修 士 論 文

情報源に対する信頼度を考慮した信念の更新

指導教官　東条敏 教授

北陸先端科学技術大学院大学
情報科学研究科情報処理学専攻

鈴木義崇

2001 年 2 月 15 日

要 旨

信念修正は無矛盾性を保存したまま新しい情報を知識ベースに編入するプロセスである. 信念修正の理論を人工知能と哲学的論理学においてもっとも活動的な研究領域の一つにした示唆の源は Alchourron, Gärdenfors, そして Makinson (AGM) による 1985 年のある一枚の論文であるが, その後 AGM の枠組みは様々な形で拡張されていった. その一つである Nebel(1992) は知識ベースにおける異なった文の重要さまたは適切さの間を区別をしようと欲し, そしてこの考えを epistemic relevance と呼ばれる完全擬順序な関係によって形式化した. しかし Cantwell(1988) は Nebel 以外にも様々な優先順位を用いたアプローチがあるにも関わらず, そのような関係がどうやって生じるのかということについての研究がこの研究領域ではほとんどなされていないことを主張した. そして彼は我々は情報源からの情報をその信頼度から評価しているものと見なしたが, しかし彼はどうやって我々は情報源を評価するのかということについて説明を与えていない. そこで本論文においては, 情報源を評価する方法を次のようなものと見なすことによって問題をこの解決した; 情報が知識ベースの更新の結果保持される場合, その情報を与えた情報源の信頼度は上昇し, 他方, 情報が更新の結果捨てられる場合, その情報を与えた情報源の信頼度は下降する. 本論文における主な目的はこのような考えに基づいたシステムを実装することにあるが, しかし信念修正は実装向きというよりもむしろ理論的なものであり, そこで信念修正の理論を調査することによってその実装の方法を提案しなければならない. この問題を解く 2 つの鍵は信念ベースと導出原理である. 信念ベースは信念集合と異なり演繹的に閉じていなくてもかまわない有限な集合であるため, 実装可能となる. 導出原理を使うと, ある論理式の集合が矛盾しているか否かの判定を高速で行うことができる. 問題を解決することによって実装されたシステムは正確にもっとも信頼できる情報源を正しく判断することになり, ある程度の信頼のおけるシステムとなる. そして修正の計算時間の指数的増大を防ぐために制限時間をもうける際に, 時間の短縮を計って正確な思考をあまりに犠牲にしすぎると, 本来信頼すべきでない情報を信頼して失敗をしてしまう点において人間と良く似た働きをなす.

目 次

1 はじめに **1**

2 belief revision の基礎 **3**

2.1 歴史的背景 3

2.2 基本的な概念 5

2.3 AGM の公準について 8

2.4 信念変更の操作の定義 12

3 AGM の問題点と epistemic relevance **15**

3.1 AGM の実装上の困難 15

3.2 epistemic relevance による方法 17

3.3 矛盾の検出法 21

3.4 実装について 23

3.5 実装に基づく実験と問題点 24

4 知識の順序の根拠とモデル化 **26**

4.1 Cantwell の提案 26

4.2 Cantwell の方法 28

4.3 Cantwell の問題点 31

4.4 情報源の信頼度の評価方法 36

4.5 計算時間の問題の解決 37

4.6 モデル化 38

5 実験 **42**

5.1 実験方法 42

5.2 実験結果 44

5.3 考察 48

6 おわりに **52**

第 1章

はじめに

外部入力によって生じる知識ベース内部の矛盾を解消することを目的とした “信念修正 (Belief Revision)” という研究テーマは (Alchourron,Gärdenfors,Makinson 1985) にはじまるものとされる. この論文で提案された方法は以下のようなものである. まず外部入力がこれまでの知識ベースが保有してきた知識に矛盾する場合, 矛盾した知識を導き出すようないくつかの知識を最小限に削除することによって元の知識ベース内の知識の集合の極大部分集合 (maximal subsets) を求める. このような集合は複数存在するのでその中でもっとも重要だとされる部分集合のいくつかを選択関数 (selection function) によって選択し, それらの集合の共通部分を求め, 最後に外部入力を加えて論理的帰結を求めることで更新は完了されるものとする. それらは AGM が定めた revision に関する合理性の公準 (rational postulates) を満たすことが証明された. そして後に, Gärdenfors and Makinson(1988) はそのような公準は epistemic entrenchment と呼ばれる制限つきの線形順序を選択の基準に用いた操作と同値関係にあることを示した.

