例外に着目した知識ベースの変換メカニズムの形式化

2000年1月

桂田　浩一

例外に着目した知識ベースの変換メカニズムの形式化

2000年 1月

桂田　浩一

内容梗概

本論文は，筆者が大阪大学大学院基礎工学研究科（情報数理系専攻ソフトウエア科学分野）後期課程在学中に行った例外に着目した知識ベースの変換メカニズムの形式化に関する研究をまとめたものである．本研究は非単調論理の枠組において，通常ルール（常に結論を導くルール）からデフォルトルール（例外に関して結論を導かないルール）への変換を代表とする幾つかの変換によって，矛盾する知識ベース，および簡潔でない知識ベースを，無矛盾で簡潔な知識ベースに変換する事を目標としている．無矛盾で簡潔な知識ベースへの変換技術は，動的に変更される知識ベースでは必要不可欠なものであり，従来研究においても一階論理や命題論理などの古典論理を知識体系としたものが提案されている．しかしながら，古典論理では例外を適切に表現し，処理することが困難であるため，無矛盾で簡潔な知識ベースへの変換には限界がある．以上の背景から，本論文では例外を適切に処理できる非単調論理に基づいた以下の二つの知識ベースの変換法を提案した．

まず知識ベースの矛盾を解消するための，例外に着目した知識ベース変換法を提案した．古典論理の枠組では幾つかのルールおよび事実を削除することによって矛盾を解消することが一般的であるが，この方法では矛盾に無関係な推論結果までが導かれなくなるという問題点がある．そこで本論文の3章では，通常ルールをデフォルトルールに変換する知識コンバージョンによって，矛盾を引き起こす例外に関する結論だけを導かなくして矛盾を解消する方法を提案した．この知識コンバージョンの対象となるルールを検出するために，まず整合状態という知識ベースの状態を定義した．整合状態では，知識ベースから導かれる推論結果のうち矛盾に関与する全ての推論結果が導かれなくなる．本論文では知識ベースを整合状態にするアルゴリズムを示し，真か偽か疑わしい全ての推論結果が知識ベースから導かれないような変換を実現した．

続いて知識の例外に関する観点変更によって簡潔な知識ベースに変換する手法を提案した．例外に関する観点とはあるクラスに関して何を例外と捉えるかという，例外の捉え方をいう．捉え方を変更して例外の少ない知識にすることによって，例外に関連するルールが削減されるため，簡潔な知識ベースへの変換が可能となる．本論文では4章において基本的な6つの変換操作を用いた観点変更アルゴリズムを示した．ア

ルゴリズムで用いられる基本的な6つの変換操作は，知識ベースから得られる推論結果を変化させないため，変換前に得られていた推論結果が知識ベースの変換後にも失われない．この点において，基本的な6つの変換操作は観点変更にとって望ましいといえる．このように形式化した観点変更アルゴリズムを計算機により実装し，観点変更によるルール数削減の実験を行った．この結果，例外が多いほどルール数が減少し，観点変更が有効であるという知見が得られた．

以上のように観点変更において提案した6つの変換操作は推論結果を変えない知識ベース変換の最も基本的な操作であるため，観点変更に限らず，非単調論理を知識体系として用いた様々な知識ベース変換において用いることができる．また本論文で提案した両知識ベース変換法は，これまで検討されてこなかった例外に着目したものであるため，例外の多い現実の知識を知識ベースにおいて取り扱うための一助に成り得ると考えられる．

関連発表論文

A. 学会論文

1. 桂田，大原，馬場口，北橋：不完全知識の例外に関する観点変更による知識ベース再構成，人工知能学会誌, Vol.14, No.3, pp.485–pp.494 (1999).
2. 桂田，小山，大原，馬場口，北橋：知識の矛盾に基づく知識コンバージョン，人工知能学会誌, (投稿中).

B. 国際会議

1. K. Katsurada, M. Koyama, K. Ohara, N. Babaguchi, T. Kitahashi : Solving Contradiction in Knowledge-Base without Interaction, Proc. 1998 IEEE International Conference on Systems, Man, and Cybernetics, pp.1546–1551 (1998).
2. K. Katsurada, K. Ohara, N. Babaguchi, T. Kitahashi : On Operations for Reconstructing the Complete/Incomplete Knowledge, Proc. The 10th European-Japanese Conference on Information Modelling and Knowledge Bases, (投稿中).
3. K. Katsurada, M. Koyama, K. Ohara, N. Babaguchi, T. Kitahashi : Converting Ordinary Rules into Default Rules Based on Contradiction of Knowledge Base, Proc. The 10th European-Japanese Conference on Information Modelling and Knowledge Bases, (投稿中).

C. 学会研究会

1. 桂田，大原，馬場口，北橋：完全・不完全ルールの使用頻度を考慮した知識洗練化，情報処理学会研究報告 97-AI-107, pp.1–8 (1997).
2. 桂田，大原，馬場口，北橋：完全/不完全知識の再構成のための知識コンバージョン，人工知能学会研究会資料 SIG-FAI-9702, pp.73–78 (1997).
3. 小山，桂田，大原，馬場口，北橋：文書自動分類における分類誤りを契機とした例外処理とその実験的評価，人工知能学会研究会資料 第46回知識ベースシステム研究会 (2000).

D. 学会講演発表

1. 桂田，大原，馬場口，北橋：完全知識の不完全知識への変換，1995年電子情報通信学会総合大会論文集，pp.245 (1995).
2. 馬場口，大原，桂田，北橋：矛盾を契機とする知識の質的転化，1995年度人工知能学会全国大会論文集，pp.53–56 (1995).
3. 桂田，大原，馬場口，北橋：ルール数の変化を考慮した完全・不完全知識の洗練化，1996年度人工知能学会全国大会論文集，pp.309–312 (1996).
4. 桂田，大原，馬場口，北橋：知識コンバージョンと完全・不完全知識の洗練化の統合について，情報処理学会第59回全国大会講演論文集 (分冊2)，pp.229–230 (1996).
5. 小山，桂田，大原，馬場口，北橋：完全/不完全ルールの属性変換による矛盾解消のための非対話的手法，1998年度人工知能学会全国大会論文集，pp.28–31 (1998).
6. 小山，桂田，大原，馬場口，北橋：Web上のニュース記事の分類における知識コンバージョンを用いた例外処理，情報処理学会第53回全国大会講演論文集 (分冊2)，pp.81–82 (1999).

目次

図目次

表 目 次

第 1 章

緒論

近年，現実に存在するような複雑な問題を，計算機によって解決する機会が増えている．こうした問題を解決するには高度な問題解決能力が必要とされるため，知識ベースシステムのように，明示的にシステムに与えられている以上のことを推論によって導き，複雑な問題を解決できるようなシステムが非常に有効であるといえる．知識ベースシステムは一般的に規則を保持する知識ベース，および演繹操作を規則に適用して推論結果を求める推論機構から構成される．知識ベース中の規則は一階論理のようなルール形式のものや，意味ネットワークのようなグラフ表現のものなどシステムによって様々であるが，中でもルール形式のものは知識ベースの変更が容易であるため，動的に変化し得る知識ベースシステムにおいては利用価値が高いといえる．

こうした知識ベースシステムの実用化は古くから検討されており，その歴史は1965年に発足した化学構造式同定システムDENDRALプロジェクト [Buchanan 78] に端を発する．当初は知識ベースおよび推論機構といった概念がなく，知識ベースの変更が非常に難しい状況であった．1970年代に入ると形式化が進められ，知識ベース，推論機構といった概念が確立されて，知識ベースシステムの古典的形態と呼ばれるものが完成した．このようなシステムの代表例が医療システムのMYCIN [Shortliffe 76] である．MYCINはプロダクションルールというif-thenルールを知識ベースに持ち，推論過程に関する説明機能などが組み込まれたシステムである．さらに1980年代に入ると知識ベースシステムは医療診断，機器の故障診断システムとして産業界でも活用されるようになり，人工知能研究の社会的認知に大いなる足跡を残した．

このように知識ベースシステムが実世界で利用されるに従って，知識ベースシステムを一層実用的なものにするために解決すべき問題点が明らかになってきた．その一

つが知識ベース変更に伴って発生する数々の問題である．例えば変更された知識ベース中のルールが誤っていた場合には知識ベースが矛盾することもあり，またルールに無駄があった場合には不要なルールが生じ得る．この結果，推論効率の悪化，推論性能の悪化，知識ベースの肥大化，推論結果の矛盾といった問題が現れる．こうした問題を解決するためにはシステム自体が自己の知識ベースのルールを適宜，削除，変形，あるいは追加すること（以下ではこれらの操作を総称して**知識ベースの変換**と呼ぶ）を検討しなければならない．

上述のような変換を目的として，様々な分野で検討がなされている．推論効率を向上させることを目的としては，知識コンパイル (Knowledge Compilation) [Selman 91, del Val 95] や論理プログラムの変換 (Transformation of Logic Programs) [Pettorossi 94, 赤間 97] が提案されている．また推論性能の向上としては知識洗練 (Knowledge Refinement) [A.Ginsberg 85, Richards 95, Antoniou 96] に関する研究が盛んに行われている．さらに知識ベースの肥大化に対処するために簡潔な知識ベースへの変換方法 [A.Ginsberg 88, Hammer 93, Schmolze 97] が提案されており，矛盾解消の手法としては信念翻意 (Belief Revision) の研究として様々な手法 [Alchourrón 85, Dadal 88, Katsuno 91, Baral 94, Witteveen 95, Antoniou 98] が提案されている．これらの諸手法のほとんどでは，知識体系として一階述語論理や命題論理などの古典論理を用い，その枠組においてそれぞれの知識ベース変換を定式化している．

しかしながら上述の諸手法のうち，特に矛盾解消に関しては，古典論理を知識体系とした場合に問題が発生する．それは古典論理を知識体系として用いたシステムにおいて矛盾が発生した場合には，ルールおよび事実を削除せずに矛盾を解消することが困難であるという問題である．なぜなら古典論理では矛盾の原因となったルールの例外に関する推論結果だけを適切に排除するには，例外に関する膨大な記述が必要となるからである．このため例外に関する推論結果だけを排除することは実質的に不可能であり，矛盾を解消する最も現実的な方法はルールを削除することとなる．しかしながらルールを削除した場合には，削除の前にそのルールから導かれていた，矛盾と無関係な推論結果までもが導かれなくなる．これは矛盾解消法として望ましいといえない．

そこで本論文では，非単調論理という，例外を適切に処理できる知識体系を導入して矛盾を解消する方法を検討する．非単調論理は一階述語論理を拡張した論理体系で

あり，その一つであるデフォルト論理 [Reiter 80] ではデフォルトルールという，例外に関して結論を導かないルールを導入することで表現力を高めている．本論文ではこのデフォルト論理に基づいた例外に着目した知識ベースの変換法として，ホーン節に相当する通常ルール（常に結論を導くルール）を，デフォルトルールに変換する操作（本論文では知識コンバージョン [馬場口 95, 馬場口 96] と呼ぶ）によって矛盾を解消する手法を検討する．この操作によって例外に関する誤った推論結果が導かれなくなるため，その誤った推論結果によって生じた矛盾を解消できる．またこの操作によって矛盾と無関係な推論結果は変化しないため，矛盾解消法としては最も望ましいといえる．

さらに，非単調論理の知識体系を用いたことにより可能となる，例外に関する観点変更による簡潔な知識ベースの獲得法を検討する．例外に関する観点とは，あるクラスの要素が持つ性質について，何を代表的な性質と捉えるか，また何を例外的な性質と捉えるか，といった例外の捉え方をいう．観点によって，例外的な性質が異なるため，例外的な性質を表すルールも異なる．このためルールの総数も異なる．したがって例外に対する観点を変えて例外の少ないデフォルトルールで性質を表すことができれば，例外的な性質を表すルールを削減できる．結果として簡潔な知識ベースを得ることができる．

以下に本論文の構成を示す.

まず本論文の 2 章ではこれまでに提案されている非単調論理のうち特にデフォルト論理について，その概要，および例外との関係を述べる．また非単調論理を知識体系として用いた知識ベース変換の従来手法として，矛盾に基づく知識ベースの変換を目的とした信念翻意 [Antoniou 98]，および簡潔な知識ベースへの変換を目的とした冗長ルールの削除法 [Schmolze 97] のそれぞれを概説する．さらに従来手法が例外に着目していないことを指摘した上で，例外に着目することによって先に述べた矛盾解消のための知識コンバージョン，および簡潔な知識ベースへの変換のための観点変更が可能となることを具体例を用いて述べる．

続いて 3 章では，2 章における考察に基づき，例外に着目した，矛盾解消のための知識ベース変換法として，矛盾に基づく知識コンバージョン [桂田 95, Katsurada 98, 小山 98, 小山 99, 小山 00, Katsurada 00b, 桂田 00] を提案する．3 章ではまずこの知識コンバージョンの対象となるルールを知識ベースに含まれるルール全体から絞り込

むために，矛盾が導かれる過程で用いられたルール集合を検出する方法を述べる．さらにその中から知識コンバージョンの対象となるルールを同定するために，整合状態 [Katsurada 00b, 桂田 00] という知識ベースの状態を定め，整合状態になるようにルールを変換するアルゴリズムを示す．またこの整合状態の性質を解析し，関連研究と比較する．

4 章では簡潔な知識ベースへの変換方法として，例外に関する観点変更 [桂田 96a, 桂田 96b, 桂田 97a, 桂田 97b, 桂田 99, Katsurada 00a] を検討する．観点変更の目的はルール数の削減であるため，この変更によって知識ベースから得られる推論結果が変化すべきではない．そこで推論結果を変化させないような，デフォルトルールの特殊化，一般化や知識コンバージョンといった 6 つの基本操作 [桂田 99, Katsurada 00a] を形式化し，それらを複合的に組み合わせた観点変更アルゴリズム [桂田 99] を示す．さらに実験によりその特性を示す．

最後に 5 章では本研究で得られた成果を総括するとともに，今後の課題について言及する．

第 2 章

非単調論理に基づいた知識ベースの変換

2.1 緒言

様々な目的のために知識ベースの変換法が提案されているが，その多くは古典論理を知識体系として用いている．こうした古典論理に基づいた手法のうち，特に矛盾解消を目的とした手法では，一般的に矛盾を解消するためにルールを削除している．しかしながらこの削除によって矛盾に無関係な推論結果までが失われるため，これは望ましい矛盾解消法であるとはいえない．これに対して非単調論理の知識体系の下で例外に着目した場合には，1 章で述べたように，例外に関する推論結果だけを適切に排除するようなルールへ変換できるため，矛盾に無関係な推論結果を失うことなく矛盾が解消できる．また非単調論理の知識体系において例外に着目した場合には，例外の捉え方によってルール数の削減が実現でき，簡潔な知識ベースへの変換も可能となる．

本章ではまずこれまでに提案されている非単調論理に関する概要と，本論文で利用するデフォルト論理 [Reiter 80] という非単調論理について簡単に説明し，例外との関係を論じる．続いて非単調論理に基づいた知識ベースの変換方法として提案されている信念翻意 [Antoniou 98]，および冗長なルールの削除法 [Schmolze 97] を概説する．さらにこれらの手法は例外に着目した手法ではないことを示した上で，例外に着目することによって可能となる知識ベースの変換法を例示する．

2.2 非単調論理の概要

古典論理を用いた知識体系では，知識ベースに新たな事例やルールが加わった場合に，知識ベースから導かれる推論結果は単調に増加する．すなわち追加の前に導かれていた推論結果は全く失われることなく，新たに加わった事例やルールに関する推論結果が追加的に導かれるようになる．このため古典論理は単調論理と呼ばれる．

一方，現実世界を考えた場合，新たな事実等の発見によって，これまで正しいと思われていたことが取り消されることは頻繁に起こり得る．例えば Tweety という鳥は飛ぶであろうと推論されていたのに対して，Tweety が飛べないことが分かった場合，飛ぶであろうという推論は取り消される．こうした現実に行われるような推論を形式化するために，古典論理を拡張した論理体系として非単調論理が検討されてきた．非単調論理では，知識ベースに新たな事例やルールが加わった場合に，必ずしも以前に導かれていた推論結果が得られるとは限らないという点において，非単調性を有する．

これまでに提案されている非単調論理としては，デフォルト論理 (Default Logic) [Reiter 80]，自己認識論理 (Autoepistemic Logic) [Moore 85, Konolige 88]，サーカムスクリプション (Circumscription) [McCarthy 80]，CWA(Closed World Assumption) [Reiter 78] を用いた論理など，様々な論理体系が提案されているが，以下ではこのうちデフォルト論理を例に挙げ，その概要，および例外との関係を述べる．

デフォルト論理 [Reiter 80] では一階論理式に加えて次のような推論規則（デフォルトルール）δ を用いることによって知識を表現する．

$$\delta = \frac{p : \mathrm{M}j_1, \ldots, \mathrm{M}j_n}{c}$$

ここで p，$j_1, \ldots, j_n$，c は一階論理式であり，上式は「p が成り立ち，$\neg j_1, \ldots \neg j_n$ が証明されなければ，c を推論する」と解釈する．

デフォルト論理では p，$j_1, \ldots, j_n$，c をそれぞれデフォルトルール δ の前提 (prerequisite)，根拠 (justification)，結論 (consequent) という．また，一階論理式の集合を W，デフォルトルールの集合を D とするとき，W と D の組 (W, D) をデフォルト理論といい，デフォルト理論 $\mathcal{T} = (W, D)$ から導かれる一階論理式の集合 $E(\mathcal{T})$ を拡張 (extension) という．拡張 $E(\mathcal{T})$ は $\mathcal{T}$ から導かれる推論結果を表し，以下のように定義される．ただし定義中の $Th(E_i)$ は一階論理式集合 E_i の論理的閉包を表す．

定義 2.1 (拡張 [Reiter 80]) デフォルト理論 $\mathcal{T} = (W, D)$ に対して一階論理式の集合 E_i $(i \geq 0)$ を次のように定義する．

$E_0 = W$

$E_{i+1} = Th(E_i) \cup \{c \mid \frac{p:Mj_1,\ldots,Mj_n}{c} \in D,$ ただし $p \in E_i$ かつ $\neg j_1, \ldots, \neg j_n \notin E\}$

$E = \bigcup_{i=0}^{\infty} E_i$ であるとき，またそのとき限り E は $\mathcal{T}$ の拡張 $E(\mathcal{T})$ である． □

例えば次のような知識をデフォルト論理で表した場合，デフォルト理論 $\mathcal{T}_1 = (W_1, D_1)$ は以下のようになる．ただし本論文では含意記号を "←" で表す．

知識： 「Tweety は鳥である」
「Tom は鳥である」
「Tweety は怪我をしている」
「鳥であり，かつ怪我をしているならば飛ばない」
「鳥ならば通常飛ぶ」

$$W_1 = \left\{ \begin{array}{l} \mathrm{bird(Tweety)} \\ \mathrm{bird(Tom)} \\ \mathrm{injured(Tweety)} \\ \neg\mathrm{fly}(x) \leftarrow \mathrm{injured}(x) \wedge \mathrm{bird}(x) \end{array} \right\}$$

$$D_1 = \left\{ \frac{\mathrm{bird}(x):\mathrm{M\ fly}(x)}{\mathrm{fly}(x)} \right\}$$

このデフォルト理論において $E = Th(W_1 \cup \{\mathrm{fly(Tom)}\})$ としたとき，定義 2.1に従うと，

$$E_0 = W_1$$
$$E_1 = Th(W_1) \cup \{\mathrm{fly(Tom)}\}$$
$$E_2 = Th(Th(W_1) \cup \{\mathrm{fly(Tom)}\}) \cup \phi = Th(W_1 \cup \{\mathrm{fly(Tom)}\})$$
$$E_3 = Th(Th(W_1 \cup \{\mathrm{fly(Tom)}\})) \cup \phi = Th(W_1 \cup \{\mathrm{fly(Tom)}\})$$
$$\vdots$$

