげ，*exc* 表現との対比を検討した．今後の課題としては，本手法では対象外であった非単調推論に伴う多重拡張への対処が挙げられる．

謝辞　なお，本研究の一部は文部省科学研究費，および電気通信普及財団の補助による．

参 考 文 献

1) Minker, J.: *Foundations of Deductive Database and Logic Programming,* Morgan Kaufmann (1987).
2) 西尾章治郎：大規模データベースにおける知識獲得，情報処理，Vol. 34, No. 3, pp. 343-350 (1993).
3) 馬場口登，大川剛直：完全/不完全知識を扱う高次推論型データベースにおける知識獲得，文部省科学研究費重点領域（知識科学）成果報告論文集，pp. 392-399 (1994).
4) 井戸譲治，馬場口登：完全/不完全なルールに対する推論手続き，情報処理学会人工知能研究会報告，93-AI-86-3, pp. 17-24 (1993).
5) Ullman, J. D.: *Principles of Database and Knowledge-base Systems,* Vol. 1, Computer Science Press (1988).
6) 大原剛三，井戸譲治，馬場口登，北橋忠宏：関係代数による高次論理型データベース ALDB の問合せ手続き，第 48 回情報処理学会全国大会論文集，7 F-5 (1994).
7) 大原剛三，井戸譲治，馬場口登，北橋忠宏：高水準論理型データベースにおける問合せ手続き，1994 年度人工知能学会全国大会論文集，19-6 (1994).
8) McCarthy, J.: Applications of Circumscription to Formalizing Common Sense Knowledge, *Artif. Intell.,* Vol. 28, pp. 89-116 (1986).
9) 横田一正，宮崎収兄：新データベース論，共立出版 (1994).
10) 大原剛三，馬場口登，北橋忠宏：不完全な知識における例外階層についての一考察，第 49 回情報処理学会全国大会論文集，6 J-9 (1994).

付　　録

(定理 1 の証明)　次のような制約

$$\perp \leftarrow s_1, \cdots, s_m$$

は，論理的に等価変換することにより次のように記述できる．

$$\neg s_i \leftarrow s_1, \cdots, s_{i-1}, s_{i+1}, \cdots, s_m$$

ここで，$1 \leq i \leq m$ である．この等価関係は任意の i について成り立つので，ある述語 s_i が不完全ルールの帰結部に現れるとしても一般性を失わない．前述のように，$s_1, \cdots, s_m$ のすべてを真にする定数の組は s_i の例外として扱われる．ここで，$s_1, \cdots, s_m$ すべてを真にする定数の組の集合を Q_1 とし，$s_1, \cdots, s_{i-1}, s_{i+1}, \cdots, s_m$ すべてを真にする定数の組の集合を Q_2 とすると，$Q_2 \supseteq Q_1$ であるから，Q_2 は s_i の例外を含む．従って exc の定義から，

$$exc\text{-}s_i \leftarrow s_1, \cdots, s_{i-1}, s_{i+1}, \cdots, s_m$$

となり，定義2より次の関係代数式を得る．

$$exc\text{-}S_i = Rel(S_1, \cdots, S_{i-1}, S_{i+1}, \cdots, S_m) \qquad \square$$

(定理2の証明) $j=i+1$ なら，$exc^j\text{-}R$ は $exc^i\text{-}R$ の例外項であるから，定義6より，

$$stratum[exc^j\text{-}R] < stratum[exc^i\text{-}R]$$

は明らか．

$j \neq i+1$ なら，$exc^{i+1}\text{-}R, exc^{i+2}\text{-}R, \cdots, exc^{i+k}\text{-}R$ という k 個 $(1 \leq k)$ の関係が存在する．ただし，$exc^{i+k}\text{-}R = exc^{j-1}\text{-}R$ である．これらの関係は順次，次数の大きいものが小さいものの例外項となるので，

$$stratum[exc^j\text{-}R] < stratum[exc^{i+k}\text{-}R] < \cdots < stratum[exc^{i+1}\text{-}R] < stratum[exc^i\text{-}R]$$

となり，従って，

$$stratum[exc^j\text{-}R] < stratum[exc^i\text{-}R] \qquad \square$$

(定理3の証明) ある依存グラフにおいて不完全ルールの帰結部を表す関係 $exc^n\text{-}R$ $(0 \leq n)$ に対応するノードから，その不完全ルールの例外項 $exc^{n+1}\text{-}R$ に対応するノードへの経路が存在するとする．定義6よりこれらの関係の階層は次のような関係をもつ．

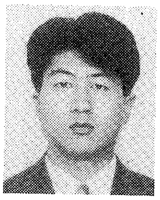
$$stratum[exc^n\text{-}R(\alpha)] \leq stratum[exc^{n+1}\text{-}R(\alpha)]$$

ところが，$exc^{n+1}\text{-}R$ は不完全ルールの例外項であるから，定義6より，

$$stratum[exc^{n+1}\text{-}R(\alpha)] < stratum[exc^n\text{-}R(\alpha)]$$

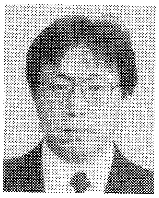
でなければならない．従って，定義6により階層化をすると，ステップ2が無限に繰り返され，階層が無限になる．このような依存グラフをもつ式集合は定義7より階層化可能ではない．従って，定義6により階層化可能な式集合に対する依存グラフには，不完全ルールの帰結部を表す関係に対応するノードからその不完全ルールの例外項を表す関係に対応するノードへの経路は存在しない． □

(平成6年10月24日受付)
(平成7年 3 月13日採録)



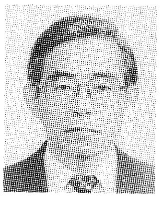
大原　剛三（学生会員）

昭和46年生．平成7年大阪大学大学院前期課程修了．現在同大学院後期課程在学中．非単調推論，例外を伴った知識ベース，知識表現に関する研究に従事．人工知能学会会員．



馬場口　登（正会員）

昭和32年生．昭和54年大阪大学工学部通信工学科卒業．昭和56年同大学院前期課程修了．昭和57年愛媛大学工学部助手．大阪大学工学部助手，講師を経て，現在大阪大学産業科学研究所助教授．工学博士．人工知能，パターン認識，画像処理の研究に従事．IEEE，電子情報通信学会，人工知能学会各会員．



北橋　忠宏（正会員）

昭和14年生．昭和37年大阪大学工学部通信工学科卒業．昭和43年同大学院博士課程修了．同年大阪大学基礎工学部助手．同助教授，豊橋技術科学大学助教授，教授を経て，昭和61年大阪大学産業科学研究所教授．工学博士．3次元物体認識のための視覚システム，自然言語処理，学習・推論機構に関する研究に従事．電子情報通信学会，IEEE，日本認知科学会，AVIRG，計量国語学会，人工知能学会各会員．