しかし AGM のモデルでの epistemic entrenchment は制限つきで信念間に順序を与えるのが特徴であり二項間の順序を自由に定めることができなかったので, Nebel は任意の信念間で二項関係が定まる線形順序として epistemic relevance を提案するようになった (Nebel 1992). ところが, これに対して (Cantwell 1998) ではそのような順位はどのように決定されるかという根拠を問い, 情報源の信頼度に応じて順位は決定されるものとした. しかしさらにそのような信頼度がどのように決定されるか, 信頼度はどのように変化するか, と言った議論に対する問いにこの論文では答えていない.

そこで今回は belief revision のモデルに「知識ベースの更新の作業の結果得られた知識

に対して正しい情報を提供した情報源の信頼度は上昇し, 誤った情報を提供した情報源の信頼度は下降する. 」ような仕組みを加えることによって自律的に信頼度を決定し矛盾を解消することができるシステムを実装してみることにする. そして通常この研究テーマにおいては更新を行う操作 (operation) を定義し, そのような操作が操作の合理性を示す公準を満たすかどうかを証明するということを行うが, そのような理論的な操作を実際に実装することは困難を極める. そこで本論文では公準の合理性を問題にしたり, そのような公準を満たす操作を定義したりするということは行わず, むしろいかにして実装上の困難を解消するかと言うことに焦点をおいて議論することにする.

第 2章

belief revisionの基礎

この章では AGM が与えた belief revision の枠組みを導入する. まず belief revision の歴史的背景について述べる. 次に 2.2 節において belief revision の背景となる基本的概念を説明する. そして 2.3 節において AGM による信念変更のための公準について述べる. 最後の 2.4 節においてそのような公準を満たす操作の定義がなされる.

2.1 歴史的背景

belief revision ないし belief change の研究は主に人工知能と哲学的論理学の二つの領域で進められている. まず人工知能の領域においてはデータベースの構築と, 新しい情報が入力された時にデータベースを更新する作業の構築が関心の的となっていた. そこで Doyle(1979) の Truth Maintenance System(TMS) や Fagin, Ullman, and Vardi(1983) の database priorities などが提案されるようになった.

次に哲学的論理学の領域においては, 信念がどのように変更するかという問題は古代以来の哲学的反省の課題の一つとなっていたが, 特に 20 世紀に入って科学理論が発展するメカニズムについて哲学者たちが議論するようになり, その議論の中で信念の変更に関する合理性の基準が確率論的な文脈において提案されるようになった. そして特に 1970 年代後半からの Levi による信念変更のための基礎的な形式的枠組みの提出と Harper によるその発展によってこのような研究は注目されるようになった. そして 1980 年代に入り, belief revision の研究におけるもっとも基本的な枠組みである the AGM model が Alchourron, Gärdenfors, and Makinson(1985) によって提案された. その 3 人の中でも特に中心的な人物だったのは Gärdenfors であり, 彼は初期の頃は反事実的条件文とラムゼイテストに関する説明の語用論, そしてそれに関連した信念の変更と条件文 (if-文) との間の関係について

の研究をしており, その研究の中から belief revision のための認識状態のモデルは産み出された.

この 3 人による研究の成果は彼らが定めた belief revision と belief contraction に関する合理性の基準となる公準 (AGM Postulates) を満たす partial meet revision および partial meet contraction の定義, そして revision に関する postulates と contraction に関する postulates の間での Levi と Harper の Identity を用いた同値関係の指摘を表した 1985 年の論文として発表された. それ以降の belief revision の研究はこの論文を下敷きとして行われるようになった. そしてさらに Gärdenfors and Makinson は 1988 年においてそのような合理的な公準が epistemic entrenchment と呼ばれる制限つきの順序関係と同値関係にあることを示した. この 3 人の研究が高く評価されたのは極端に単純化された表現で多くの結果を示すことができたことにある.