となり，

$$\textstyle\bigcup_{i=0}^{\infty} E_i = Th(W_1 \cup \{\mathrm{fly(Tom)}\}) = E$$

が成り立つため，$\mathcal{T}_1$の拡張は$E(\mathcal{T}_1) = Th(W_1 \cup \{\mathrm{fly}(\mathrm{Tom})\})$となる．したがってデフォルト理論 $\mathcal{T}_1$からは $\{\mathrm{bird}(\mathrm{Tweety}), \mathrm{bird}(\mathrm{Tom}), \mathrm{injured}(\mathrm{Tweety}), \neg\mathrm{fly}(\mathrm{Tweety}), \mathrm{fly}(\mathrm{Tom})\}$ の基礎原子式集合が推論結果として導かれる．これら以外の基礎原子式については推論結果として得られなかったと見做す．また，推論結果に関する信頼度のようなものは設けず，デフォルトルールの例外であるなしに関わらず，拡張に含まれるか否かのみで推論結果に含まれるか否かを判断する．なお，ある拡張に論理式 a と$\neg a$ が同時に含まれる場合，その拡張を導くデフォルト理論は矛盾すると呼ぶ．

定義 2.1の条件を満たす他の Eは存在しないため，$\mathcal{T}_1$ の拡張は$Th(W_1 \cup \{\mathrm{fly}(\mathrm{Tom})\})$のみとなる．ただしデフォルト理論によっては拡張が複数になる場合や，あるいは拡張が存在しない場合がある事が知られている．例えば次のようなデフォルト理論 $\mathcal{T}_2 = (W_2, D_2)$ を想定する．

$$W_2 = \left\{ \begin{array}{l} \mathrm{bird}(\mathrm{Tweety}) \\ \mathrm{injured}(\mathrm{Tweety}) \end{array} \right\}$$
$$D_2 = \left\{ \begin{array}{l} \frac{\mathrm{bird}(x):\mathrm{M}\ \mathrm{fly}(x)}{\mathrm{fly}(x)} \\ \frac{\mathrm{injured}(x)\wedge\mathrm{bird}(x):\mathrm{M}\ \neg\mathrm{fly}(x)}{\neg\mathrm{fly}(x)} \end{array} \right\}$$

このデフォルト理論では $E1 = Th(W_2 \cup \{\mathrm{fly}(\mathrm{Tweety})\})$，および $E2 = Th(W_2 \cup \{\neg\mathrm{fly}(\mathrm{Tweety})\})$ のいずれもが定義 2.1の Eの条件を満たす．したがって $E1$ および $E2$ はいずれも $\mathcal{T}_2$の拡張である．このように複数の拡張が求まる理由は，定義 2.1に従うと D_2に含まれる二つのデフォルトルールを同時に用いることができず，どちらを用いるかによって異なる拡張になるからである．このように複数のデフォルトルールが存在する場合には，その相互作用によって拡張が一つに定まらない場合がある．

本論文では「ルールにあてはまらないもの」を「例外」と考える．デフォルト理論 $\mathcal{T} = (W, D)$ において形式的に議論すると，Dに含まれるデフォルトルール$\delta = \frac{p:\mathrm{M}j_1,\ldots,\mathrm{M}j_n}{c}$に関して，ある基礎代入 θ について，$p\theta$ が拡張に含まれるにも関わらず，$\neg j_1\theta, \ldots, \neg j_n\theta$のいずれかが拡張に含まれるために $c\theta$ が拡張に含まれないときに基礎代入 θ が表す個体をδ の例外と呼ぶ．先のデフォルト理論 $\mathcal{T}_1 = (W_1, D_1)$ では，Tweety という個体について bird(Tweety) が拡張に含まれるにも関わらず$\neg$fly(Tweety) が拡張に含まれるために fly(Tweety) が拡張に含まれない．したがって Tweety はデフォルトルール$\frac{\mathrm{bird}(x):\mathrm{M}\ \mathrm{fly}(x)}{\mathrm{fly}(x)}$の例外である．またデフォルト理論 $\mathcal{T}_2 = (W_2, D_2)$ では，拡張 $E1$ については Tweety はデフォルトルール$\frac{\mathrm{injured}(x)\wedge\mathrm{bird}(x):\mathrm{M}\ \neg\mathrm{fly}(x)}{\neg\mathrm{fly}(x)}$の例外となり，拡張 $E2$

についてはデフォルトルール$\frac{\mathrm{bird}(x):\mathrm{M\ fly}(x)}{\mathrm{fly}(x)}$の例外となる.

2.3 非単調論理に基づいた知識ベース変換の手法

これまでに提案されている知識ベースの変換法の多くは知識体系として古典論理を用いているが，デフォルト論理に代表される非単調論理を用いている手法も幾つか提案されている．本節ではこのうちデフォルト論理の知識体系を用いた矛盾解消のための信念翻意の手法 [Antoniou 98]，および CWA を用いた論理を知識体系として用いた冗長なルールの削除法 [Schmolze 97] について概略を示す.

2.3.1 非単調論理に基づいた信念翻意

知識ベースが矛盾するとき，論理的には任意の論理式が導かれることになるため，誤った推論結果が導かれないようにするためにも，早急に矛盾を解消しなければならない．そこで信念翻意や知識ベース更新の諸研究 [Alchourrón 85, Dadal 88, Katsuno 91, Baral 94, Witteveen 95, Antoniou 98] では矛盾に基づいた知識ベースの変換方法が検討されてきた.

信念翻意における信念とは知識ベースのモデルのことを指し，これが翻意するとは新たに加わった論理式によってそれまでのモデルの一部が翻されることを意味する．モデルが翻されるためには知識ベースが変換されることが必要であり，実装的には知識ベース変換の一手法であるといえる．こうした信念翻意が満たすべき条件として，一般的に次のようなものが用いられている．ただし，ここでは知識ベースが古典論理からなり，知識ベース W に新たな論理式 ψ が加わったことを契機として信念が翻意され，知識ベース W が W' に変換されたとする.

(1) W' は矛盾しない.

(2) W' のモデルに ψ が含まれる.

(3) W と W' のモデルの差を極小にする.

条件 (1) は知識ベースの変換後に矛盾が解消されることを意味し，また条件 (2) は新たに加わった論理式 ψ は必ず正しく，変換後においても論理的帰結として導かれる

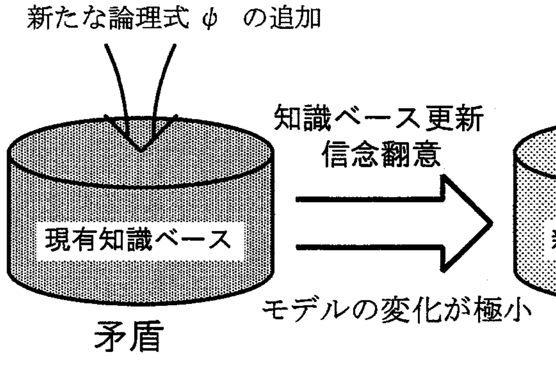


図 2.1 信念翻意と知識ベースの更新

ことを意味する．さらに条件 (3) はできるだけ推論結果が変化しないように知識ベースが変換されることを表す．信念本意を概念的に表したものを図 2.1 に示す．

ここで信念翻意をデフォルト論理の体系において実現した一つの手法 [Antoniou 98] について概説する．Antoniou の手法ではデフォルト論理の知識体系を用いた信念翻意の操作として数種類を挙げているが，ここではその一つを示す．ただし次の一連の操作ではデフォルト理論 $\mathcal{T} = (W, D)$ に新たな論理式 ψ が加わり，デフォルト理論 $\mathcal{T}$ を $\mathcal{T}' = (W', D')$ に変換したとする．

(I) $\neg\psi \notin Th(W')$ となるように W から論理式を取り除く．

(II) デフォルトルール $\delta' = \frac{true:M\psi\wedge p}{\psi\wedge p}$ を D に追加する．ただし p は新たな原子式．

(III) 全ての $\delta \in D$ の根拠および結論に $\neg p$ を加える．ただし根拠には $M\ \neg p$ の形で，結論には連言の形で加える．

(IV) 操作 (I) において取り除いた論理式の集合を W_{del} とするとき，D に $\{\frac{true:M\neg p}{\neg p\wedge w} \mid w \in W_{del}\}$ を追加する．

操作 (I) によって知識ベースの変換後には矛盾が解消され，操作 (II)〜(IV) によって新たに加わった論理式 ψ が少なくとも一つ以上の拡張に含まれるようになる．また操作 (IV) はできるだけ拡張が変化しないように知識ベースを変換するためのものである．

このAntoniouの手法では ψ が T'の全ての拡張に含まれるべきであるとは考えておらず，手法の適用後のデフォルト理論の拡張は，Tのそれぞれの拡張に原子式$\neg p$を加えたもの，およびTの拡張から$\neg\psi$を取り除き，pとψを加えたものとなる．

ここで上の操作(I)〜(IV)による信念翻意の例を示す．次のようなデフォルト理論$T_3 = (W_3, D_3)$を想定する．

$$W_3 = \left\{ \begin{array}{l} \text{penguin(Tweety)} \\ \neg\text{fly(Tweety)} \\ \text{fly}(x) \leftarrow \text{bird}(x) \end{array} \right\}$$

$$D_3 = \phi$$

このデフォルト理論にbird(Tweety)が加わったことを契機として信念翻意が実行された場合を考える．まず操作(I)によって$\neg$bird(Tweety)を導く論理式がW_3から削除される．ここではfly$(x) \leftarrow$ bird(x)が削除されたとする．続いて操作(II)によってDにデフォルト$\delta' = \frac{true:\text{M bird}(x)\wedge p}{\text{bird}(x)\wedge p}$が追加される．$T_3$にはデフォルトルールが存在しないため操作(III)では何も行われないが，続く操作(IV)では$\delta'' = \frac{true:\text{M}\neg p}{\neg p\wedge(\text{fly}(x)\leftarrow\text{bird}(x))}$が追加される．最終的に次のデフォルト理論$T_3' = (W_3', D_3')$へ変換される．

$$W_3' = \left\{ \begin{array}{l} \text{penguin(Tweety)} \\ \neg\text{fly(Tweety)} \end{array} \right\}$$

$$D_3' = \left\{ \begin{array}{l} \frac{true:\text{M bird(Tweety)}\wedge p}{\text{bird(Tweety)}\wedge p} \\ \frac{true:\text{M}\neg p}{\neg p\wedge(\text{fly}(x)\leftarrow\text{bird}(x))} \end{array} \right\}$$

このデフォルト理論は以下の二つの拡張を持つ．

$$E1(T_3') = Th(\{\text{penguin(Tweety)}, \neg\text{fly(Tweety)}, \text{bird(Tweety)}, p\})$$
$$E2(T_3') = Th(\{\text{penguin(Tweety)}, \neg\text{fly(Tweety)}, \text{fly}(x) \leftarrow \text{bird}(x), \neg p\})$$

このように変換前のデフォルト理論T_3の拡張に含まれる$\neg$ bird(Tweety)を取り除き，bird(Tweety)とpを加えた拡張$E1$と，T_3の拡張に$\neg p$を加えた拡張$E2$が求まる．

2.3.2 非単調論理に基づいた冗長なルールの削除

現実世界を取り扱う知識ベースは非常に大きくなることが予想される．こうした知識ベースにおいては，記憶容量や推論効率の観点から，できるだけ簡潔な知識ベースに変換してルール数を少なくすることが望まれる．こうした背景から図2.2のような冗

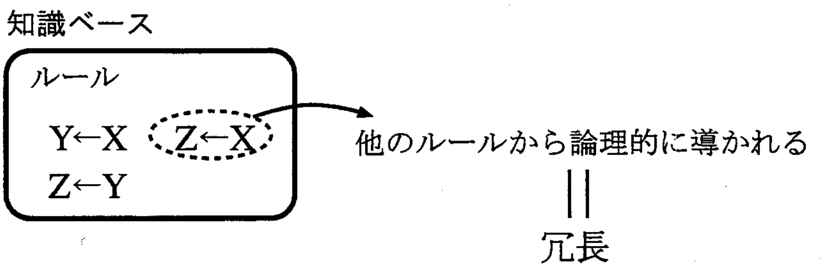


図 2.2 冗長なルール

長なルールを削除することによって簡潔な知識ベースに変換する手法が幾つか提案されている [A.Ginsberg 88, Hammer 93, Schmolze 97].

これらの手法で共通して用いられている冗長性の定義は，ルールの包含関係に基づくものである．ここでいう包含関係に基づいた冗長なルールとは以下のように定義される．

定義 2.2 (冗長なルール) 知識ベースを T，ルールを rとするとき，次の条件が満たされるならば rは冗長なルールである．

$$T - \{r\} \vdash r \qquad \square$$

上に定義されるような冗長なルールを検出し，それを削除する諸手法のうち，ここでは Schmolze の手法 [Schmolze 97] を取り上げ，実際に冗長なルールがどのように削除されるかを例示する．Schmolze の手法では知識体系として次のような CWA を用いたプロダクションルールが使われている．

$$r = C_1, \ldots, C_n \Rightarrow A_1, \ldots, A_m$$

このルールは WM(Working Memory) と呼ばれる原子式の集合に作用するルールであり，作用するための条件が $C_1, \ldots, C_n$，作用が $A_1, \ldots, A_m$である．$C_1, \ldots, C_n$はそれぞれ通常原子式か，あるいは否定原子式 (negated atom) を表す．また $A_1, \ldots, A_m$はそれぞれ $REMOVE\ A$ あるいは $ADD\ A$ であり，それぞれ WM Wから原子式 A を取り除く，あるいは Wに原子式 A を追加するという意味を持つ．ただし $REMOVE\ A$ が存在するときには $C_i = A$ となる通常原子式 C_iが存在するとする．

いま上のようなルール r の C_i について，代入 σ が存在し，C_i が通常原子式のとき $C_i\sigma \in W$ が成り立ち，C_i が否定原子式のとき $C_i\sigma \notin W$ が成り立つならば r は適合するという．このとき $ADD\ A$，および $REMOVE\ A$ の作用によって W から WM W' が生成されることを r と σ による具象化 (instantiation) という．

ここで $C_1,\ldots,C_n$ のうち通常原子式のみからなる集合を $\{C'_1,\ldots,C'_i,C'_{i+1},\ldots,C'_k\}$，$A_1,\ldots,A_m$ のうち ADD のみからなる集合を $\{A'_1,\ldots,A'_l\}$ とする．ただし $\{C'_1,\ldots,C'_i\}$ は $A_1,\ldots,A_m$ 中の $REMOVE$ に対応しないもの，$\{C'_{i+1},\ldots,C'_k\}$ は $A_1,\ldots,A_m$ 中の $REMOVE$ に対応するものとする．このとき Schmolze の手法では WM 中の $\{C'_1,\ldots,C'_k\}$ を $\{A'_1,\ldots,A'_l,C'_1,\ldots,C'_i\}$ に更新する．この具象化を r 以外のルールを用いて連続して行い，その結果 r による具象化と同様に WM を更新できるならば，r が冗長であると判断する．

この冗長性を調べるために具象化の連鎖に関して記号を定める．P，および Q を WM，R をルールの集合とするとき，R 中の幾つかのルールによる具象化によって P を Q に更新できるとき，

$$P \longrightarrow_R Q$$

と表す．特に WM $P_0,\ldots,P_n$ が存在し，

$$P = P_0 \longrightarrow P_1 \longrightarrow \ldots \longrightarrow P_n = Q$$

であるとき $P \longrightarrow^* Q$ と表す．

以上の定義を用いて，あるルール $r = C_1,\ldots,C_n \Rightarrow A_1,\ldots,A_m$ が，ルール集合 R において冗長か否かを次の条件によって判定する．ただし $R' = R-\{r\}$ とし，$C_1,\ldots,C_n$ のうち通常原子式のみからなる集合を $\{C'_1,\ldots,C'_i,C'_{i+1},\ldots,C'_k\}$，$A_1,\ldots,A_m$ のうち ADD のみからなる集合を $\{A'_1,\ldots,A'_l\}$ とする．また，$\{C'_1,\ldots,C'_i\}$ は $A_1,\ldots,A_m$ 中の $REMOVE$ に対応しないもの，$\{C'_{i+1},\ldots,C'_k\}$ は $A_1,\ldots,A_m$ 中の $REMOVE$ に対応するものとする．

$$\{C'_1,\ldots,C'_k\} \longrightarrow^*_{R'} \{A'_1,\ldots,A'_l,C'_1,\ldots,C'_i\}$$

上式が成り立てば r は R' において冗長であると判断する．こうしたルールを削除す

ることによって簡潔な知識ベースへの変換が可能となる.

2.4 例外に着目した知識ベースの変換

節 2.3.1 で示したデフォルト理論 $\mathcal{T}_3$に対して bird(Tweety) を追加したデフォルト理論 $\mathcal{T}_4 = (W_4, D_4)$ を想定する.

$$W_4 = \left\{ \begin{array}{l} \text{penguin(Tweety)} \\ \text{bird(Tweety)} \\ \neg\text{fly(Tweety)} \\ \text{fly}(x) \leftarrow \text{bird}(x) \end{array} \right\}$$

$$D_4 = \phi$$

このデフォルト理論 $\mathcal{T}_4 = (W_4, D_4)$ からは¬fly(Tweety) と fly(Tweety) の両方が導かれるため矛盾する. この矛盾が発生した理由は, 直観的には, 通常ルールfly$(x) \leftarrow$ bird(x) は本来デフォルトルールとして表されるべきであるものが通常ルールとして表されているため, Tweety に関して fly(Tweety) が導かれるからであると考えられる. したがってこの通常ルールをデフォルトルール$\frac{\text{bird}(x):\text{M fly}(x)}{\text{fly}(x)}$に変換すべきである. これによって, 定義 2.1より, fly(Tweety) が導かれず¬fly(Tweety) のみが導かれるようになり, 望ましい推論結果が得られる. この変換の操作を知識コンバージョンと呼ぶ.

ここで知識コンバージョンと節 2.3.1 において示した Antoniou の信念翻意の手法を比較する. 知識コンバージョンと Antoniou の信念翻意との最大の相違点は, 知識コンバージョンが例外に着目し, 例外による誤った推論結果を導かれなくすることを目的としているのに対して, Antoniou の信念翻意では新しく加わった論理式 ψ が変換後のデフォルト理論の拡張に含まれるか否かに着目しており, 例外に着目していない点である. このため Antoniou の手法では例外による誤った推論結果が知識ベース変換後にも導かれる場合がある.

例えば節 2.3.1 の $\mathcal{T}_3$の例において, 操作 (I) によって削除される論理式を¬fly(Tweety) とした場合には, 知識ベース変換後の拡張の一つに次のようなものが含まれる.

$$E3(\mathcal{T}_3') = Th(\{\text{penguin(Tweety)}, \text{bird(Tweety)}, \text{fly}(x) \leftarrow \text{bird}(x), p\})$$

この拡張には fly(Tweety) が含まれるため, 例外に着目した場合における望ましい拡張ではない. 一方, $\mathcal{T}_4$の通常ルールfly$(x) \leftarrow$ bird(x) をデフォルトルール$\frac{\text{bird}(x):\text{M fly}(x)}{\text{fly}(x)}$

に変換した後のデフォルト理論を T_4' とすると，T_4' の拡張は以下のようになる．

$$E(T_4') = Th(\{\mathrm{penguin(Tweety)}, \mathrm{bird(Tweety)}, \neg\mathrm{fly(Tweety)}\})$$

この拡張には fly(Tweety) が含まれておらず，例外に着目した場合における望ましい拡張であるといえる．このように非単調論理の知識体系において例外に着目した場合には，知識コンバージョンという新たな知識ベースの変換法による矛盾解消が可能となる．同様に例外に着目することによって，簡潔な知識ベースへの変換が可能となる．次のようなデフォルト理論 $T_5 = (W_5, D_5)$ を想定する．

$$W_5 = \left\{ \begin{array}{l} \mathrm{bird}(x) \leftarrow \mathrm{penguin}(x) \\ \mathrm{bird}(x) \leftarrow \mathrm{swallow}(x) \\ \mathrm{bird}(x) \leftarrow \mathrm{eagle}(x) \\ \mathrm{bird}(x) \leftarrow \mathrm{sparrow}(x) \\ \mathrm{fly}(x) \leftarrow \mathrm{swallow}(x) \\ \mathrm{fly}(x) \leftarrow \mathrm{eagle}(x) \\ \mathrm{fly}(x) \leftarrow \mathrm{sparrow}(x) \end{array} \right\}$$

$$D_5 = \left\{ \frac{\mathrm{bird}(x) : \mathrm{M}\ \neg\mathrm{fly}(x)}{\neg\mathrm{fly}(x)} \right\}$$

このデフォルト理論では bird という一つのクラスの一般的性質がデフォルトルール $\frac{\mathrm{bird}(x):\mathrm{M}\ \neg\mathrm{fly}(x)}{\neg\mathrm{fly}(x)}$ によって表されており，bird は通常 $\neg\mathrm{fly}(x)$ であるが，その例外として $\mathrm{swallow}(x)$，$\mathrm{eagle}(x)$，$\mathrm{sparrow}(x)$ は $\mathrm{fly}(x)$ という性質を持つことを表している．明らかにこの知識ベースはより簡潔に表すことができる．すなわち，デフォルトルールを $\frac{\mathrm{bird}(x):\mathrm{M}\ \mathrm{fly}(x)}{\mathrm{fly}(x)}$ とすれば，swallow，eagle，spallow に関するルールが不要となるため，次のような簡潔なデフォルト理論 $T_5' = (W_5', D_5')$ として表現可能となる．

$$W_5' = \left\{ \begin{array}{l} \mathrm{bird}(x) \leftarrow \mathrm{penguin}(x) \\ \mathrm{bird}(x) \leftarrow \mathrm{swallow}(x) \\ \mathrm{bird}(x) \leftarrow \mathrm{eagle}(x) \\ \mathrm{bird}(x) \leftarrow \mathrm{sparrow}(x) \\ \neg\mathrm{fly}(x) \leftarrow \mathrm{penguin}(x) \end{array} \right\}$$

$$D_5' = \left\{ \frac{\mathrm{bird}(x) : \mathrm{M}\ \mathrm{fly}(x)}{\mathrm{fly}(x)} \right\}$$

こうしたデフォルトルールの変換を，知識の例外に関する観点変更と呼ぶ．観点を変更することによって上の例に挙げたように例外に関するルールを削減できる場合が

あるため，簡潔な知識ベースに変換できる場合がある．これは節 2.3.2 で取り上げた冗長なルールの削除による簡潔な知識ベースへの変換とは根本的に異なり，非単調論理の知識体系において例外に着目したことにより可能となった知識ベースの変換法である．

以上で述べたように，非単調論理の知識体系を用いて，例外に着目した場合には，従来手法にないような知識ベース変換が可能となる．これらの知識ベース変換については次章以降で形式化する．

2.5 結言

本章では非単調論理の代表的な論理体系であるデフォルト論理について概説し，例外との関連について述べた後に，この非単調論理を知識体系とする従来の知識ベース変換法として信念翻意，および冗長なルールの削除方法を紹介した．またこれらの手法が例外に着目していないことを示した上で，例外に着目することによって従来手法にないような知識ベースの変換法が実現可能なことを示唆した．次章以降では，2.4 節で述べた矛盾に基づく知識コンバージョン，および知識の例外に関する観点変更について，それぞれ形式化する．

第 3 章

矛盾に基づく知識コンバージョン

3.1 緒言

本章ではデフォルト論理で表現した知識において，知識の矛盾に基づいてホーン節で表される通常ルールをデフォルトルールに変換する知識コンバージョン (Knowledge Conversion:以下では KC と略す) [馬場口 95, 馬場口 96] を提案する．KC で変換の対象となる通常ルールは例外に関して誤った推論結果を導くようなルールである．しかしながらどの推論結果が誤っているのかを何の情報も無しに同定することは難しい．そこで本章では，矛盾に関与する全ての推論結果を等しく疑わしいと考え，これらを導かなくするという条件を設け，その条件に基づいた KC を定式化する．

本章ではまず定式化の基礎となるデフォルト論理のうち，本章で必要となる部分クラスについて述べた後に，KC に必要とされる事項を定式化し，これを用いて KC の詳細を述べる．さらに関連研究として信念翻意，およびその他の幾つかの知識の変換法を取り上げ，それぞれと本手法を比較する．

3.2 準備

ここでは 2 章で述べたデフォルト論理のうち本章で取り扱う部分クラスを説明する．本章では次の条件を満たすデフォルト理論 $T = (W, D)$ を取り扱う．

1. Wに含まれる論理式がホーン節からなる．ただし W中の単位節は具象化されたもののみを取り扱う．

2. Dに含まれるデフォルトルール $\delta = \frac{p:\mathrm{M}j_1,\ldots,\mathrm{M}j_n}{c}$ が次の条件を満たす．

- $p \neq \phi$，かつ p は原子式の連言．
- c は原子式，$n = 1$，かつ $j_1 = c$.

上の条件のうちデフォルトの根拠と結論が一致するという条件を満たすデフォルトルールを**正規デフォルトルール**という．正規デフォルトルールは，定義 2.1に従うと，$\neg c$ が拡張に含まれる（つまり $\neg c$ が証明できる）場合には適用されない．すなわち結論に関して矛盾が生じる場合には適用されないという性質を持つ．

本章ではデフォルト理論中の Wに含まれる規則節，単位節，目標節をそれぞれ**通常ルール**，**ファクト**，**制約**と呼ぶ．またデフォルトルール$\frac{b:\mathrm{M}h}{h}$ $(b = b_1 \wedge \ldots \wedge b_m)$ を $h \Leftarrow b_1 \wedge \ldots \wedge b_m$と表記する．すなわち本章における式は以下の 4 種類のいずれかとなる．

通常ルール	: $h \leftarrow b_1 \wedge \ldots \wedge b_m$ $(m \geq 1)$
デフォルトルール	: $h \Leftarrow b_1 \wedge \ldots \wedge b_m$ $(m \geq 1)$
ファクト	: l
制約	: $\leftarrow b_1 \wedge \ldots \wedge b_m$ $(m \geq 1)$

本論文では通常ルール $r : h \leftarrow b_1 \wedge \ldots \wedge b_m$の h，および $b_1 \wedge \ldots \wedge b_m$をそれぞれ r の**結論部**，**条件部**と呼び，それぞれ $head(r)$，$body(r)$ と表す．またデフォルトルール δ についても同様とする．なお，本章では一般性を失うことなく，異なるルールに共通の変数が含まれないように変数が改名されているものとする．

ここで用いる知識表現ではルールの条件部に負のリテラルの使用が許されない．そこで原子式 a の否定を表す原子式として，$\overline{a}$のような形式の原子式を導入する．このとき制約 $\leftarrow a \wedge \overline{a}$ が必ずデフォルト理論中に存在するものとする．ただし制約は条件部が成り立つ場合に矛盾することを表す．これは ATMS における NOGOOD 環境 [deKleer 86]，もしくは演繹データベースにおける整合性制約 [小林 90] と同じ働きをするものであり，整合性制約同様，予め知識ベースに与えられるものとする．

例えば前章のデフォルト理論 T_1 を本章における枠組で表した場合，以下のようなデフォルト理論 $T_1' = (W_1', D_1')$ となる．

$$W_1' = \left\{ \begin{array}{l} \mathrm{bird(Tweety)} \\ \mathrm{bird(Tom)} \\ \mathrm{injured(Tweety)} \\ \overline{\mathrm{fly}}(x_1) \leftarrow \mathrm{injured}(x_1) \wedge \mathrm{bird}(x_1) \\ \leftarrow \overline{\mathrm{fly}}(x_2) \wedge \mathrm{fly}(x_2) \end{array} \right\}$$

$$D_1' = \{\mathrm{fly}(x_3) \Leftarrow \mathrm{bird}(x_3)\}$$

3.3 知識の矛盾に基づく知識コンバージョン

本節ではKCを適用した後のデフォルト理論が満たすべき整合状態を定め，その整合状態になるように通常ルールをデフォルトルールに変換する．整合状態とは，矛盾に関係するような推論結果が拡張に含まれないデフォルト理論を指す．本節ではまずKCの概要を述べた後に，整合状態を定義し，アルゴリズムと実行例を示す．

3.3.1 知識コンバージョンの概要

KCを適用する際には，まずデフォルトルールへ変換する通常ルールを同定する必要がある．そのために本章では，通常ルールからデフォルトルールへの変換が，例外によって生じた矛盾を解消する事に着目する．言い換えるならば，矛盾を引き起こすような結論を導く通常ルールこそが，デフォルトルールに変換されるべきものといえる．このような通常ルールは，矛盾を導く過程で必ず用いられる事から，変換対象となる通常ルールの同定はデフォルト理論全体を対象にする必要はなく，矛盾を導くのに必要となる通常ルールとファクトからなる部分的な理論を対象にすればよいことになる．ここでは，そのような理論を**極小矛盾理論**と呼び，同定の第1段階としてデフォルト理論から極小矛盾理論を検出する．

この極小矛盾理論に含まれる通常ルールの中から誤った結論を導くルールを同定するために，矛盾に関連する基礎原子式について考察する．デフォルト理論が矛盾するときには制約の条件部を満たすような基礎原子式の集合が存在するが，そのいずれかが偽である．またそれらの基礎原子式を導く過程で用いられた通常ルールから導かれる他の基礎原子式もやはりいずれかが偽である．しかしながらこれらの基礎原子式のうち，いずれの基礎原子式が偽であるかを判断するのは困難であるため，誤った結論

を導くルールを同定するのは難しい．そこで本研究ではこれらの基礎原子式が等しく疑わしいと考える．これらの基礎原子式を導かないデフォルト理論を整合状態であるといい，デフォルト理論が整合状態になるようにルールを変換する．

このように整合状態に基づいてルールを変換することによって，デフォルト理論の変化に応じたデフォルトルールの修正が可能となる．なぜなら整合状態は極小矛盾理論に基づいており，その極小矛盾理論はデフォルト理論の変化に応じて変わり得るためである．こうしたデフォルト理論の変更を契機としたデフォルトルールの修正を実現するために，ここでは，デフォルトルールを一旦通常ルールに変換した後に，極小矛盾理論の検出およびデフォルトルールへの変換を再実行する．次節以降ではこうしたルールの変換を定式化していく．

3.3.2 知識コンバージョンの定式化

ここでは，通常ルールへの変換，極小矛盾知識，および整合状態について順に詳細を述べる．

デフォルトルールの通常ルール化

まず節 3.3.1の最後に述べた通常ルールへの変換を関数によって次のように定義する．

定義 3.1 (変換関数 Γ) デフォルト理論 $\mathcal{T} = (W, D)$ に対して次のように $\Gamma(\mathcal{T})$ を定義する．

$$\Gamma(\mathcal{T}) = (W \cup W_D, \phi) \quad 但し\ W_D = \{head(\delta) \leftarrow body(\delta) \mid \delta \in D\} \qquad \square$$

$\Gamma(\mathcal{T})$ にはデフォルトルールが存在しないため，$E(\Gamma(\mathcal{T})) = Th(W \cup W_D)$ となる．

極小矛盾理論

デフォルトルールに変換するルールの候補を限定するために，$\Gamma(\mathcal{T})$ から極小矛盾理論を検出する．$\Gamma(\mathcal{T})$ は通常ルールのみからなるため，一般的な導出法を用いて空節を導出する事によって矛盾するルールおよびファクトの集合を検出する事が可能である．そこで制約を目標節とした SLD 反駁を求め，反駁に現れる入力節から極小矛盾理

論を検出する．まず矛盾理論を定義し，その矛盾理論に基づいて極小矛盾理論を定義する．

定義 3.2 (矛盾理論) $\Gamma(\mathcal{T}) = (W \cup W_D, \phi)$ 中の制約 $inc \in W$ を目標節とした *SLD* 反駁 $\langle G, C, \Theta \rangle$ が存在するとする．ただし G，C，Θ はそれぞれ目標節の列 $\langle g_0, \ldots, g_n \rangle$，入力節の列 $\langle c_1, \ldots, c_n \rangle$，単一化代入の列 $\langle \theta_1, \ldots, \theta_n \rangle$ とする．このとき $t = (\{c_1, \ldots, c_n, inc\}, \phi)$ を $\mathcal{T}$ の**矛盾理論**と呼ぶ． □

矛盾理論の定義中の SLD 反駁における代入，および矛盾理論に含まれる通常ルールに関して極小なものを極小矛盾理論という．

定義 3.3 (極小矛盾理論) $\mathcal{T}$ の矛盾理論 $t = (\{c_1, \ldots, c_n, inc\}, \phi)$ に対応する *SLD* 反駁を $\langle G, C, \Theta \rangle$ $(C = \langle c_1, \ldots, c_n \rangle, \Theta = \langle \theta_1, \ldots, \theta_n \rangle)$ とする．このとき次の二つの条件を同時に満たす $\mathcal{T}$ の矛盾理論 $t' = (\{c'_1, \ldots, c'_m, inc\}, \phi)$ $(t' \neq t)$ が存在しないならば，t を極小矛盾理論と呼ぶ．ただし t' に対応する *SLD* 反駁を $\langle G', C', \Theta' \rangle$ $(C' = \langle c'_1, \ldots, c'_m \rangle, \Theta' = \langle \theta'_1, \ldots, \theta'_m \rangle)$ とする．

1. C および C' に含まれる通常ルールの集合をそれぞれ R，R' とするとき $R' \subseteq R$.

2. $\theta_1, \ldots, \theta_n$ および $\theta'_1, \ldots, \theta'_m$ のそれぞれの合成代入を θ，θ' とするとき $\theta' \subseteq \theta$. □

例えば以下のデフォルト理論 $\mathcal{T}_6 = (W_6, D_6)$ を考える．

$$W_6 = \left\{ \begin{array}{l} \leftarrow \overline{\mathrm{fly}}(x_1) \wedge \mathrm{fly}(x_1) \\ \overline{\mathrm{fly}}(x_2) \leftarrow \mathrm{injured}(x_2) \wedge \mathrm{bird}(x_2) \\ \mathrm{fly}(x_3) \leftarrow \mathrm{bird}(x_3) \\ \mathrm{bird}(x_4) \leftarrow \mathrm{swallow}(x_4) \\ \mathrm{bird}(\mathrm{Tweety}) \\ \mathrm{injured}(\mathrm{Tweety}) \end{array} \right\}$$

$$D_6 = \{\phi\}$$

$\mathcal{T}_6$ からは図 3.1のような SLD 反駁が検出される．したがって次のような矛盾理論 $t_6 = (W_{t_6}, D_{t_6})$ が検出される．$\mathcal{T}_6$ からは図 3.1以外の SLD 反駁が検出されないため，t_6 は極小矛盾理論でもある．

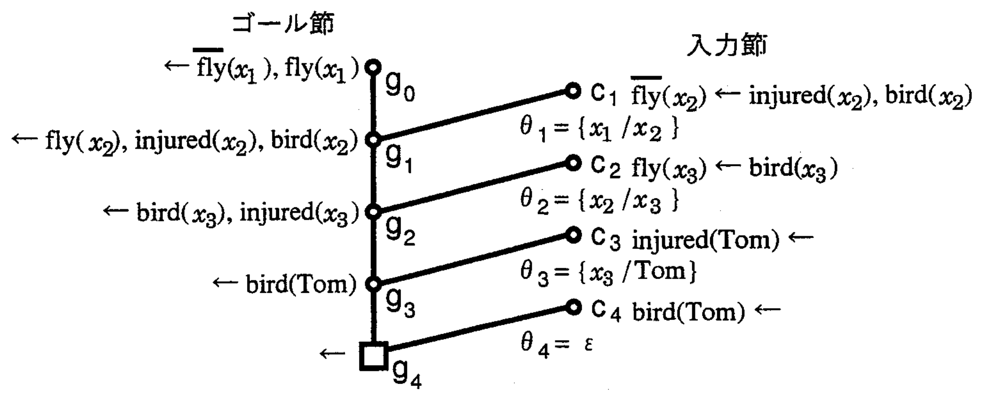


図 3.1 デフォルト理論 T_6から検出される SLD 反駁

$$W_{t_6} = \left\{ \begin{array}{l} \leftarrow \overline{\mathrm{fly}}(x_1) \wedge \mathrm{fly}(x_1) \\ \overline{\mathrm{fly}}(x_2) \leftarrow \mathrm{injured}(x_2) \wedge \mathrm{bird}(x_2) \\ \mathrm{fly}(x_3) \leftarrow \mathrm{bird}(x_3) \\ \mathrm{bird(Tweety)} \\ \mathrm{injured(Tweety)} \end{array} \right\}$$

$D_{t_6} = \{\phi\}$

以降ではデフォルト理論 T の全ての極小矛盾理論 t の集合を $S_t(T)$ と表す.

整合状態

KC を適用した後のデフォルト理論が満たすべき条件として整合状態を定義する. この整合状態を, 極小矛盾理論の通常ルールから導かれる推論結果, すなわち, SLD 反駁において導出に用いられた原子式に基礎代入を施した基礎原子式が, 拡張に含まれるか否かによって定義するために, まずこの基礎原子式について定義する.