この研究を発端として様々な研究が出現したが, まず Katsuno and Mendelzon(1992) などの AGM のシンタックスの立場とは異なるセマンティクスの立場からの研究, そして Nebel(1991,1992) の AGM と同様のシンタックスの立場からの研究である. セマンティクスの立場において大きな成果を上げた Katsuno and Mendelzon は, 信念の変更に関する理論は AGM の公準のみによって決まるものではないとして, 静的な現実世界に対する正しい認識を得るのにふさわしい belief revision に対し, 動的な世界に対していくつかの可能世界を想定してその変化をおっていく belief update を考案し, そのための公準も設定した. そして AGM のシンタックスの立場を受け継いだのが Nebel であるが, Nebel の AGM との最大の違いは二つある. まず第一点は AGM は epistemic state として論理的帰結において閉じている, つまり論理的閉包となっている belief set を用いたが Nebel は論理的帰結において閉じている必要のない belief base を用いた. 第二点は AGM の epistemic entrenchment のような制限なしで任意の信念間において恣意的に線形順序を定めても良いとする epistemic relevance を採用した. これによって Nebel は base を用いた revision と set を用いた revision との関係を考察し, AGM の公準を満たすかどうかを議論した.

そして Hansson(1998b) によると近年の AGM の枠組みの拡張の研究は 3 つのグループにわかれている. まず Li(1998) のような単一の文ではなく複数の文を入力として認める立場. 次に Olsson(1998) のような AGM とは異なった信念変更の操作を定義する立場. 最後に Cantwell(1988) のような belief state のさらに豊かな表現を追求する立場が存在する. Hansson の指摘以外にも Witteveen(1996) の非単調推論から belief revision を研究する立

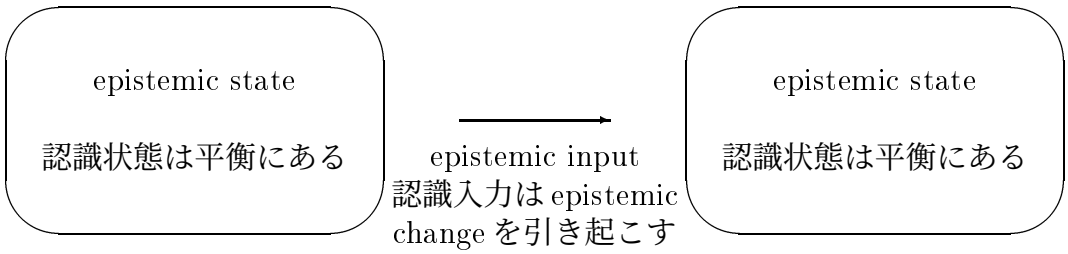


図 2.1: 認識変更の様子

場や, Friedman and Halpern(1997) の知識と信念の論理によって belief revision を定義する立場などが存在している.

2.2 基本的な概念

belief revision は認識状態 (epistemic states) そして認識状態の変更 (changes of epistemic state) についての研究である. そのような研究は一般的には認識論 (epistemic theory) と呼ばれ様々なモデルが研究されてきた. 認識論の主な仕事は知識と信念の変更についての問題を調査するための概念的な装置を提供すること, そして認識の要素 (epistemic elements) と認識のダイナミクスの合理性の基準 (criteria of rationality) の表現を提供することにある. そこでまず認識論の核となる認識の要素を簡単に導入する.

第一の要素はある時点におけるエージェントの知識と信念を表現した認識状態 (epistemic states) または信念の状態 (states of belief) である. これは何かの心理的な内容としてみなされるものではなく, 心理的状態を合理的に理想化したものとして表される. 第二の要素は認識状態に含まれる信念の様々な要素のステータスを描写する認識態度 (epistemic attitudes) である. 第三の要素は認識状態の変化を引き起こす認識入力 (epistemic inputs) である. これらの入力は経験内容としてまたは言語的な情報として与えられるものと考えられる. そして最後に第四の要素は belief revision において中心的な関心事となる認識変更 (epistemic changes) または信念の変更 (changes of beliefs) である. これら四つの要素を基礎として belief revision の基本的な枠組みを構築する.

まず AGM の枠組みにおいてはエージェントの認識状態は文の集合によって表現される. そのような集合は「エージェントがそのモデル化された信念の状態において受理する

(accepts) 文からなっている」ものとして解釈される. エージェントが文を「受理する」という表現はまた「真であると信じる」とか「確かなものと見なす」という表現で呼ばれる. したがって AGM の枠組みにおいては「完全に信じている」ということと「知っている」ということの間に Hintikka のように様相論理を用いて知識と信念を表現するといった belief revision とは異なる領域での知識と信念の研究のような区別をつけない. このような種類の信念の状態のモデルでは集合における文は様相オペレータを必要としないなんらかの対象言語 L から選択されることを前提としている. AGM の枠組みでは L は標準的な文の結合子についての表現を含むということのみを仮定してその細部はどうなっているかということについては議論しない. つまり, 使っている記号自体は命題言語とほとんど変わるところがなくても問題にならない. これらの結合子は次のように記号化される:

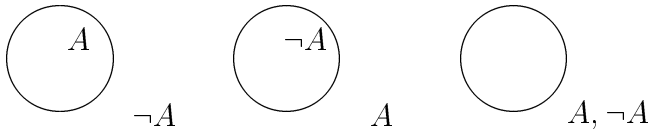
否定:$\neg$
連言:$\wedge$
選言:$\vee$
実質的含意:$\rightarrow$

L の文変数としては $A, B, C, ...$ を使い, 二つの文定数として真 $\top$ と偽 $\bot$ を導入することにする. これらの文は観念的なものとしてのみ使われる. つまり, これらは信念の状態と外部世界の一致については何も含意しておらず, 真理値を表現していない.

ここで認識状態の合理性の基準を導入することにする. というのも, あらゆる文の集合が合理的な信念の状態を表現するために使われるわけではないからである. エージェントによって合理的に保持された文の集合を信念集合 (belief set) という. 合理性の基準となるものは以下の二つである. すなわち (1) 受理された文の集合は無矛盾でなければならない, (2) 受理された文の論理的帰結も受理されなければならない, ということである. このような基準を満たしている場合, 信念の状態は平衡状態 (equilibrium state) にあるという. ここで言語 L は帰結関係 $\vdash$ を持つ論理によって決定されるものとすると, 以上の二つの基準はさらに精密に次のように定義される.

文の集合 K が信念集合 (belief set) であるとは
(i) $\bot$ は K における文の論理的帰結ではない
(ii) $K \vdash B$ ならば $B \in K$ を満たす
場合, そしてその場合に限りいう.

以降, 集合 K のすべての論理的帰結の集合 $\{A : K \vdash A\}$ を $Cn(K)$ で表すことにする.



(i)A is accepted (ii)A is rejected (iii)A is indetermined

図 2.2: 認識態度の 3 種類 (円は信念集合)

したがって, 信念集合は $K = Cn(K)$ が成り立つことがわかる.

次に認識態度について議論する. もし K が信念集合であるならば, あらゆる文 A について A に関して三つの異なった認識態度が表現されうる:

(i) $A \in K$:A は受理される (accepted)

(ii) $\neg A \in K$:A は拒否される (rejected)

(iii) $A \in K$ かつ $\neg A \in K$:A は未決定である (indetermined)

したがって A に関する信念の変更は三つの認識態度のうちの一つから別な一つへの変更として描写される. 全部で 6 つのそのような変更が可能であるが, しかしそれらの変更のうち対称性を持つものはひとまとめにできるので 3 つのタイプにそれらを分けるのが自然である.

最初のタイプは認識態度が「A は未決定である」から「A は受理される」または「A は拒否される」に変更される場合である. この種類の変更のことを拡張 (expansion) と呼ぶことにする. なぜならこのような変更は古い信念を全く削除することなしに新しい信念を belief set に加えるからである. 第二のタイプは「A は受理される」または「A は拒否される」から「A は未決定である」に変更される場合である. この種類の変更のことを縮小 (contraction) と呼ぶことにする. なぜならこのような変更は A(または $\neg A$) における信念を捨てるからである. この二つの変更の場合信念の状態は無矛盾なまま変更するが, 最後のタイプは矛盾を起こす. 最後のタイプは「A は受理される」から「A は拒否される」または「A は拒否される」から「A は受理される」に変更される場合である. この種類の変更を修正 (revision) と呼ぶことにする. なぜなら以前はその否定が受理されていたような