定義 3.4 (矛盾基礎原子式集合) 極小矛盾理論 $t = (\{c_1, \ldots, c_n, inc\}, \phi)$ に対応する *SLD* 反駁の単一化代入の列を $\langle \theta_1, \ldots, \theta_n \rangle$ とし, $\theta_1, \ldots, \theta_n$の合成代入を θ とするとき, 次の条件を満たす基礎原子式の集合 L を**矛盾基礎原子式集合**と呼ぶ.

$$L = \{l \mid l = h\theta,\ h \leftarrow b_1 \wedge \ldots \wedge b_m \in \{c_1, \ldots, c_n\}\ (m \geq 0)\} \qquad \square$$

先の例の極小矛盾理論 t_6では，{injured(Tweety), bird(Tweety), fly(Tweety), $\overline{\text{fly}}$(Tweety)} が矛盾基礎原子式集合に含まれる．本章ではこの矛盾基礎原子式集合に含まれる基礎原子式のうち通常ルールから導かれるものは拡張に含まれるべきでないという条件を設ける．

条件 3.1 KCを適用した後のデフォルト理論 $T = (W, D)$ の任意の極小矛盾理論 t に対応する矛盾基礎原子式集合を L とする．また W中のファクトの集合を $F \subseteq W$とする．このとき $l \in L$ のうち

- $l \in F$であるものは全ての拡張に含まれ，
- $l \notin F$であるものは全ての拡張に含まれない． □

例えば t_6では fly(Tweety) および$\overline{\text{fly}}$(Tweety) は KC を適用した後のデフォルト理論の拡張に含まれるべきでないと考える．この条件を満たすようにデフォルト理論を変換したとき，変換後のデフォルト理論は整合状態であるという．

定義 3.5 (整合状態) デフォルト理論 T を T'に変換したとき，T' が次の条件を満たすならば T' は整合状態であるという．

- T' が無矛盾．
- T' が条件 3.1を満たす． □

上の定義 3.5に示されるように，整合状態は現有のデフォルト理論のみに基づいて定義される．したがってデフォルト理論を変化させることによって整合状態も変化する．これはデフォルト理論の変化によって，変換対象となるルールを変えることが可能であることを意味する．一般的に，矛盾を解消するだけであれば，極小矛盾知識に含まれる通常ルールのうち任意のルールをデフォルトルールに変換すればよい．しかしながら現実の世界は動的に変化し得るという点から考えた場合，現実の変化に応じて変換すべきルールも変えるべきである．このように現有のデフォルト理論に基づいた整合状態を定義し，それに基づいて変換対象のルールを決定することは，デフォルトルールの修正という点において有効であると考えられる．

3.3.3 知識コンバージョンアルゴリズム

極小矛盾理論 t は矛盾を導くデフォルト理論であるため，t が制約とファクトのみからなる場合を除いては，t に含まれる通常ルールの中にはファクトによって満たされるルールが少なくとも一つ存在する．このようなファクトによって満たされる全ての r をデフォルトルールに変換することによって t 中のどの通常ルールも条件部が満たされなくなるため，t 中の通常ルールを用いて導かれる基礎原子式がなくなる．したがって条件 3.1が満たされるようになる．さらに r は正規デフォルトルールに変換されるため，節 3.2で述べたように，矛盾を引き起こすような結論は導かなくなり，無矛盾になる．例えば $\mathcal{T}_6$ ではルール $\overline{\mathrm{fly}}(x_2) \leftarrow \mathrm{injured}(x_2), \mathrm{bird}(x_2)$ およびルール $\mathrm{fly}(x_3) \leftarrow \mathrm{bird}(x_3)$ をデフォルトルールに変換する必要がある．

ただしここで述べた通常ルール以外に，例えば矛盾に全く関係ない通常ルールをデフォルトルールに変換しても整合状態に成り得ることに注意されたい．矛盾に関係ないデフォルトルールは，定義 2.1より，通常ルールと同様に結論を導くためである．一般的にデフォルトルールを用いた推論は効率が悪いため，こうしたデフォルトルールは推論効率の悪化を引き起こす．そこで以下では，最小限のデフォルトルールへの変換によってデフォルト理論を整合状態にするアルゴリズムを示す．

アルゴリズム $KC(\mathcal{T}, \mathcal{T}')$
入力：デフォルト理論 $\mathcal{T} = (W, D)$.
出力：デフォルト理論 $\mathcal{T}'$.

1. $\mathcal{T}' := \Gamma(\mathcal{T}) = (W', D')$　（ただし $D' = \phi$）
2. $W' := W' - \{\leftarrow head(r')\sigma \mid \leftarrow head(r')\sigma \in W', r' \in W'$，かつ $body(r')$ が W' 中のファクトの集合に満たされ，その代入が $\sigma\}$;
3. 各 $t = (W_t, \phi) \in S_t(\mathcal{T}')$ について
   (a) $R := \{r \mid r$ は W_t 中の通常ルール, $body(r)$ が W_t 中のファクトの集合 F に満たされる $\}$
   (b) 各 $r \in R$ について
      - $body(r)$ がファクトの集合 F によって満たされるときの基礎代入を θ とする．

- $W' := W' \cup \{\leftarrow head(r)\theta\} - \{r\}$;
- $D' := D' \cup \{head(r) \Leftarrow body(r)\}$;

4. $\mathcal{T}' := (W', D')$ □

上のアルゴリズムのステップ (2) において削除される制約は，このアルゴリズムが以前に実行されたときにステップ (3)-(b) において追加された制約である．以下にアルゴリズム KCの出力 $\mathcal{T}'$ と定義 3.5の関係，およびアルゴリズム KCの出力 $\mathcal{T}'$ の性質を示す．なお，本章の定理の証明および定理に関する補題は節 3.5に示す．

定理 3.1 アルゴリズム KCの Wに含まれるファクトおよび制約からなる集合を W_{FR} とするとき，$W_{FR} \in W$が矛盾しない限り，出力 $\mathcal{T}'$ は整合状態である． □

定理 3.2 アルゴリズム KCでは整合状態にするために必要な極小の通常ルール集合がデフォルトルールに変換される． □

3.3.4 実行例

アルゴリズム KCの実行例として，通常ルールがデフォルトルールに変換される例を示し，さらに一部のデフォルトルールが通常ルールへの変換関数によって通常ルールに戻る例を示す．以下のようなデフォルト理論 $\mathcal{T}_7 = (W_7, D_7)$ を想定する．

$$W_7 = \left\{ \begin{array}{l} \mathrm{pacifist}(x_1) \leftarrow \mathrm{quaker}(x_1) \\ \overline{\mathrm{pacifist}}(x_2) \leftarrow \mathrm{republican}(x_2) \\ \leftarrow \mathrm{pacifist}(x_3) \wedge \overline{\mathrm{pacifist}}(x_3) \end{array} \right\}$$

$$D_7 = \{\phi\}$$

この $\mathcal{T}_7$ に 2 つのファクト quaker(Nixon)，および republican(Nixon) が追加されたとすると，アルゴリズム KCが適用されるが，$\mathcal{T}_7$ にはデフォルトルールが存在しないため，ステップ (1) では何も行われない．またステップ (2) でも何も行われない．続いて次のような極小矛盾理論 $t_7 = (W_{t_7}, D_{t_7})$ が検出される．

$$W_{t_7} = \left\{ \begin{array}{l} \text{pacifist}(x_1) \leftarrow \text{quaker}(x_1) \\ \overline{\text{pacifist}}(x_2) \leftarrow \text{republican}(x_2) \\ \leftarrow \text{pacifist}(x_3) \wedge \overline{\text{pacifist}}(x_3) \\ \text{quaker(Nixon)} \\ \text{republican(Nixon)} \end{array} \right\}$$

$$D_{t_7} = \{\phi\}$$

その後アルゴリズムのステップ (3) においてルールの変換が行われ，以下のデフォルト理論 $T_7' = (W_7', D_7')$ が出力される．

$$W_7' = \left\{ \begin{array}{l} \leftarrow \text{pacifist}(x_3) \wedge \overline{\text{pacifist}}(x_3) \\ \text{quaker(Nixon)} \\ \text{republican(Nixon)} \\ \leftarrow \text{pacifist(Nixon)} \\ \leftarrow \overline{\text{pacifist}}\text{(Nixon)} \end{array} \right\}$$

$$D_7' = \left\{ \begin{array}{l} \text{pacifist}(x_1) \Leftarrow \text{quaker}(x_1) \\ \overline{\text{pacifist}}(x_2) \Leftarrow \text{republican}(x_2) \end{array} \right\}$$

この状態では Nixon が pacifist であるというファクトも，$\overline{\text{pacifist}}$であるというファクトも与えられていないため，$D_7'$のような二つのデフォルトルール（および付随する制約）が生成され，pacifist(Nixon) も$\overline{\text{pacifist}}$(Nixon) も導かれないデフォルト理論となっている．ここで Nixon は pacifist であるというファクトが与えられ，ファクト pacifist(Nixon) がデフォルト理論に加わったとすると，再び KCが適用され最終的に以下のようなデフォルト理論 $T_7'' = (W_7'', D_7'')$ が得られる．

$$W_7'' = \left\{ \begin{array}{l} \text{pacifist}(x_1) \leftarrow \text{quaker}(x_1) \\ \leftarrow \text{pacifist}(x_3) \wedge \overline{\text{pacifist}}(x_3) \\ \text{quaker(Nixon)} \\ \text{republican(Nixon)} \\ \text{pacifist(Nixon)} \\ \leftarrow \overline{\text{pacifist}}\text{(Nixon)} \end{array} \right\}$$

$$D_7'' = \{\overline{\text{pacifist}}(x_2) \Leftarrow \text{republican}(x_2)\}$$

この T_7' から T_7'' への変換ではアルゴリズムのステップ (1) においてデフォルト理論に通常ルールへの変換関数が施される．このため，例えばデフォルトルール $\text{pacifist}(x_1) \Leftarrow$

quaker(x_1) については通常ルールへの変換が施されたと見做すこともできる.

3.4 関連研究との比較

まず本手法と同様に矛盾解消を条件とした諸手法との操作および性質の違いを比較した後に，本手法と類似した操作を行う手法との相違点を述べる.

知識の矛盾解消に関して TMS[Doyle 79] では事実を表す節点（基礎リテラル）に関する信念を変化させる方法が検討されている. TMS では信念の状態として「信じる」「信じない」を表す *In* と *Out* を用い，*NOGOOD* という制約のような節点の追加によって矛盾が生じた場合に各節点の信念の状態を変化させる. TMS ではある節の信念の状態を変化させるときに，その節の根拠となる別の節の信念の状態を連鎖的に変化させていく. 知識表現が異なるため厳密には比較できないが，直観的には，整合状態では矛盾に関連する全ての基礎原子式を疑わしいと見做すのに対して，TMS では矛盾を導く一つの連鎖を選択してその連鎖に現れる節の信念のみを変更するため，TMS ではその選択によって結果が異なる場合がある.

アブダクションの枠組を用いた知識の変換を目的とした研究として，非単調論理の一つである自己認識論理 [Moore 85] を用いて表した知識を対象に，極小の節を削除・追加して矛盾を解消する手法 [Inoue 95] が提案されている. この手法の知識表現をデフォルト論理に変更した場合を考える. ここで削除する節を通常ルール，追加する節をデフォルトルールとしたとき，通常ルールとデフォルトルールの条件部が一致し，かつ通常ルールとデフォルトルールの結論部が一致すれば KC が実現できるように思われるが，実際にはこれは不可能である. なぜならこの枠組では追加・削除する節の数に関して極小性を求めるため，通常ルールを削除するだけで矛盾が解消できるならば，デフォルトルールを追加することができないからである. したがってこの枠組では KC を実現することができない.

また非単調論理を用いた知識においてモデルが存在しない状態を解消する研究として，基礎原子式の追加を用いた手法 [Pereira 91] が検討されている. しかしながらこの手法は，デフォルト論理の枠組で実装した場合，デフォルトルールが既に与えられている前提の下で，そのデフォルトルールを無効にするような基礎リテラルを追加していると見做すことができる. したがってこの手法では KC のような操作は対象とされ

ていないため，デフォルトルールが存在しない場合に発生する矛盾は解消できない．

信念翻意 [Alchourrón 85] および知識ベース更新 [Katsuno 91] の研究分野では，知識の矛盾を契機とした様々な知識変換の前提条件が提案されている．これらの条件のうち，最も代表的なものに，知識変換前の知識のモデルと変換後のモデルの差を極小にするという条件があるが，本手法ではこの条件が成り立たない．なぜなら整合状態では真か偽か疑わしい全ての推論結果を導かないことを条件としているからである．例えば節 3.3.4のデフォルト理論 T_7'のように，本手法では複数の通常ルールが同時にデフォルトルールに変換される場合もある．この場合，いずれか一方のみをデフォルトルールに変換する方がモデルの変化が少なくなる．しかしながら，ルールの選択には何らかの根拠が必要であり，そうした根拠がない場合には疑わしい結論を導かないために T_7'のように双方を変換する方が望ましいと考える．

一方，KC と同様の操作を行う研究として CWS (Closed World Specialization) [Bain 92] および OWS (Open World Specialization)[Inoue 97, 井上 99] が提案されている．これらの手法では学習によって獲得されたルールが他のルールやファクトと矛盾する場合に，獲得されたルールに失敗による否定 (Negation As Failure) の項を加えることによって特殊化する．これは通常ルールからデフォルトルールへの変換と見做すことができる．しかしながらこれらの研究は学習において矛盾しないルールを獲得することが目的であるため，変換の対象となるのは学習で獲得されたルールのみである．一方，本手法では学習によって獲得されたルールを含めた，矛盾する理論全体の中から変換の対象ルールを選択する．この点が本手法とこれらの手法との相違点である．

3.5 定理の証明

定理 3.1の証明に先立ち，アルゴリズム KC適用後の矛盾基礎原子式集合に含まれる基礎原子式に関して以下の補題 3.1を証明する．

補題 3.1 KCを適用した後のデフォルト理論を $T' = (W', D')$ とし，T' の矛盾基礎原子式集合に含まれる任意の基礎原子式を $h\theta$ とする．このとき $\Gamma(T')$ において$\leftarrow h\theta$ を目標節とした任意の SLD 反駁を $\langle G_h, C_h, \Theta_h \rangle$ とするならば，その入力節の列 $C_h = \langle c_{h1}, \ldots, c_{hm} \rangle$ に含まれる通常ルールの集合 $R_h \subseteq \{c_{h1}, \ldots, c_{hm}\}$ のルールは全て矛盾理論に含まれる．また，全ての $\langle G_h, C_h, \Theta_h \rangle$ に対して，次の条件を満たすような$\leftarrow h\theta$

を目標節とする *SLD* 反駁 $\langle G'_h, C'_h, \Theta'_h \rangle$ が存在する．ただし上の$\leftarrow h\theta$ の θ は定義 3.4 の θ とする．

- 入力節の列 C'_hに含まれる通常ルールの集合 R'_hについて $R'_h \subseteq R_h$が成り立つ．
- R'_h中の全てのルールが極小矛盾理論に含まれる． □

ここで補題 3.1を証明する前に証明中で用いる包摂の定義を述べる．

定義 3.6 α，β を二つの節とするとき，ある代入 σ について$\alpha\sigma \subseteq \beta$ がなりたつとき，α は β を代入 σ で包摂するという． □

以下に補題 3.1の証明を示す．

$\langle\langle$ 証明 $\rangle\rangle$ $h\theta$ は矛盾基礎原子式集合に含まれることから，T'には極小矛盾理論が存在する．したがって $\Gamma(T') = (W' \cup W_{D'}, \phi)$ としたとき，入力節に通常ルール $h \leftarrow b_1, \ldots, b_k \in \{W' \cup W_{D'}\}$ を含み，単一化代入の合成代入が θ となるような，W'中の制約を目標節とする SLD 反駁が一つ以上存在する．これらの SLD 反駁のうち，目標節の列が $G' = \langle g'_0, \ldots, g'_i, \ldots, g'_n \rangle$ （$g'_i = \leftarrow h\theta', h\theta'$は $h\theta$をある代入σで包摂$, g'_n = \Box$）となるような SLD 反駁（つまり目標節に h の原子式が現れてもできる限り入力節を $h \leftarrow b_1, \ldots, b_k$とせず，目標節が$\leftarrow h\theta'$ となったときに初めて入力節を $h \leftarrow b_1, \ldots, b_k$ として導出したような SLD 反駁）を $\langle G', C', \Theta' \rangle$ とする．ここで，この SLD 反駁の入力節の列 $C' = \langle c'_1, \ldots, c'_i, c'_{i+1} \ldots, c'_n \rangle$ 中の $c'_{i+1}, \ldots, c'_n$を $c_{h1}, \ldots, c_{hm}$で置き換えた列 $C'' = \langle c'_1, \ldots, c'_i, c_{h1}, \ldots, c_{hm} \rangle$ を考える．このとき $g'_i = \leftarrow h\theta'$で，かつ $h\theta'$ は $h\theta$ を代入 σ で包摂し，さらに $c_{h1}, \ldots, c_{1m}$は$\leftarrow h\theta$ を目標節とした SLD 反駁の入力節の列であるため，C''も W'中の制約を目標節とする SLD 反駁の入力節となる．したがって C''に対応するデフォルト理論 (C'', ϕ) は矛盾理論である．

また$\leftarrow h\theta$ を目標節とした SLD 反駁のうち入力節のルール数に関して極小となる任意の反駁を $\langle G'_h, C'_h, \Theta'_h \rangle$ とし，$C'_h = \langle c_{h1}', \ldots, c_{hl}' \rangle$ とする．このとき C'および C'_hに含まれるルールの極小性よりデフォルト理論 $(\{c'_1, \ldots, c'_i, c_{h1}', \ldots, c_{hl}'\}, \phi)$ は極小矛盾理論である． □

また任意のデフォルト理論 (W, D) に関して，次の定理が成り立つことが知られている．

定理 3.3 (デフォルト論理の矛盾 [**Reiter 80**]) Wが矛盾するとき，またそのときに限り (W, D) は矛盾する唯一の拡張を持つ. □

この定理よりデフォルト理論の無矛盾性は Wのみを対象に無矛盾性を調ればよいことが保証される. 以上で述べた補題および定理を用いて定理 3.1を証明する.

定理 3.1 アルゴリズム KCの Wに含まれるファクトおよび制約からなる集合を W_{FR} とするとき，$W_{FR} \in W$が矛盾しない限り，出力 T' は整合状態である.

⟨⟨ 証明 ⟩⟩ （T' は無矛盾である）

定理 3.3より T' の無矛盾性は W'の無矛盾性のみを調べればよい. 定義 3.2および定義 3.3より，任意の矛盾理論のルール集合 R_1について $R \subseteq R_1$を満たす極小矛盾理論のルール集合が存在する. また制約とファクトのみからなる集合 W_{FR} が無矛盾であるため，極小矛盾理論には少なくとも一つ以上の通常ルールが含まれる. この極小矛盾理論に含まれる通常ルールのうち条件部がファクトの集合に満たされるルールはアルゴリズム KCにおいて必ずデフォルトルールに変換される. したがって矛盾理論に対応する SLD 反駁の入力節に含まれる通常ルールの幾つかは T' ではデフォルトルールになる. このため矛盾理論に対応する SLD 反駁は T' の通常ルールのみでは作ることができない. さらに KCにおいて追加される制約$\leftarrow head(r)\theta$ については，極小矛盾理論のルールの極小性より，単一化可能なファクトが存在しないため，$\leftarrow head(r)\theta$を目標節とした SLD 反駁は存在しない. 以上より T' は無矛盾である.

（T' は条件 **3.1**を満たす）

W'に含まれるファクトの集合を F'とし，T' の任意の拡張を E' とする. このとき T' の矛盾基礎原子式集合 L に含まれる基礎原子式 $h\theta \in L$ が $h\theta \in F'$であるならば $h\theta \in E'$であることは定義 2.1より明らかである. ただし θ は定義 3.4の θ とする.

一方，$h\theta \notin F'$であるならば $h\theta \notin E'$であることを示す. 補題 3.1より，$\Gamma(T')$ において $h\theta$ を導くために用られる$\leftarrow h\theta$ を目標節とした SLD 反駁について，その入力節に含まれるルール集合 R_h中のルールは全て矛盾理論に含まれる. またそのルール集合の部分集合 $R'_h \subseteq R_h$中の全てのルールが極小矛盾理論に含まれるような，R'_hを含む節集合を入力節とし，$\leftarrow h\theta$ を目標節とした SLD 反駁が存在する. $h\theta \notin F'$であるならば，この SLD 反駁の入力節の中には必ず条件部が F'によって満たされる通常ルール

が存在し，そのルールはアルゴリズム KCのステップ (3) においてデフォルトルールに変換されている．そのデフォルトルールは，アルゴリズム KCで追加される制約によって，代入 θ に関する推論結果を導かない．したがって $h\theta \notin E'$である． □

定理 3.2 アルゴリズム KCでは整合状態にするために必要な極小の通常ルール集合がデフォルトルールに変換される．

⟨⟨ 証明 ⟩⟩ アルゴリズム KCにおいて変換の条件を満たす一つの通常ルール rがデフォルトルールに変換されなかったとする．rは条件部がファクトの集合に満たされるため，必ずその結論部を導く．このとき rから導かれた結論は，整合状態の定義に含まれる条件 3.1に反する．したがって KC適用後のデフォルト理論を整合状態にするためには，全ての rをデフォルトルールに変換する必要がある．また定理 3.1 より，KCを適用した後のデフォルト理論は整合状態であることから，KCでは整合状態にするために必要な極小の通常ルール集合がデフォルトルールに変換される． □

3.6 結言

本章では通常ルールをデフォルトルールに変換する KC を提案した．またこの変換の基準として，整合状態という知識の状態を定めた．整合状態は，知識の矛盾に関係する推論結果のうち，その矛盾に関係する通常ルールから導かれる推論結果は誤りである可能性があるとの考えに基づいており，KC を適用した後にはこれらの推論結果のうちファクトに対応する推論結果のみが導かれるようになる．こうしたファクトに基づいた KC を形式化することによって，知識ベースの変化に応じたデフォルトルールの変更が可能となった．

このようにファクトを判断基準としてルールを修正するアプローチの応用として，電子新聞記事の分類システムにおける分類ルールの修正手法を検討している [小山 99, 小山 00]．この手法では分類ルールの例外によって矛盾が発生した場合に分類ルールをデフォルトルールに変換する．この変換によって例外に関する誤った分類のみを適切に排除できるため，ルールの削除などによって矛盾を解消した場合に比べて，分類性能を向上することができる．このように KC は応用面においても有効であると考えられる．

今後の課題としては，通常ルールから導かれる全ての推論結果を疑わしいと見做さなくともよい場合における KC の検討，および本手法の階層デフォルト体系への拡張などが挙げられる．また具体的応用として上述の記事分類システムの構築などが残されている．

第 4 章

知識の例外に関する観点変更

4.1 緒言

本章では非単調論理の知識体系におけるルール数を削減するための知識ベースの変換法を検討する．ここでは知識ベースの変換を，知識の例外に関する観点変更によって実現する [桂田 97a, 桂田 97b]．例外に関する観点とは，あるクラスの性質を表すときに，どのような性質を持つ個体をクラスの一般的な個体とし，どのような性質を持つ個体を例外とするか，といった例外の捉え方をいう．観点を変更することによって例外に関するルールの数が変化するため，ルール数がより少ない観点に変更することによって，ルール数を削減することが可能となる．

以降，知識の例外に関する観点変更の概略を述べた後，観点変更による知識ベースの変換法を定式化し，実験によってその特性を示す．

4.2 知識の例外に関する観点変更

非単調論理の知識体系においてあるクラスに関する代表的な性質をデフォルトを用いて表した場合，何を例外と捉えるか，すなわち**知識の例外に関する観点**によって異なるルールが考えられる．例えば鳥の飛翔性に関する知識の場合，“飛ばない鳥” を例外と捉えた場合には「鳥ならば通常飛ぶ」というデフォルトルールが考えられ，“飛ぶ鳥” を例外と捉えた場合には「鳥ならば通常飛ばない」というデフォルトルールが考えられる．このとき,「鳥ならば通常飛ばない」から「鳥ならば通常飛ぶ」への変更のように，ルールの結論部を変更して例外を変化させることを，**知識の例外に関する観**

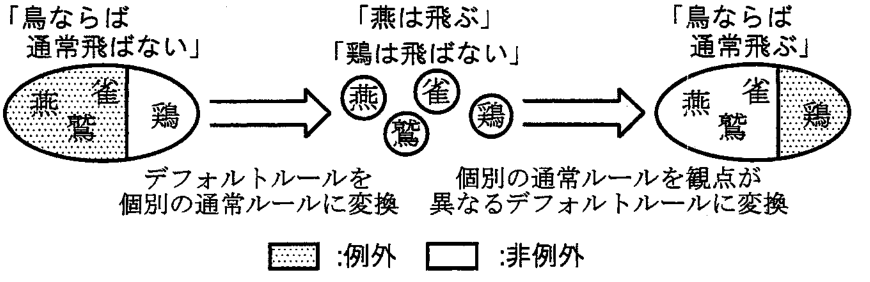


図 4.1 知識の例外に関する観点変更

点変更と呼ぶ.

この観点を変更する際には，観点に応じて保持すべき例外に関するルール（鳥の例では「ニワトリは飛ばない」や「ツバメは飛ぶ」など）も変化するため，知識ベースの構成が変化する．それぞれの知識ベースの中には，ルール数が少ないものもあるため，観点変更によってルール数が削減可能となる．

こうした観点変更を，図 4.1に示されるような方法で実現する．本手法では「鳥ならば通常飛ばない」のようなデフォルトルールについて，その条件部を一つのクラスと考える．このとき，クラスの階層関係を表すルールの集合を参照してデフォルトルールの条件部のクラスを，例えば「ツバメ」や「ニワトリ」などのような部分クラスに細分し，細分化されたそれぞれの部分クラスを条件部とするような，通常ルールに変換する．これによって一旦例外・非例外といった区別を無くし，例外に関する観点を変更して再び条件部のクラスを統合することにより，元のデフォルトルールと異なる「鳥ならば通常飛ぶ」といったルールに変換でき，新たな知識ベースを得ることができる．

4.3 ADAG 表現

観点を変更する際にはルール間の関係を調べる必要がある．したがってルールの依存関係が明確な知識表現が望まれる．そこで，本章では前章で用いたデフォルト表現を，図 4.2のような属性付き有向アサイクリックグラフ（Attributes added Directed Acyclic Graph: **ADAG**）[馬場口 95] というネットワークで表したものを知識表現と

して用いる．ネットワーク表現ではルールの依存関係をノードやリンクの結合関係として捉えることができるため，特定の依存関係を持つルールやファクトの集合の検出が容易になる．

定義 4.1 (ADAG) $\mathcal{G}$ を以下のように定義されたノード，有向リンク，およびファクトから構成されるグラフとする．

- ノード*:* ノード名と引数から構成され，*0* 個以上のファクトを保持する．
- 有向リンク*:* ノードを始点，終点とし，通常もしくはデフォルトの属性を表すラベルを持つ．
- ファクト*:* ノードの引数 x_i と定数 c_i の組の集合 "$x_1 : c_1, \ldots, x_k : c_k$"．

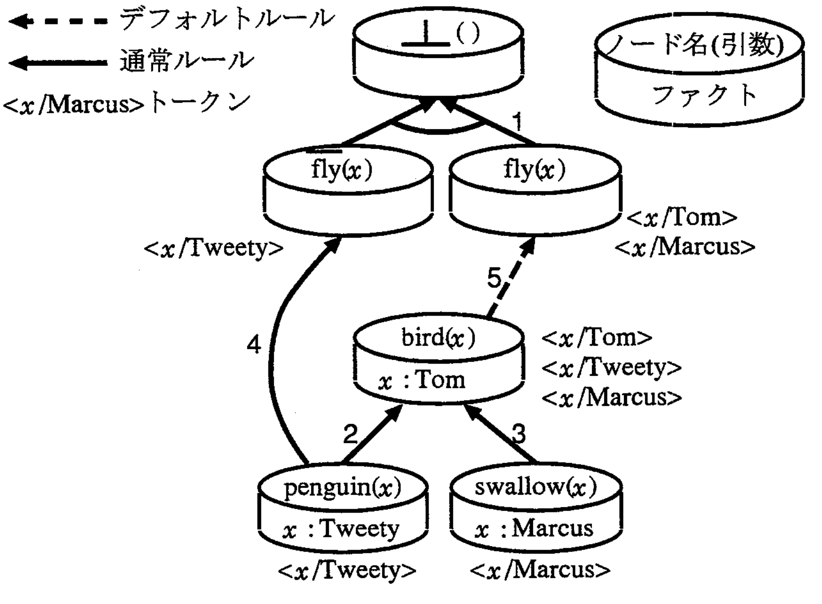
このとき，グラフ $\mathcal{G}$ の全てのノードとリンクが次に示すルールの構成要素であり，かつ $\mathcal{G}$ に再帰がないならば，$\mathcal{G}$ を **ADAG** と呼ぶ．

- ルール*:* 一つの終点ノード n と m 個の始点ノード $N = \{n_1, \ldots, n_m\}$，および始点と終点を繋ぐ m 本の同一属性の有向リンクから構成される．一つのルール中の有向リンクは弧で繋がれ，ノード n の引数は N のいずれかのノードに必ず現れる．

ただしリンクの属性が通常であるルールを通常ルール，デフォルトであるルールをデフォルトルールと呼ぶ． □

ADAG の通常ルール，デフォルトルール，およびファクトはそれぞれ前章で用いたデフォルト論理の通常ルール，デフォルトルール，およびファクトに対応する．R_O を通常ルールの集合，R_Dをデフォルトルールの集合，Zをファクトの集合とするとき，ADAG $\mathcal{G}$ を 3 字組 $\mathcal{G} = \langle R_O, R_D, Z \rangle$ で表す．ただし制約は通常ルールの集合に含まれるものとする．ルール rの条件部の連言は，ADAG においてはノードの集合となる．本章ではこのノードの集合を $body(r)$ として参照する．また結論部は一つのノードであり，本章では $head(r)$ として参照する．図中では，通常，デフォルトルールをそれぞれ実線，破線リンクによって表し，各ルールには固有の識別番号を与える．

ADAG ではトークンをノード上に伝播させることによって推論を行う．ここで，トークンおよびトークンに関連する用語を次のように定義する．



知識

1.「飛ぶと飛ばないは同時に成り立たない」

2.「ペンギンならば鳥である」　3.「ツバメならば鳥である」

4.「ペンギンならば飛ばない」　5.「鳥ならば通常飛ぶ」

「Tweety はペンギンである」　「Marcus はツバメである」

「Tom は鳥である」

図 4.2　ADAG 表現

定義 4.2 (トークン) ノード n の引数 $x_1, \ldots, x_k$に代入される定数が $c_1, \ldots, c_k$であるとき $\langle x_1/c_1, \ldots, x_k/c_k \rangle$ を n のトークンと呼ぶ. □

定義 4.3 (ADAG の矛盾) *ADAG* $\mathcal{G}$ のノード$\perp$にトークンが存在するとき $\mathcal{G}$ は矛盾するという. □

定義 4.4 (トークン集合の共通の基礎代入) トークンの集合 $T = \{t_1, \ldots, t_h\}$ に含まれるトークン t_iの基礎代入 $\{x_1/c_1, \ldots, x_k/c_k\}$ を θ_{t_i}とするとき，次の三つの条件を満たす極小の基礎代入 $\theta = \{x_1/c_1, \ldots, x_m/c_m\}$ をトークン集合 Tの共通の基礎代入と呼び $sub(T)$ と表す.

- 任意の t_iに関して $\theta_{t_i} \subseteq \theta$.
- $s \neq t$ ならば $x_s \neq x_t$ $(1 \leq s, t \leq m)$.
- 全ての x_j/c_j $(1 \leq j \leq m)$ は，一つ以上の t_iに対して $x_j/c_j \in t_i$. □

定義 4.5 (ノード集合を満たすトークン集合) ノード集合 $N = \{n_1, \ldots, n_m\}$ とするとき，ある共通の基礎代入 θ を持つ m 個のトークン集合 $T = \{t_i | 1 \leq i \leq m, t_i$は n_i上のトークン$\}$ をノード集合 Nを満たすトークン集合と呼ぶ. □

推論の際には，まず ADAG 中の全てのファクト $n_i(c_1, \ldots, c_k)$ に対応してノード n_i 上にトークン $\langle x_1/c_1, \ldots, x_k/c_k \rangle$ を設置し，以下のトークン伝播規則に従ってこれらを伝播させる.

定義 4.6 (トークン伝播規則) *ADAG* $\mathcal{G} = \langle R_O, R_D, Z \rangle$ に含まれるルール rの条件部 $body(r)$ を満たすトークン集合 Tが存在し，$sub(T) = \theta$ であるとき，$sub(T \cup \{t\}) = \theta$ となるような $head(r)$ のトークンを t とする．このとき，$r \in R_O$であるならば t を $head(r)$ に伝播する．$r \in R_D$であるならば，$\mathcal{G}$ の矛盾を引き起こさない場合に限って t を $head(r)$ に伝播させる. □

ADAG では 2 章で述べたデフォルト論理における例外と同様に，$r \in R_D$であり，矛盾のために t が伝播できないときに，Tをルール rの例外と呼ぶ．例えば図 4.2では，ノード bird のトークン $\langle x/\text{Tweety} \rangle$ はデフォルトルール 5 の例外である．以上の伝播

規則を繰り返し用いてトークンを伝播させるとき，ノード Nを満たすトークンから順次トークンを生成してノード n に伝播できるならば，Nから n に伝播が可能であるという．ここで，ADAG の推論に関して次の用語を定義する．

定義 4.7 (推論パス) ノード集合 Nを満たすトークン集合が存在すると仮定し，あるルール集合 R に含まれる全てのルールを通常ルールと見なしたときに，R中のルールを用いて Nからノード n に伝播が可能であるならば，ルール集合 Rを Nから n への**推論パス**と呼ぶ．特に全てのルールが通常ルールである推論パスを**通常ルール推論パス**と呼ぶ． □

定義 4.8 (伝播完了状態) いかなるトークンもトークン伝播規則に従って伝播できないような *ADAG* のトークンの状態を**伝播完了状態**と呼ぶ． □

伝播完了状態のトークンは論理的には知識ベース中のルール・ファクト集合の極小モデルを表し，本章ではこれを ADAG の推論結果と考える．例えば，図 4.2のトークンの状態は伝播完了状態である．この図ではノード swallow からノード fly へ伝播可能であり，ルール集合 { 3，5 } はノード swallow からノード fly への推論パスである．このうちルール集合 { 3 } はノード swallow からノード bird への通常ルール推論パスである．

ここで，デフォルトルールが複数ある場合，デフォルト論理における拡張と同様に複数の伝播完了状態が導かれる場合があることに注意されたい．このとき，複数の伝播完了状態の原因となるデフォルトルールをここでは**競合するデフォルトルール**という．

4.4 観点変更のための基本操作

知識の例外に関する観点変更のための基本操作は，次の 3 つのタイプに分類される．

1. **デフォルトルールの特殊化・一般化**：あるデフォルトルールと，その条件部の部分クラスを条件部とするデフォルトルールの集合とを相互変換する．

2. **デフォルトルールの削除・追加**：冗長なデフォルトルールを削除，もしくは追加する．

3. ルールの属性の変更：通常ルールと同じ働きをするデフォルトルールの通常ルール化，もしくは通常ルールのデフォルトルール化を行う．

本手法は新規概念を獲得するような知識獲得手法とは異なるため，観点変更前後においてADAGの伝播完了状態が不変であることを条件とする．また，観点変更前後のADAGは矛盾しないとの前提をおく．このとき，もしノードの追加・削除があれば伝播完了状態のトークンが変化する可能性がある．そこで本章ではリンクの追加，削除，および属性の変更のみを対象とし，ノードの追加・削除は対象外とする．本節ではまずADAGにおいて伝播完了状態が変化しないための条件を示す．続いて，上述の各変換操作を定式化する．

4.4.1 伝播完了状態の保存

上述の各変換操作はノードにおけるトークンの伝播に影響を与えることから，変換操作の前後において伝播完了状態が必ずしも変化しないとは限らない．ここで次の補題の条件が満たされる場合には，変換操作の前後において伝播完了状態が変化しないことが保証される．

補題 4.1 *ADAG* $\mathcal{G} = \langle R_O, R_D, Z\rangle$，$g = \langle R'_O, R'_D, Z'\rangle$ $(R'_O \subseteq R_O, R'_D \subseteq R_D, Z' \subseteq Z)$ とし，$\mathcal{G}$ のうち gの部分をノードの追加，削除を伴わずに $g' = \langle R''_O, R''_D, Z''\rangle$ に変換した *ADAG* を $\mathcal{G}'$ とする．ここで，gに含まれるノードの集合を Nとするとき，$\mathcal{G}$ と $\mathcal{G}'$ において N中の全てのノード上で伝播完了状態のトークンが同じであるならば，$\mathcal{G}$ と $\mathcal{G}'$ の伝播完了状態は一致する． □

この補題 4.1を用いて，次節以降で形式化する各変換操作が伝播完了状態に影響を及ぼさないことを示す．以降，本章の全ての補題および定理の証明は節 4.7に示す．

4.4.2 デフォルトルールの特殊化・一般化

観点変更における主要な操作はデフォルトルールの条件部のクラスの細分化と統合であり，これはルールの特殊化・一般化に相当する．ここで，**特殊化**および**一般化**について，以下のように定義を与える*．

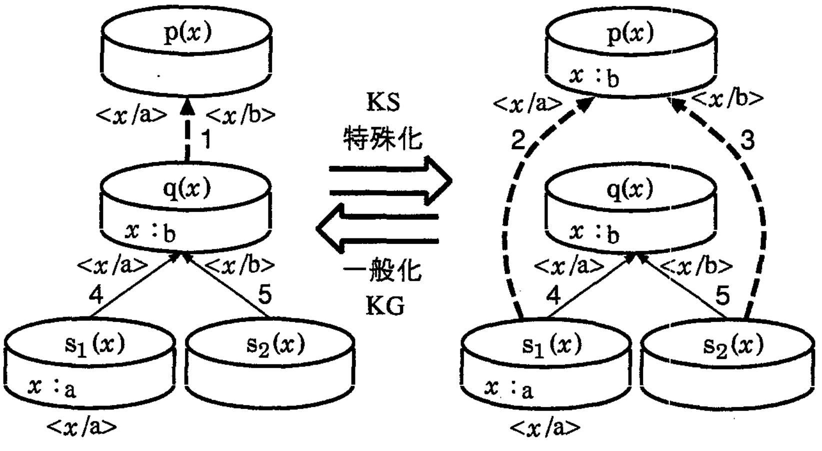


図 4.3 デフォルトルールの特殊化と一般化

定義 4.9 (特殊化) ルールrについて$q \in body(r)$であり，通常ルール$r_O : q \leftarrow s_1, \ldots, s_m$が存在するとき，$r$の条件部の$q$を$s_1, \ldots, s_m$に置き換えることを$r_O$の条件部を用いた$r$の**特殊化**と呼ぶ. □

定義 4.10 (一般化) ルールrについて$\{s_1, \ldots, s_m\} \subseteq body(r)$であり，通常ルール$r_O$: $q \leftarrow s_1, \ldots, s_m$が存在するとき，$r$の条件部の$s_1, \ldots, s_m$を$q$に置き換えることを$r_O$の条件部を用いた$r$の**一般化**と呼ぶ. □

図 4.3のルール 1 からルール 2，3 の二つのルールへの特殊化のように，デフォルトルールr_Dを特殊化することによって，$body(r_D)$の部分クラスを条件部とするルール集合に変換でき，r_Dの「例外」もしくは「非例外」といった区別を一旦なくすことができる．また,「例外」「非例外」といった区別を新たに作ってルール集合のルールを一つのデフォルトルールr_Dに一般化することによって，条件部のクラスを統合することができる．ここで特殊化を行う**特殊化関数** ψ を次のように定義する.

定義 4.11 (特殊化関数) $ADAG\ \mathcal{G} = \langle R_O, R_D, Z \rangle$，$r_D \in R_D$，$q \in body(r_D)$とする．ルール集合 $R = \{r |\ head(r) = q\}$ 中のルールの条件部を用いたr_Dの特殊化で生成さ

*一般的には特殊化，一般化とはそれぞれ条件強化，条件緩和を指すため，条件部のノードの追加や削除なども考えられるが，ここでは取り扱わない.