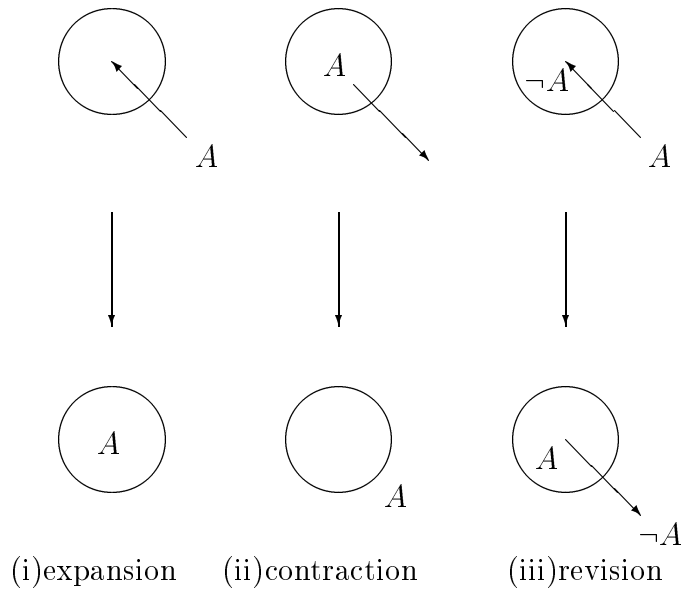


図 2.3: 認識変更の 3 種類

信念を受理する, つまり以前は受理されていた信念が拒否されるからである.

このような認識変更を起こす外力となるものは認識入力である. 認識入力には二つの種類が考えられる. 一つ目はそれまで受理されていなかった文 A (または $\neg A$) が新しい証言の結果として, または仮説として「加えられる (added)」場合, そして二つ目はそれまで受理されていた文 A(または $\neg A$) が矛盾した証言が与えられた結果として, または以前受理されていた文に矛盾する文を受け入れる用意として「捨てる (give up)」場合である.

2.3 AGM の公準について

前節において 3 種類の認識変更を導入した. しかし認識変更の在り方はどのようなものであっても良いというものではなく, 合理的な変更の基準を要求する. そこで, AGM はその変更の在り方を公準の形で示した. 以下においてその公準を導入する. まず revision($\dot{+}$ で表す) の公準は以下の通りになる.

($K\dot{+}1$) 任意の文 ϕ および信念集合 K について, $K\dot{+}\phi$ は信念集合である.
($K\dot{+}2$) $\phi \in K\dot{+}\phi$.

$(K\dot{+}3)$ $K\dot{+}\phi \subseteq K + \phi$.
$(K\dot{+}4)$ もし $\neg\phi \notin K$ ならば, $K + \phi \subseteq K\dot{+}\phi$.
$(K\dot{+}5)$ $K\dot{+}\phi = K_\perp$ であるのは $\vdash \neg\phi$ の場合に, そしてその場合に限り成り立つ.
$(K\dot{+}6)$ もし $\vdash \phi \leftrightarrow \psi$ ならば, $K\dot{+}\phi = K\dot{+}\psi$.
$(K\dot{+}7)$ $K\dot{+}\phi \wedge \psi \subseteq (K\dot{+}\phi) + \psi$.
$(K\dot{+}8)$ もし $\neg\psi \notin K\dot{+}\phi$ ならば, $(K\dot{+}\phi) + \psi \subseteq K\dot{+}\phi \wedge \psi$.

以上の公準において $+$ で示されているものは expansion の操作である. 順を追って公準を説明していく. まず $(K\dot{+}1)$ は入力と belief set がどんなものであれ変更の結果は平衡状態に保たれなければならないという要請である. $(K\dot{+}2)$ は入力が信念の状態の変更後に受理されることを保証する. $(K\dot{+}3)$ と $(K\dot{+}4)$ は $\neg\phi \notin K$ の場合, revision の操作が expansion と等しくなることを保証する. ただし, $(K\dot{+}3)$ において $\neg\phi \notin K$ の前提がないのは, $\neg\phi \in K$ の場合 $K\dot{+}\phi = K_\perp$ になり, 一般性を保つからである. したがって expansion は revision の特殊な場合であることがわかる. $(K\dot{+}5)$ は $\neg\phi$ が論理的な必然性を持たない限り, 変更の結果は無矛盾になることを保証する. $(K\dot{+}6)$ は同値となる文が入力の場合は revision の結果は同じとするものであり, revision の結果は入力された文の形式よりもむしろ内容に依存していることを示している. 残り二つの公準は複合された belief revision に関する公準である. つまり $(K\dot{+}7)$ と $(K\dot{+}8)$ はもし $K\dot{+}\phi$ が K の revision であり, $K\dot{+}\phi$ がさらに文 ψ によって revision または expansion などの変更をされるならば, そのような変更は可能な限り $K\dot{+}\phi$ の expansion であるべきであるという要請である. ただし, $(K\dot{+}7)$ において $\neg\phi \notin K$ の前提がないのは, $\neg\phi \in K$ の場合 $(K\dot{+}\phi) + \psi = K_\perp$ になり, 一般性を保つからである.