れたデフォルトルールの集合を R' とするとき，

$$\psi(\mathcal{G}\ , r_D, q) := \begin{cases} R' & \mathit{if}\ \ R \subseteq R_O \\ \phi & \mathit{otherwise} \end{cases} \qquad \square$$

特殊化もしくはその逆の一般化の前後において，$body(r_D)$ のファクト，および r_D の例外と競合について考慮した場合に，伝播完了状態が一致することが次の定理によって証明される．

定理 4.1 *ADAG* $\mathcal{G} = \langle R_O, R_D, Z\rangle$，$\mathcal{G}' = \langle R_O, R_D - \{r_D\} \cup R_\psi, Z \cup Z_1\rangle$ とする．ただし $r_D \in R_D$，$R_\psi = \psi(\mathcal{G}\ , r_D, q)$ $(q \in body(r_D), R_\psi \neq \phi)$ であり，Z_1は次のファクト集合である．

- qのファクトに対応して定められる q上のトークン t'を含む，$body(r_D)$ を満たすトークン集合を Tとするとき，$sub(T) = sub(T \cup \{t\})$ となる $head(r_D)$ のトークン t に対応するファクトの集合のうち $head(r_D)$ に加えて矛盾を引き起こさないもの．

このとき $\mathcal{G}$ と $\mathcal{G}'$ 中の競合するデフォルトルールが同じならば伝播完了状態は等しい． $\square$

定理 4.1の $\mathcal{G}$ から $\mathcal{G}'$ への変更は，r_Dを R_ψへ変更する特殊化である．$body(r_D)$ のファクトに応じた Z_1 の追加，およびデフォルトルールの競合が同じであることによって，伝播完了状態とその数が変化しないことを保証している．この $\mathcal{G}$ から $\mathcal{G}'$ へのデフォルトルールの特殊化を **KS**（Knowledge Specialization），$\mathcal{G}'$ から $\mathcal{G}$ へのデフォルトルールの一般化を **KG**（Knowledge Generalization）と呼び，それぞれ出力 $\mathcal{G}'$ の関数 $F_{KS}(r_D$，q，$\mathcal{G})$，出力 $\mathcal{G}$ の関数 $F_{KG}(R_\psi$，q，$\mathcal{G}')$ として定義する．例えば，図 4.2の ADAG を $\mathcal{G}$ とするとき，r_D:デフォルトルール 5 とその条件部 bird に特殊化関数を用いた場合 $R_\psi = \{\mathrm{fly}(x) \Leftarrow \mathrm{penguin}(x), \mathrm{fly}(x) \Leftarrow \mathrm{swallow}(x)\}$ が求まり，$Z_1 = \{\mathrm{fly(Tom)}\}$ となる．この知識ベースでは r_Dと競合するデフォルトルールが存在しないことから，$F_{KS}(r_D$，bird，$\mathcal{G})$ を用いて r_Dを R_ψに変換し，Z_1を加える操作によって伝播完了状態は変化しない．

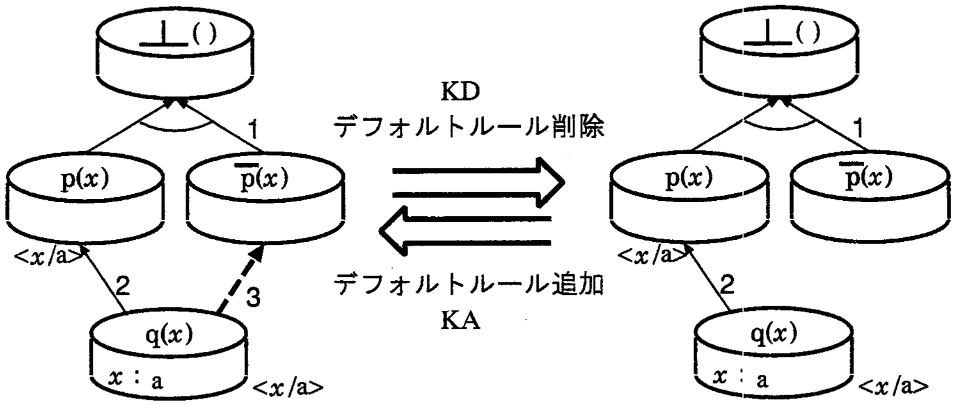


図 4.4 デフォルトルールの削除と追加

4.4.3 デフォルトルールの削除・追加

前節の KS によって生成されたデフォルトルールの中には，条件部を満たす全てのトークン集合 $\{T_1, \ldots, T_m\}$ が例外である図 4.4のデフォルトルール 3 のようなルールが含まれる．この図ではデフォルトルール 3 の結論部および条件部から，制約 1 の条件部の全てのノードへの通常ルール推論パスが存在するため，ルール 3 を用いてトークンを伝播させると制約 1 の条件部が満たされ矛盾が発生する．従っていかなるトークンも伝播させることができない．このようなデフォルトルールの削除，およびその逆の追加により伝播完了状態が変化しないことが次の定理によって保証される．

定理 4.2 *ADAG* $\mathcal{G} = \langle R_O, R_D, Z\rangle$，$r_D \in R_D$，$\perp \leftarrow s_1, \ldots, s_m \in R_O$とする．このとき全ての $s_i (1 \leq i \leq m)$ について $head(r_D) \cup body(r_D)$ の部分集合からの通常ルール推論パス p_iが存在するならば，*ADAG* $\mathcal{G}' = \langle R_O, R_D - \{r_D\}, Z\rangle$ と $\mathcal{G}$ の伝播完了状態は等しい． □

定理 4.2のデフォルトルール削除を **KD**（Knowledge Deletion），逆のデフォルトルール追加を **KA**（Knowledge Addition）と呼び，それぞれ出力 $\mathcal{G}'$ の関数 $F_{KD}(r_D,$ $\mathcal{G})$，出力 $\mathcal{G}$ の関数 $F_{KA}(p,$ $\{q_1, \ldots, q_n\}$，$\mathcal{G}')$ として定義する．ただし KA で新たに生成されるルール r_Dは $head(r_D) = p$，$body(r_D) = \{q_1, \ldots, q_n\}$ を満たすとする．例えば，図 4.2に通常ルール$\overline{\mathrm{fly}}(x) \leftarrow \mathrm{bird}(x)$ を追加した知識ベースを考える．この場合，デフォルトルール 5 の条件部 bird から制約 1 の条件部のノード $\overline{\mathrm{fly}}$への通常ルール推論

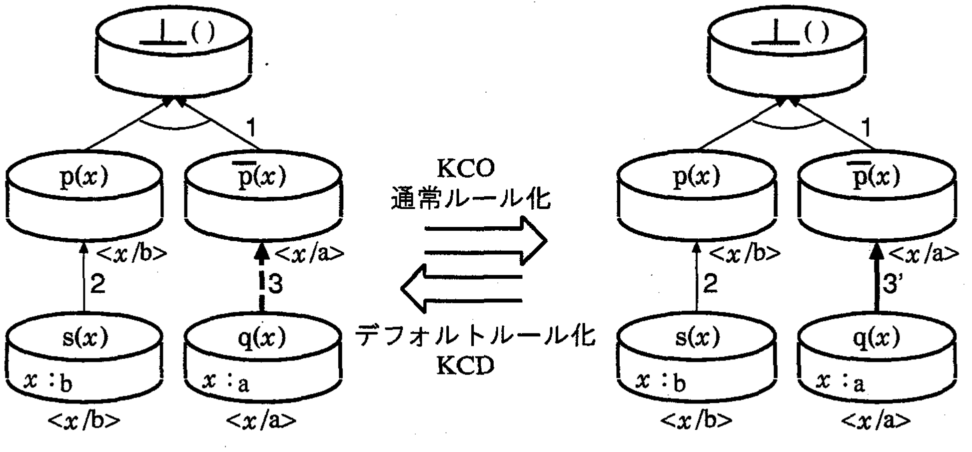


図 4.5 ルールの属性の変更

パスがあり，デフォルトルール 5 の結論部flyから制約 1 の条件部のノードflyへの通常ルール推論パス ϕ（空集合）がある．従って KD，もしくは KA を適用して，伝播完了状態を変化させることなく，デフォルトルール 5 を削除あるいは追加できる．

4.4.4 ルールの属性の変更

前節とは逆に，KS によって生成されたデフォルトルールの中には，条件部を満たす全てのトークン集合 $\{T_1,\ldots,T_m\}$ が非例外になるものも有り得る．このようなデフォルトルールは，図 4.5のルール 3 のように条件部を満たすトークン集合から生成される全てのトークンを伝播でき，いかなるトークンの伝播によっても矛盾を生じさせない．従って通常ルールと同様に働く．

一般的には，図 4.5に示されるような通常ルール化，およびその逆のデフォルトルール化によって伝播完了状態が変化する場合があるが，次の定理に示される条件が満たされる場合には伝播完了状態が変化しない．

定理 4.3 *ADAG* $\mathcal{G} = \langle R_O, R_D, Z\rangle$ とし，$r_D \in R_D$を通常ルール r_O : $head(r_D) \leftarrow body(r_D)$ と置き換えた *ADAG* を $\mathcal{G}'$ とする．このとき，$\mathcal{G}$ 中に r_Dと競合するデフォルトルールが存在せず，$\mathcal{G}'$ が無矛盾であるならば，$\mathcal{G}$ と $\mathcal{G}'$ の伝播完了状態は等しい．
□

定理 4.3のデフォルトルールの通常ルール化を **KCO** (Knowledge Conversion to Ordinary), 逆の通常ルールのデフォルトルール化を **KCD** (Knowledge Conversion to Default) と呼び, それぞれ $\mathcal{G}'$ を出力する関数 $F_{KCO}(r_D, \mathcal{G})$, $\mathcal{G}$ を出力する関数 $F_{KCD}(r_O, \mathcal{G}')$ として定義する. 例えば, 図 4.2からルール 4 が削除された ADAG $\mathcal{G}$ を想定し, 関数 F_{KCO}(ルール 5, $\mathcal{G}$) を用いた場合を考える. このときルール 5 と競合するデフォルトルールが存在せず, ルール 5 を通常ルール化して$\mathrm{fly}(x) \leftarrow \mathrm{bird}(x)$ に変換しても矛盾しないため, ルール 5 をこの通常ルールに変換できる.

4.5 観点変更と知識ベース変換の形式化

4.5.1 観点変更アルゴリズム

観点変更は任意の ADAG に対して適用可能である. しかしながらここでは簡略化のために以下の条件を満たすような, 特に観点変更に関連するルールおよびファクトのみからなる図 4.6のような ADAG を対象として, 節 4.2で述べた観点変更を形式化する. なお, 以下ではクラスとはノードもしくはその連言を指す.

- 図 4.6の r_Dのように, ノード $q_1, \ldots, q_n$の連言からなるクラスがおおよそどの性質を持つかを表すために, $q_1, \ldots, q_n$を条件部, 一つの性質 p_uを結論部とするデフォルトルールが存在する.

- クラスの階層関係, および条件強化によるクラスの定義を表すルールからなる AND/OR 木形式の通常ルールの集合を含み, クラスが持つ二つ以上の性質を条件部に持つような制約 r_Oを含む.

- 図 4.6のノード a, b, d, e の連言およびノード c, d, e の連言のような, $q_1, \ldots, q_n$の部分クラス $N_j (1 \leq j \leq h)$ について, 各 N_jを条件部, 一つの性質を結論部とする性質ルールが存在する. ただし性質ルールのうちルール $p_u \leftarrow N_j$については N_jから p_uへの r_Dを含む推論パスがあるため, 存在しないとする. なお図 4.6ではグラフが複雑にならないよう性質ルールは別記している.

上述の $q_1, \ldots, q_n$の部分クラスとは, r_Dに一回以上 KS を適用して特殊化したときに生成されるデフォルトルールの条件部を指す. このうち, それ以上特殊化できない

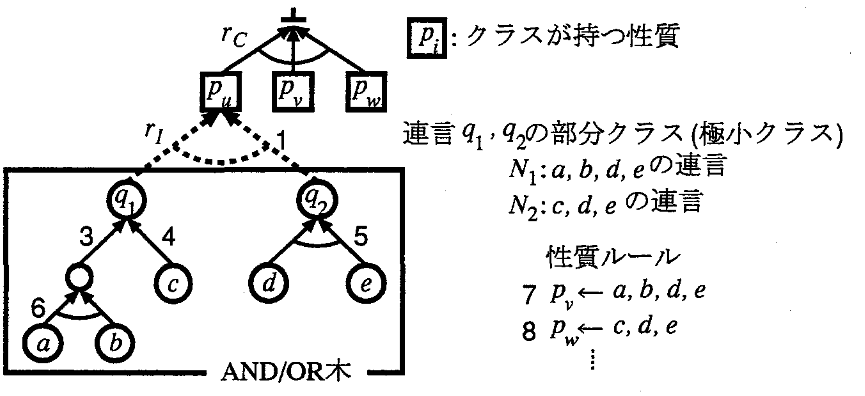


図 4.6 対象とする ADAG

デフォルトルールの条件部を特に極小クラスと呼ぶ．このような ADAG は，例えば様々な化学物質が持つ性質を表す知識ベースのように，例外を含み，分類階層の各クラスが幾つかの性質を持つような知識を表すことができる．

こうした ADAG において $head(r_D)$ を p_uと異なる性質 p_vに変更してデフォルトルール $r'_D : p_v \Leftarrow q_1, \ldots, q_n$にすることが観点変更に相当する．なぜなら r_Dの例外を例外たらしめるトークンは p_uを除く全ての性質に伝播するため，r_Dの例外は r'_Dに関しては非例外になり，r_Dの非例外のうち p_vを除く全ての性質に伝播するものは r'_Dの例外となるからである．以下では，観点変更における r_Dの条件部のクラスの細分化による通常ルール集合の生成，および細分化したクラスの再統合による r'_Dの生成を，前節で定義した 6 つの変換操作で実現する．

まず細分化に関しては，r_Dに KS を用いることにより実現する．このとき生成されるのはデフォルトルールの集合であるため，KCO と KD を用いてルールの削除と通常ルール化を行い，通常ルールのみにする．続いて再統合に関しては KG を用いて実現する．KG を適用するためには，細分化された各クラスを条件部，p_vを結論部とするデフォルトルールが必要であるため，KA および KCD を用いてこれらを生成する．ここで生成されたデフォルトルール集合に KG を適用することにより，p_vを結論部とするデフォルトルールが生成できる．ただしクラスの細分化において KCO も KD も適用できないデフォルトルールが存在する場合がある．このルールに対しては，KS から KG までの変換操作を再帰的に用いて p_vを結論部とするデフォルトルールを生成し，

KA および KCD を用いて生成したデフォルトルールと再統合する.

以上の観点変更をアルゴリズムとして下に示す. アルゴリズムへの入力は, 制約 r_O を R_Oに含む ADAG $\mathcal{G} = \langle R_O, R_D, Z \rangle$ と, $r_D : p_u \Leftarrow q_1, \ldots, q_n \in R_D$ $(p_u \in body(r_O))$, および p_v $(p_v \in body(r_O), p_v \neq p_u)$ であり, 出力は $p_v \Leftarrow q_1, \ldots, q_n$を含む ADAG である.

[観点変更アルゴリズム]

1. これまでに選択していない $q_i \in body(r_D)$ を一つ選択し, $R_3 := \phi$として (2)〜(8) を実行. ただし適用すべき変換操作が適用できない場合, 改めて q_iを選択する. q_iが選択できない場合にはアルゴリズムは何も出力せずに終了.
2. $\mathcal{G} := F_{KS}(r_D, q_i, \mathcal{G})$ とし KS で生成されたデフォルトルール集合を R_ψとする.
3. $r \in R_\psi$のルールのうち $F_{KCO}(r, \mathcal{G})$, もしくは $F_{KD}(r,\ \mathcal{G})$ が適用できるルール集合を R_1, 両方とも適用できないルール集合を R_2とする.
4. $r_1 \in R_1$の各ルールについて $\mathcal{G} := F_{KCO}(r_1, \mathcal{G})$ か $\mathcal{G} := F_{KD}(r_1, \mathcal{G})$ のいずれかを適用.
5. $R_2 = \phi$ならば (7) へ.
6. $r_2 \in R_2$について, $\mathcal{G}$, r_2, p_vを入力として観点変更アルゴリズムを適用. 出力の ADAG からデフォルトルール $r'_2 : p_v \Leftarrow body(r_2)$ を検出し, $R_3 := R_3 \cup \{r'_2\}$, $R_2 := R_2 - \{r_2\}$ として (5) へ. アルゴリズムの出力がなければ, (1) へ.
7. 各 $r_1 \in R_1$について性質ルール $r'_1 := p_v \leftarrow body(r_1) \in R_O$ならば, $\mathcal{G} := F_{KCD}(r'_1,$ $\mathcal{G})$ とする. なければ $\mathcal{G} := F_{KA}(p_v, body(r_1), \mathcal{G})$ とする. KCD, KA によって生成されたデフォルトルール集合を R_4とする.
8. $\mathcal{G} := F_{KG}(R_3 \cup R_4, q_i, \mathcal{G})$ を出力. □

本アルゴリズムは事例中にノイズ, すなわち例外と同じくルールを満たさない事例であるがそれ自体が誤りであるような事例が存在したとしても問題なく動作する. なぜなら本アルゴリズムはクラスを基本単位として実行され, アルゴリズム中で個々の

事例が参照されることはないからである．この意味において，本アルゴリズムは例外とノイズを区別していない．一般的にはクラスの生成などにおいてノイズと例外とを区別することは重要であるが，新しいクラスの生成を伴わない本研究の範囲を超えるものであるので，これについては別の機会に議論したい．

なお，本アルゴリズムではルールの条件部を一つのクラスと見るため，観点変更と関係ない不要属性がルールの条件部に加わったとしても影響なく実行できる．

4.5.2 知識ベース変換アルゴリズムとその実行例

観点変更前後のそれぞれの知識ベースに含まれるルールは異なるため，p_v の選択によってはルール数を削減できる場合がある．一般的にはルール数の削減によって必ずしも良い知識ベースが生成されるとは限らないが，伝播完了状態を変化させず，ファクトがあまり増加しない本手法の場合には，記憶容量の点からルール数が少ない方が望ましい．そこで本手法ではルール数をできるだけ削減するために，ルールが最も減少するような p_v を選択して観点変更アルゴリズムを実行する．

以下に示す知識ベース変換アルゴリズムへの入力は前節の観点変更アルゴリズムへの入力 p_v を制約 $\perp \leftarrow p_1, \ldots, p_m \in R_O$ に変えたものであり，出力は変換後の ADAG である．知識ベース変換アルゴリズム中の Δ_{p_j} は $\mathcal{G}$，r_D，p_j を入力として観点変更アルゴリズムを実行したときのルール数の減少を表す．

[知識ベース変換アルゴリズム]

I. 全ての p_j $(1 \leq j \leq m, j \neq u)$ について Δ_{p_j} を求める．

II. $\Delta_{p_j} > 0$ かつ Δ_{p_j} が最大になる p_j を選択し，$\mathcal{G}$，r_D，p_j を入力として観点変更アルゴリズムを実行．もし p_j が選択できなければ $\mathcal{G}$ を出力して終了． □

この知識ベース変換アルゴリズムは知識ベースをメンテナンスする任意の時点で駆動できる．例えば図 4.7のような ADAG $\mathcal{G}$ を考える．この知識はある業種 ind とそれに属する業者 $corp_i, com_j$ $(1 \leq i \leq 3, 1 \leq j \leq 2)$ が，政党 p_k $(1 \leq k \leq 3)$ を支持するかどうかを表している．$\mathcal{G}$ は，従来 ind に属する業者はおおよそ p_1 を支持していたが，ind に不利な政策をとったために p_1 以外の政党への支持が多くなったような状況を示す．

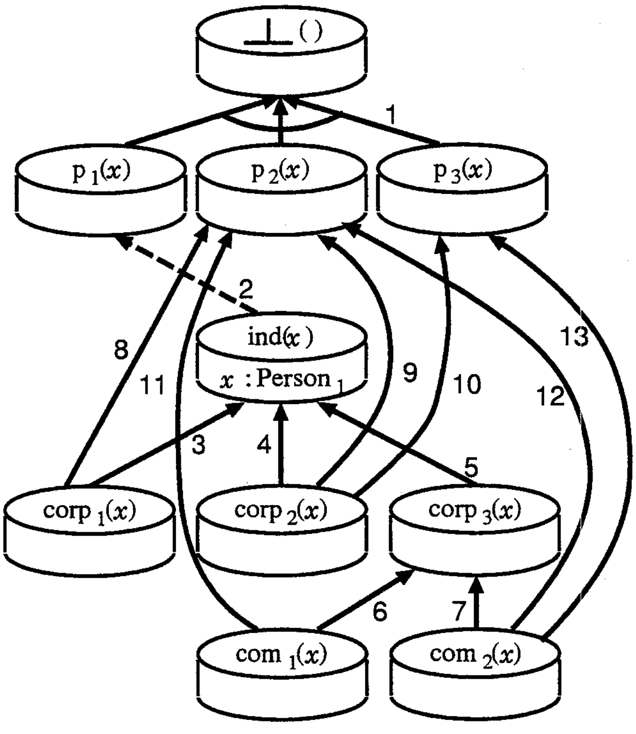


図 4.7 変換前の ADAG

このとき知識ベース変換アルゴリズムへの入力を $\mathcal{G}$，デフォルトルール 2，および制約 1 とすると，I. において $\Delta_{\mathrm{p}_2} = 2, \Delta_{\mathrm{p}_3} = 0$ と求まる．従って II. において $\mathcal{G}$，ルール 2，p_2を入力として観点変更アルゴリズムが実行される．

観点変更アルゴリズムでは，まず (1) において $q_i = ind$ が選択される．次に (2) において，F_{KS}(ルール 2, ind, $\mathcal{G}$) を適用し，ルール 2 を $R_\psi = \{\mathrm{p}_1(x) \Leftarrow \mathrm{corp}_1(x),\ \mathrm{p}_1(x) \Leftarrow \mathrm{corp}_2(x),\ \mathrm{p}_1(x) \Leftarrow \mathrm{corp}_3(x)\}$ に変換し，ファクト $\mathrm{p}_1(\mathrm{Person}_1)$ を知識ベースに追加する．続いて (3) において，$R_1 = \{\mathrm{p}_1(x) \Leftarrow \mathrm{corp}_1(x),\ \mathrm{p}_1(x) \Leftarrow \mathrm{corp}_2(x)\}$，$R_2 = \{\mathrm{p}_1(x) \Leftarrow \mathrm{corp}_3(x)\}$ と設定される．さらに (4) において $\mathrm{p}_1(x) \Leftarrow \mathrm{corp}_1(x)$ に KCO が，$\mathrm{p}_1(x) \Leftarrow \mathrm{corp}_2(x)$ に KD が適用される．ここで $R_2 \neq \phi$であるため，(6) において $\mathcal{G}$，$\mathrm{p}_1(x) \Leftarrow \mathrm{corp}_3(x)$，制約 1 を入力として観点変更アルゴリズムが再帰的に呼ばれる．再帰的に呼ばれた観点変更アルゴリズムにおいて，上と同様にクラスを細分化し，全てのルールを通常ルールで表した状態を図 4.8に示す．

再帰的に呼ばれた観点変更アルゴリズムが終了し，$R_3 = \{\mathrm{p}_2(x) \Leftarrow \mathrm{corp}_3(x)\}$ が求

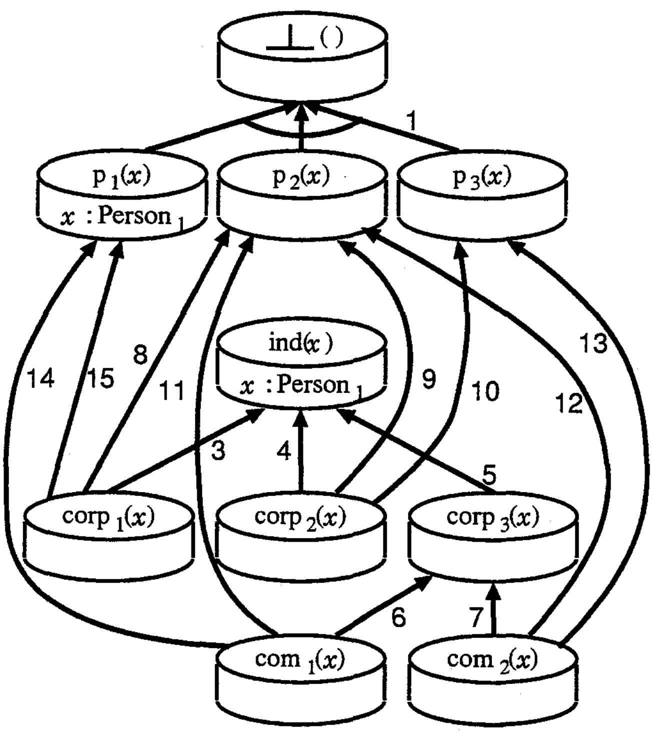


図 4.8 全てのルールが通常ルールで表された ADAG

まったとき，$R_2 = \phi$となるため，(7) を実行する．ここでは，図 4.8の通常ルール 8，9 に KCD を適用してデフォルトルール 8'，9' に変換し，最後に (8) において $F_{KG}(R_3 \cup \{8', 9'\}, ind, \mathcal{G})$ を適用し，知識ベース変換アルゴリズムを終了する．

この一連の観点変更操作によって，p_1を結論部とするデフォルトルールを持つ図 4.7 の知識ベースを，p_2を結論部とするデフォルトルールを持つよりルール数が少ない図 4.9の知識ベースに変換できる．さらに前節までに示した定理より，図 4.7と図 4.9の知識ベースで伝播完了状態が同じであることが保証される．

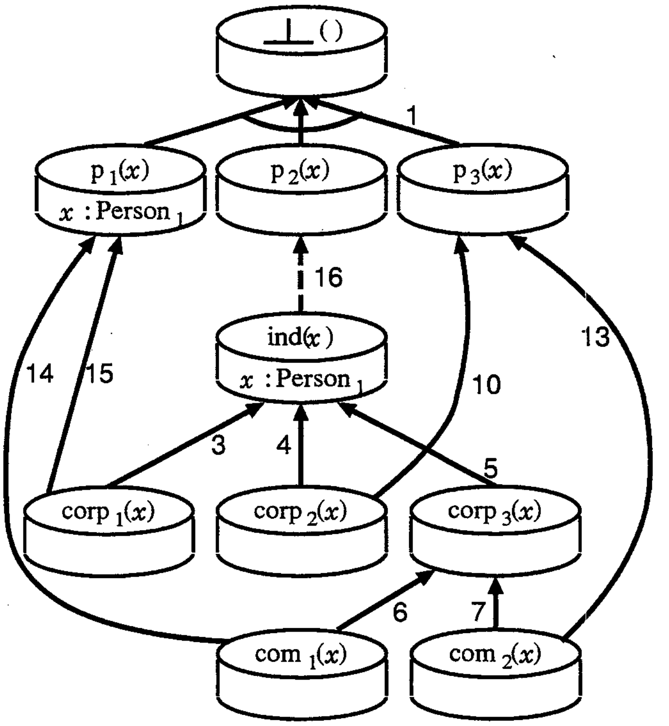


図 4.9 変換後の ADAG

4.6 実験と考察

4.6.1 実験の概要

本手法によるルール数の減少率を求めて，特性を確かめるために実験を行った．ここでは実験の簡単化のために，図 4.6のような ADAG $\mathcal{G}$，デフォルトルール r_D，制約 r_Oに次の制限を加えたものを対象とした．

- r_Dの条件部が一つのノードからなる．
- 部分クラスがある性質 p_iを満たさないことを表すために，性質の否定を表すノード$\overline{p_i}$が存在し，p_iと$\overline{p_i}$を条件部とする制約が存在する．
- 性質ルールは極小クラス N_j $(1 \leq j \leq h)$ を条件部とする．また，全ての性質 p_iについて，各 N_jを条件部とし，p_iもしくは$\overline{p_i}$を結論部とするような性質ルールが

表 4.1 実験対象とした ADAG

ADAG の特徴	**条件**
AND/OR 木の枝の分岐数	最大 4
AND/OR 木の枝が連言になる率	任意
木の深さ	4
性質の数 n	$2 \leq n \leq 5$
例外の率 e	$0 \leq e \leq 1$

存在する．

このような ADAG $\mathcal{G}$ を対象に，$\mathcal{G}$，r_D，r_Oを入力として変換アルゴリズムを適用した．実験では，表 4.1のような条件の ADAG を対象とした．ただし表 4.1の例外の率 e とは各 N_jから $body(r_D)$ に伝播されるトークンが r_Dの例外になる確率，すなわち N_j から p_uを除く $body(r_O)$ の全てのノードにトークンが伝播される確率である．以上の条件を満たす任意の ADAG を生成し，パラメータの変化によるルール数の減少率の変化についてそれぞれ 100 回の試行の平均値を調べた．

4.6.2 実験結果と考察

生成した二つの任意の AND/OR 木 R_1，R_2に関して，R_1について性質の数 $n = 2$, $n = 3$ とした場合と，R_2について $n = 3$ とした場合の実験結果を図 4.10に示す．横軸は e であり，縦軸はルールの全体数に対する知識ベース変換アルゴリズム適用によるルールの減少数の割合 γ を表す．

まず全てのグラフに共通の結果として，e の増加に応じて γ が増加することが分かる．これは e が大きい場合には性質ルールのうち削除できるルール，すなわち p_vを結論部とする性質ルールの割合が増加するためであると考えられる．

次に n を変化させた場合，直観的には n が小さいほど性質ルールのうち削除できる性質ルールの割合が大きくなるため，γ が大きくなると考えられる．しかしながら，実験結果では $0.8 \leq e \leq 1$ のときにはそのような結果が得られているが，$0 \leq e \leq 0.7$ のときには逆の結果が得られていることが分かる．そこで $e = 0.4$ のときの性質ルール

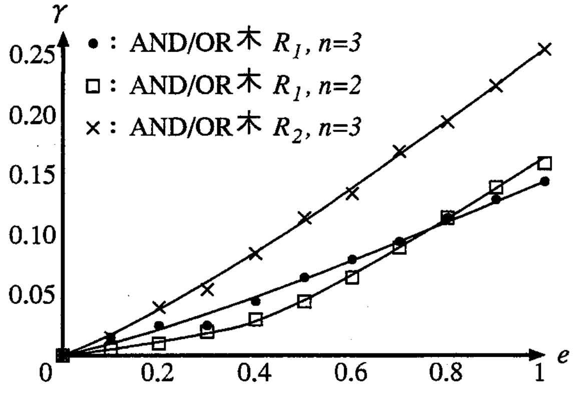


図 4.10 ADAG の変化によるルール数の減少率の変化

を調べたところ，$n=2$ のときに性質ルールが平均 24.5 であるのに対し，$n=3$ のときには平均約 42 であった．単純に n に比例して性質ルールが増加したとすると $n=3$ のときに平均約 37 になるはずであるが，実際にはそれよりも多いことが分かった．これは，e は極小クラス N_j から p_u を除く全ての性質にトークンが伝播される確率であるため，e が低い場合であっても n が大きいと性質ルールが存在する可能性は高くなるためであると考えられる．従って性質ルール全体の増加のために削除される性質ルールが多くなり，このような結果が得られたと思われる．

また，$n=3$ の場合で AND/OR 木を R_1，R_2 とした場合，R_1 を用いた方が R_2 を用いた場合に比べて γ が大きい．この理由を調べるために R_1 と R_2 を比較したところ，R_1 では極小クラスが少ないためにルール数全体に対する性質ルールの割合が少なかった（$e=1$ のとき 44%）のに対して，R_2 ではその逆（$e=1$ のとき 75%）であることが分かった．性質ルールが多いと削除できるルールの数が多くなるため，R_2 を用いた場合に γ が大きくなったと考えられる．

以上より，次のような場合に γ が大きくなることが分かった．

- ルール全体に対する性質ルールの比率が高い場合
  - 極小クラスが多い
  - e が小さいときに n が大きい
- 性質ルールのうち削除できるルールが多い場合

表 4.2 $e=1$ のときのルール数と性質の数 n に応じた減少率 γ

	R_1, $n=3$	R_1, $n=2$	R_2, $n=3$
α	105	88	118
β	46	29	89
γ	0.146	0.165	0.251

・e が大きい

・e が大きいときに n が小さい

特に $e=1$ のときの γ は，ADAG の総ルール数 α に対する性質ルールの数 β の比率 β/αと，性質ルールのうち削除できるルールの比率 $1/n$ との積 $\beta/(n\times\alpha)$ となる．この式に図 4.10の 3 つの ADAG における具体的な値を代入したところ，表 4.2のような結果が得られた．この γ の値はほぼ図 4.10の結果と一致している．

4.7 定理の証明

補題 4.1 *ADAG* $\mathcal{G}=\langle R_O,R_D,Z\rangle$，$g=\langle R'_O,R'_D,Z'\rangle$ $(R'_O\subseteq R_O,R'_D\subseteq R_D,Z'\subseteq Z)$ とし，$\mathcal{G}$ のうち gの部分をノードの追加，削除を伴わずに $g'=\langle R''_O,R''_D,Z''\rangle$ に変換した *ADAG* を $\mathcal{G}'$ とする．ここで，gに含まれるノードの集合を Nとするとき，$\mathcal{G}$ と $\mathcal{G}'$ において N中の全てのノード上で伝播完了状態のトークンが同じであるならば，$\mathcal{G}$ と $\mathcal{G}'$ の伝播完了状態は一致する．

⟨⟨ 証明 ⟩⟩ gを g'に変換した場合に伝播完了状態のトークンが変化する可能性があるノードは，定義 4.6より，Nに含まれるノードおよびこれらのノードにリンクで継るノードのみである．条件より，Nの伝播完了状態におけるトークンは変化しない．また，gおよび g'に含まれないルール・ファクトは変化しないため，Nにリンクで継るノードについても伝播するトークンは変化しない．従って，$\mathcal{G}'$ と $\mathcal{G}$ の伝播完了状態は一致する． □

定理 4.1 *ADAG* $\mathcal{G} = \langle R_O, R_D, Z\rangle$，$\mathcal{G}' = \langle R_O, R_D - \{r_D\} \cup R_\psi, Z \cup Z_1\rangle$ とする．ただし $r_D \in R_D$，$R_\psi = \psi(\mathcal{G}\ , r_D, q)$ $(q \in body(r_D), R_\psi \neq \phi)$ であり，Z_1は次のファクト集合である．

- qのファクトに対応して定められる q上のトークン t'を含む，$body(r_D)$ を満たすトークン集合を Tとするとき，$sub(T) = sub(T \cup \{t\})$ となる $head(r_D)$ のトークン t に対応するファクトの集合のうち $head(r_D)$ に加えて矛盾を引き起こさないもの．

このとき $\mathcal{G}$ と $\mathcal{G}'$ 中の競合するデフォルトルールが同じならば伝播完了状態は等しい．

⟨⟨ 証明 ⟩⟩ 補題 4.1を用いる．$R = \{r_i | head(r_i) = q \quad (1 \leq i \leq m)\}$，$r_D = p \leftarrow q, q_1, \ldots, q_n$とするとき，$g = \langle R, \{r_D\}, \phi\rangle$，$N = \{p, q, q_1, \ldots, q_n, body(r_1), \ldots, body(r_m)\}$ である．Nのノードのうち $q, q_1, \ldots, q_n$および $body(r_i)$ $(1 \leq i \leq m)$ について，これらのノードのファクトは変化せず，これらのノードを結論部とするルールは変化しないため，伝播完了状態におけるトークンは変化しない．また p に伝播するトークンのうち変化する可能性があるものは qのトークンを用いて生成されたトークンであり，これらのトークンは qのファクトおよび，qに伝播するトークンを用いて生成される．このうち qのファクトから生成されるトークンについては，ファクト Z_1の追加により，p においても同一のトークンが生成されるため変化しない．一方，qに伝播するトークンは，各 $body(r_i)$ から伝播するトークンであり，これらは特殊化したルールを用いて q を通らずに直接 p に伝播する．さらにデフォルトルールの競合が変化しないことから，p に伝播するトークンは変化せず，Nの全てのノード上の伝播完了状態におけるトークンは変化しない．従って補題 4.1より $\mathcal{G}$ と $\mathcal{G}'$ の伝播完了状態は等しい． □

定理 4.2 *ADAG* $\mathcal{G} = \langle R_O, R_D, Z\rangle$，$r_D \in R_D$，$\perp \leftarrow s_1, \ldots, s_m \in R_O$とする．このとき全ての s_i $(1 \leq i \leq m)$ について $head(r_D) \cup body(r_D)$ の部分集合からの通常ルール推論パス p_iが存在するならば，*ADAG* $\mathcal{G}' = \langle R_O, R_D - \{r_D\}, Z\rangle$ と $\mathcal{G}$ の伝播完了状態は等しい．

⟨⟨ 証明 ⟩⟩ 補題 4.1を用いる．$P = p_1 \cup \ldots \cup p_m$とするとき，$g = \langle P, \{r_D\}, \phi\rangle$，$N = \{$“$P$に含まれるノード”$\}$ である．まず Pに含まれるノード $head(r_D)$ について，定義 4.6およ

び定義 4.7より，r_Dに従って $head(r_D)$ に伝播するトークンは存在しない．従って r_Dの削除によって $head(r_D)$ 上の伝播完了状態におけるトークンは変化しない．また Pに含まれる $head(r_D)$ を除く各ノードについて，これらのノードのファクトおよびこれらのノードを結論部とするルールは変化しないため，これらのノード上のトークンは変化しない．従って，Nの全てのノード上の伝播完了状態におけるトークンは変化しないため，補題 4.1より $\mathcal{G}$ と $\mathcal{G}'$ の伝播完了状態は等しい． □

定理 4.3 *ADAG* $\mathcal{G} = \langle R_O, R_D, Z\rangle$ とし，$r_D \in R_D$を通常ルール $r_O : head(r_D) \leftarrow body(r_D)$ と置き換えた *ADAG* を $\mathcal{G}'$ とする．このとき，$\mathcal{G}$ 中に r_Dと競合するデフォルトルールが存在せず，$\mathcal{G}'$ が無矛盾であるならば，$\mathcal{G}$ と $\mathcal{G}'$ の伝播完了状態は等しい．

⟨⟨ 証明 ⟩⟩ 補題 4.1を用いる．$g = \langle \phi, \{r_D\}, \phi\rangle$，$N = \{body(r_D) \cup head(r_D)\}$ である．まず $body(r_D)$ の各ノードについて，これらのノードのファクトおよびこれらのノードをを結論部とするルールは変化しないため，$body(r_D)$ 上の伝播完了状態におけるトークンは変化しない．また $head(r_D)$ について，r_Oを用いて伝播可能な全てのトークンを伝播させても無矛盾であり，r_Dと競合するデフォルトルール，すなわち，r_Dを用いたトークン伝播に影響を与えるデフォルトルールが無いことから，定義 4.6より，r_Dを用いてもこれらの全てのトークンを必ず伝播できる．従って，r_Dから r_Oへの変換，またはその逆の変換によって $head(r_D)$ 上の伝播完了状態におけるトークンは変化しない．従って，N の全てのノード上の伝播完了状態におけるトークンは変化しないため，補題 4.1より $\mathcal{G}$ と $\mathcal{G}'$ の伝播完了状態は等しい． □

4.8 結言

本章では，知識の例外に関する観点を変更することによって，非単調論理の枠組において知識ベースを変換する手法を提案した．提案した観点変更は，知識ベース中のルール・ファクト集合の極小モデルを変化させない複数の基本操作を組み合わせることにより実行される．現実世界において人間も例外に関する観点を変更する場合があることから，提案した観点変更は人間が行う知識の変更プロセスの一部を論理的な面から説明し得るものと考えられる．

今後の課題としては，観点変更の効率化，適用に関する契機の検討，観点変更後の例外の割合に関する統計的な評価などが挙げられる．また，基本操作を組み合わせた観点変更以外の知識ベース変換法，および実際の知識ベースシステムへの応用についても検討したい．

第 5 章

結論

現実世界の変化に対応して知識ベースを繰り返し変更したとき，変更されたルールに誤りや無駄があった場合には，知識ベースが矛盾するという問題や，知識ベースが簡潔でなくなるといった問題が発生し得る．こうした問題の解決法として，知識ベースを変換する手法がこれまでに検討されてきた．しかしながらこれらの従来手法では知識体系として主に古典論理を用いているため，現実世界に多く存在する例外の取り扱いには限界がある．例外は上述の問題のうち知識ベースの矛盾の原因ともなるため，矛盾を解消するためにも適切に取り扱う必要がある．そこで非単調論理という，例外を適切に取り扱うことができる論理体系を導入し，非単調論理に基づいた知識ベースの変換法を提案した．これまでにも非単調論理を知識体系とした知識ベースの変換法は幾つか提案されているが，これらの手法は基本的に古典論理における従来の知識ベース変換を拡張したものであり，例外に着目したものではない．本論文では従来手法とは異なり，例外に着目した知識ベースの変換法を提案し，その性質を示した．以下に本論文で得られた諸成果をまとめる．

第 2 章では非単調論理の一つであるデフォルト論理について概説し，デフォルト論理と例外の関係を述べた後に，デフォルト論理を知識体系として用いた信念翻意による矛盾解消法，および非単調論理の一つを知識体系として用いた冗長なルールの削除による簡潔な知識ベースへの変換法についてそれぞれ説明した．これらの従来手法のうち信念翻意の手法に関しては，例外に着目した場合に考えられる望ましい推論結果が，この変換方法では得られない場合があることを示し，例外に着目した知識ベース変換の必要性を示した．また，例外に着目することによって，従来の冗長なルールの削除による簡潔な知識ベースへの変換法と根本的に異なる，観点変更による簡潔な知

識ベースへの変換法が考えられることを示した．以上の従来手法の説明および例外に着目した知識ベース変換の例により，例外に着目した知識ベース変換の有用性に関する知見が得られた．

続いて本論文の第 3 章では，通常ルールをデフォルトルールに変換する知識コンバージョンによって，矛盾と無関係な推論結果を失わずに矛盾を解消する方法を提案した．知識コンバージョンの対象となるルールを検出するために，まず SLD 導出を用いて矛盾を導くルール集合を検出する方法を示し，さらにこのルール集合の中から変換の対象となるルールを検出するために整合状態という知識ベースの状態を定義した．整合状態とは，知識ベースから導かれる推論結果のうち矛盾に関与するものを導かれないようにするという，変換後の知識ベースに関する条件である．本論文では知識ベースを整合状態にするアルゴリズムを示すことによって，知識ベースから真か偽か疑わしい推論結果が導かれないような変換を実現した．このように提案した知識コンバージョンでは，1) 矛盾に無関係な推論結果は失われず，2) 矛盾に関連する推論結果は導かれなくなる．これは正しい推論結果は必ず知識ベースから得られ，誤りの可能性がある推論結果は導かれないという点から，例えば，ユーザへの回答に正確さが求められるような情報提供システムにおける知識修正法として利用できると考えられる．こうしたシステムの一つとして電子新聞記事の分類システムにおける分類ルールの修正手法を検討している．このシステムでは分類ルールの例外によって分類誤りが生じた場合に，人間が分類の修正情報を与え，この情報を用いて分類ルールをデフォルトルールに変換する．この変換によってこの変換によって例外に関する誤った分類のみを適切に排除できるため，ルールの削除などによって矛盾を解消した場合に比べて，分類性能を向上することができる．

最後に本論文の第 4 章では知識の例外に関する観点変更によって簡潔な知識ベースに変換する手法を提案した．観点変更によるルール数削減の実験を行った結果，例外が多いほどルール数が減少することが確かめられ，観点変更は例外が多い場合に有効であるということが実証された．また，観点変更は 6 つの基本的な変換操作によって実現されている．これらの基本的な変換操作は，知識ベースから得られる推論結果を変化させないため，変換操作の適用前に得られていた推論結果は適用後にも失われないという特徴を持つ．この点において，これらの変換操作は知識ベースの変換全般において有用な操作であるといえ，観点変更とは異なるような様々な知識ベース変換法

においても利用できると考えられる．

以上，本論文で論じた知識ベース変換法の特徴を総括的に述べると，以下の 3 点に集約される．

- 非単調論理の知識体系において，例外に着目することによって，知識コンバージョン，観点変更といったこれまでに検討されていない知識ベースの変換が形式化できた．

- 知識コンバージョンにおける整合状態の定義によって，矛盾に無関係な推論結果は失われず，矛盾に関連する推論結果は導かれなくなるような知識ベースの変換を実現した．

- 例外が多い場合には観点変更が有効であることを実証した．また観点変更で用いる 6 つの基本操作は，観点変更に限らず知識ベース変換の一般的操作として利用できることを示唆した．

最後に本研究に関する課題を挙げる．

まず第一に，矛盾に基づく知識コンバージョンにおいて定義した整合状態ではルールから導かれる推論結果のうち，矛盾に関連する全ての推論結果を疑わしいと見做す．しかしながら，こうした整合状態はシステムの利用形態によっては必ずしも望ましいと言えない場合がある．例えばユーザがあらゆる可能性を推論によって検索するようなシステムでは，疑わしい推論結果も導かれたほうが望ましい．したがってシステムの利用形態によっては，疑わしい推論結果を導くような知識ベースの状態を定義し，その状態と整合状態と使い分ける必要がある．こうした，整合状態と異なる状態の定義が課題の一つとして挙げられる．

第二に，本論文では矛盾に基づく知識コンバージョンにおいては正規デフォルト表現を，例外に関する観点変更においては ADAG 表現を知識表現として用いている．このように両手法では異なる知識表現を用いているが，両手法を一つのシステムとして統合するためには知識表現の統一が欠かせない．両手法で用いられている表現のうち，ADAG 表現は 4 章で述べたように知識の依存関係が検出しやすいという利点を持つため，統合の際には ADAG 表現を用いることが望ましい．しかしながら ADAG 表現では再帰的な知識が表現できないという欠点も持っている．したがって ADAG におい

て再帰的な知識を表現できるよう拡張することが一つの課題となる．また両手法で用いられている知識表現ではデフォルトルールを用いた推論におけるルールの優先順序，すなわち，どのデフォルトルールを先に用いて推論結果を導くか，が定められていないため，複数の推論結果が同時に導かれる多重拡張が発生するという問題がある．これを解消するためには優先順序を定めたデフォルト論理である階層デフォルト論理などを導入する必要があり，これが課題の一つである．

第三に，上でも述べたが，例外に関する観点変更において示した6つの基本操作は，知識ベースの変換において広く利用できると考えられる．したがってこの基本操作を組み合わせた，新たな知識ベース変換法を検討することが課題である．

知識ベースシステムが広く一般に用いられる上で，その知識ベースが現実とそぐわなくなった場合に，知識ベースを変換することによって現実との一致を計る方法の検討は急務である．特に現実を表す知識は多分に例外を含むため，その例外に着目した知識ベースの変換法は，今後益々重要性が高まってくるものと考えられる．本研究の成果が多少なりともその先駆的な役割を担えば筆者の最も幸いとするところである．

謝辞

本論文は，大阪大学産業科学研究所 北橋 忠宏 教授の御指導の下に筆者が同大学基礎工学研究科在学中に行なった研究の成果をまとめたものである．

まず最初に本研究を遂行するにあたり，懇切なる御指導，および御鞭撻を戴いた本論文主査の大阪大学産業科学研究所 北橋 忠宏 教授に対し，深甚なる感謝の意を表す．

本論文をまとめるにあたり貴重な御教示を戴いた副査の大阪大学大学院基礎工学研究科 谷口 健一 教授，井上 克郎 教授，東野 輝夫 教授に衷心より御礼申し上げる．

大阪大学大学院基礎工学研究科 今井 正治 教授，柏原 敏伸 教授，菊野 亨 教授，都倉 信樹 教授，故 西川 清史 教授，萩原 兼一 教授，橋本 昭洋 教授，故 藤井 護 教授，藤原 融 教授，宮原 秀夫 教授，村田 正幸 教授，同大学医学部 田村 進一 教授，首藤 勝 教授（現 大阪工業大学情報科学部 教授）には，講義などを通じ様々な御指導と御教示を賜わった．心から感謝する．

本研究の遂行にあたり，叱咤激励と共に熱心なる御助言を戴いた大阪大学産業科学研究所 馬場口 登 助教授には終始御支援を戴いた．ここに深謝申し上げる．

本研究の全過程を通じ，直接の御指導を賜わり，数々の議論と共に，終始有益なるアイデアを戴き，懇切なる御指導を戴いた大阪大学産業科学研究所 大原 剛三 助手に深く御礼申し上げる．

また，筆者の所属する研究室の元スタッフである大阪学院大学 淡 誠一郎 助教授，京都大学 角所 考 助教授，松下電工株式会社 顧 海松 博士，兵庫大学 高野 敦子 講師には研究室において日頃から始終御配慮を賜わった．

筆者の共同研究者である小山 誠 氏にはミーティングなどを通じ，熱心な御討論，本研究を遂行する上での様々なアイデアを戴いた．ここに心から御礼申し上げる．

筆者が現在所属する北橋研究室において，柴田 史久 助手には公私にわたって色々とお世話になった．心より感謝申し上げる．

最後に，日頃から研究などいろいろな面でお世話になった北橋研究室の皆様に心より感謝の意を表する．

参考文献

[赤間 97] 赤間，繁田，宮本: 論理プログラムの等価変換による問題解決の枠組，人工知能学会誌，Vol.12，No.2，pp.266–275 (1997).

[井上 99] 井上，工藤，羽根田: デフォルト規則を含む拡張論理プログラムの学習，人工知能学会誌，Vol.14，No.3，pp.437–445 (1999).

[桂田 95] 桂田，大原，馬場口，北橋: 完全知識の不完全知識への変換，1995 年電子情報通信学会総合大会論文集，pp.245 (1995).

[桂田 96a] 桂田，大原，馬場口，北橋: ルール数の変化を考慮した完全・不完全知識の洗練化，1996 年度人工知能学会全国大会論文集，pp.309–312 (1996).

[桂田 96b] 桂田，大原，馬場口，北橋: 知識コンバージョンと完全・不完全知識の洗練化の統合について，情報処理学会第 53 回全国大会講演論文集 (分冊 2)，pp.229–230 (1996).

[桂田 97a] 桂田，大原，馬場口，北橋: 完全・不完全ルールの使用頻度を考慮した知識洗練化，情報処理学会研究報告 97-AI-107，pp.1–8 (1997).

[桂田 97b] 桂田，大原，馬場口，北橋: 完全/不完全知識の再構成のための知識コンバージョン，人工知能学会研究会資料 SIG-FAI-9702，pp.73–78 (1997).

[桂田 99] 桂田，大原，馬場口，北橋: 不完全知識の例外に関する観点変更による知識ベース再構成，人工知能学会誌，Vol.14，No.3，pp.485–pp.494 (1999).

[桂田 00] 桂田，小山，大原，馬場口，北橋: 知識の矛盾に基づく知識コンバージョン．人工知能学会誌，(投稿中).

[小林 90] 小林 功武: 古典データベースから演繹データベースへ，情報処理学会誌，Vol.31，No.2，pp.189–197 (1990).

[小山 98] 小山，桂田，大原，馬場口，北橋: 完全/不完全ルールの属性変換による矛盾解消のための非対話的手法，1998 年度人工知能学会全国大会論文集，pp.28–31 (1998).

[小山 99] 小山，桂田，大原，馬場口，北橋: Web 上のニュース記事の分類における知識コンバージョンを用いた例外処理，情報処理学会第 59 回全国大会講演論文集 (分冊 2)，pp.81–82 (1999).

[小山 00] 小山，桂田，大原，馬場口，北橋: 文書自動分類における分類誤りを契機とした例外処理とその実験的評価，人工知能学会研究会資料，第 46 回知識ベースシステム研究会 (2000).

[馬場口 95] 馬場口，大原，桂田，北橋: 矛盾を契機とする知識の質的転化，1995 年度人工知能学会全国大会論文集，pp.53–56 (1995).

[馬場口 96] 馬場口 登: 完全／不完全知識を含むデータベースにおける知識獲得，大須賀節雄ほか編，知識科学の展開，オーム社，pp.134-142 (1996).

[Alchourrón 85] Alchourrón, C.E., Gärdenfors, P. and Makinson, D.: On The Logic of Theory Change: Partial Meet Contraction and Revisions Functions, The Journal of Symbolic Logic, Vol.50, No.2, pp.510–530 (1985).

[Antoniou 96] Antoniou, G. and MacNish, C.K.: On the Refinement of Nonmonotonic Knowledge Bases, Proc. PRICAI'96 Workshop on Verification, Validation and Refinement of Knowledge Based Systems, pp.1–6 (1996).

[Antoniou 98] Antoniou, G. and Williams, M.A.: Revising Default Theories, Proc. International Conference on Tools for Artificial Intelligence, pp.423–430 (1998).

[Bain 92] Bain, M. and Muggleton, S.: Non-Monotonic Learning, In S. Muggleton ed., Inductive Logic Programming, Academic Press, pp.145–161 (1992).

[Baral 94] Baral, C.: Rule Based Updates on Simple Knowledge Base, Proc. AAAI-94, pp.136–141 (1994).

[Buchanan 78] Buchanan, B and Feigenbaum, E: DENDRAL and Meta-DENDRAL: Their Applications Dimension, Artificial Intelligence, Vol.11, No.1–2 ,pp.5–24 (1978).

[Dadal 88] Dadal, M,: Investigations Into a Theory of Knowledge Base Revision: Preliminary Report, AAAI-88, pp.475–479 (1988).

[deKleer 86] deKleer, J.: An Assumption-Based Truth Maintenance System, Artificial Intelligence, Vol.28, pp.127–162 (1986).

[del Val 95] delVal, A.: An Analysis of Approximate Knowledge Compilation, Proc. IJCAI'95, pp.830–836 (1995).

[Doyle 79] Doyle, J.: A Truth Maintenance System, Artificial Intelligence, Vol.12, pp.231–272 (1979).

[Gelfond 91] Gelfond, M. and Lifschitz, V.: Classical Negation in Logic Programs and Disjunctive Databases, New Generation Computing, Vol.9, pp.365–385 (1991).

[A.Ginsberg 85] Ginsberg, A., Weiss, S. and Politakis, P.:SEEK2:A Generalized Approach to Automatic Knowledge Base Refinement, Proc. IJCAI'85, pp.367–374 (1985).

[A.Ginsberg 88] Ginsberg, A.: Knowledge-Base Reduction: A New Approach to Checking Knowledge Bases for Inconsistency & Redundancy, AAAI-88, pp.585–589 (1988).

[M.Ginsberg 93] Ginsberg, M.: Essentials of Artificial Intelligence, Morgan Kaufmann (1993).

[Inoue 95] Inoue, K. and Sakama, C.: Abductive Framework for Nonmonotonic Theory Change, Proc. IJCAI'95, pp.204–210 (1995).

[Inoue 97] Inoue, K. and Kudoh, Y.: Learning Extended Logic Programs, Proc. IJCAI'97, pp.176–181 (1997).

[Hammer 93] Hammer, P. L. and Kogan, A.: Optimal Compression of Propositional Horn Knowledge Bases: Complexity and Approximation, Artificial Intelligence, Vol.64, No.1, pp.131–145 (1993).

[Katsuno 91] Katsuno, H. and Mendelzon, A.O.: On the Difference between Updating a Knowledge Base and Revising it, Proc. KR'91, pp.387–394 (1991).

[Katsurada 98] Katsurada, K., Koyama, M., Ohara, K., Babaguchi, N. and Kitahashi, T.: Solving Contradiction in Knowledge-Base without Interaction: Proc. ICSMC-98, pp.1546–1550 (1998).

[Katsurada 00a] Katsurada, K., Ohara, K., Babaguchi, N. and Kitahashi, T.: On Operations for Reconstructing the Complete/Incomplete Knowledge, Proc. The 10th European-Japanese Conference on Information Modelling and Knowledge Bases, (投稿中).

[Katsurada 00b] Katsurada, K., Koyama, M., Ohara, K., Babaguchi, N. and Kitahashi, T.: Converting Ordinary Rules into Default Rules Based on Contradiction of KnowledgeBase, Proc. The 10th European-Japanese Conference on Information Modelling and Knowledge Bases, (投稿中).

[Konolige 88] Konolige, K.: On the Relation between Default and Autoepistemic Logic, Artificial Intelligence, Vol.35, No.3, pp.343–382 (1988).

[McCarthy 80] McCarthy, J.: Circumscription — A Form of Non-monotonic Reasoning, Artificial Intelligence, Vol.13, No.1/2, pp.27–39 (1980).

[Moore 85] Moore, R.C.: Semantical Considerations on Nonmonotonic Logic, Artificial Intelligence, Vol.25, pp.75–94 (1985).

[Pereira 91] Pereira, L.M., Alferes J.J. and Aparício J.N.: Contradiction Removal within Well Founded Semantics, Proc. First Workshop on Logic Programming and Nonmonotonic Reasoning, pp.105–119 (1991).

[Pettorossi 94] Pettorossi, A. and Proietti, M.: Transformation of Logic Programs: Foundations and Techniques, J. Logic Programming, Vol.19/20, pp.261–320 (1994).

[Reiter 80] Reiter, R.: A Logic for Default Reasoning, Artificial Intelligence, Vol.13, pp.81–132 (1980).

[Reiter 78] Reiter, R.:On Closed World Data Bases, In H. Gallaire and J. Minker ed., Logic and Databases, Plenum, pp.55–76 (1978).

[Richards 95] Richards, B. L. and Mooney, R. J.: Automated Refinement of First-Order Horn-Clause Domain Theories, Machine Learning, Vol.19, No.2, pp.95–131 (1995).

[Schmolze 97] Schmolze, J. G. and Snyder, W.: Detecting Redundant Production Rules, AAAI-97, pp.417–423 (1997).

[Selman 91] Selman, B. and Kauts, H.:Knowledge Compilation Using Horn Approximations: Proc. AAAI-91, pp.904–909 (1991).

[Shortliffe 76] Shortliffe, E.H.: Computer Based Medical Consaltations: MYCIN, American Elsenier, (1976).

[Witteveen 95] Witteveen, C and Hoek, W. van der.: Revision by Communication, In V. Marek et al ed., Logic Programming and Nonmonotonic Reasoning, Lecture Notes for Artificial Intelligence, Vol.928, Springer, pp.189–202 (1995).