次に contraction($\dot{-}$ で表す) の公準は以下の通りになる.

$(K\dot{-}1)$ 任意の文 ϕ および信念集合 K について, $K\dot{-}\phi$ は信念集合である.
$(K\dot{-}2)$ $K\dot{-}\phi \subseteq K$.
$(K\dot{-}3)$ もし $\phi \notin K$ ならば, $K\dot{-}\phi = K$.
$(K\dot{-}4)$ もし $\vdash \phi$ でないならば, $\phi \notin K\dot{-}\phi$.
$(K\dot{-}5)$ もし $\phi \in K$ ならば, $K \subseteq (K\dot{-}\phi) + \phi$.
$(K\dot{-}6)$ もし $\vdash \phi \leftrightarrow \psi$ ならば, $K\dot{-}\phi = K\dot{-}\psi$.
$(K\dot{-}7)$ $K\dot{-}\phi \cap K\dot{-}\psi \subseteq K\dot{-}\phi \wedge \psi$.
$(K\dot{-}8)$ もし $\phi \notin K\dot{-}\phi \wedge \psi$ ならば, $K\dot{-}\phi \wedge \psi \subseteq K\dot{-}\psi$.

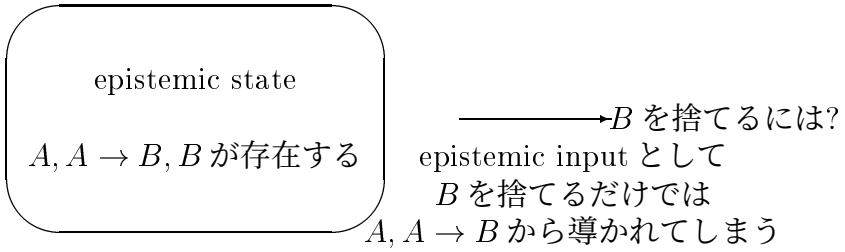


図 2.4: 認識変更の様子

順を追って公準を説明していく. まず $(K\dot{-}1)$ は入力と belief set がどんなものであれ変更の結果は平衡状態に保たれなければならないという要請である. $(K\dot{-}2)$ は contraction はいくつかの文を捨てることによって成り立つ操作なので, $K\dot{-}\phi$ には新しい信念は生じてはならないとする要請である. $(K\dot{-}3)$ は $\phi \notin K$ のときには経済的にみて K から何も捨てられるべきではないという要求である. $(K\dot{-}4)$ は ϕ が論理的に恒真でない限り取り除かれた文は $K\dot{-}\phi$ において保持される信念の論理的帰結とはならないということを示している. $(K\dot{-}5)$ は最初に contract した信念と同じ文に関して expand した後で K におけるすべての信念が回復されることを示している. $(K\dot{-}6)$ は $(K\dot{+}6)$ のアナロジーである. そして $(K\dot{-}7)$ と $(K\dot{-}8)$ も $(K\dot{+}7)$ と $(K\dot{+}8)$ と同じ動機に基づいた公準である.

expansion にはこのような公準をおかないことにする. なぜなら expansion の操作は論理学と集合論の概念で単純に定義することができるからである.

(Def +) $K + \phi = \{\psi : K \cup \{\phi\} \vdash \psi\}$

ところが revision と contraction はある文を捨てる場合にそれに付随してどういったほかの文を捨てるのかということに関していくつかの候補が生じてしまうために論理的な根拠だけでどの文を削除するかということが決定できない. つまり revision と contraction は expansion ほど単純な操作ではない. そのことを以下で例を用いて説明する.

図 2.4 のように $A,\ A \to B,\ B$ が受理されている認識状態を考える. この場合 contraction によってただ単に B を捨てるか, revision によって受理されている文に矛盾する $\neg B$ が入力されたという理由により B を捨てなければならないことになったとする. そうすると B のみを捨てればそれで良いという問題ではなくなってしまう. なぜなら $A,\ A \to B$ から B

が導かれてしまうので, $A,\ A \to B$ のうちどちらか一方, または両方を捨てなければならない. しかし両方を捨てることは知識は貴重なものなのでできるだけ多く持っていた方がいいという経済性の原理に反するのでどちらか一方を捨てた方が良いということになる. このように contraction と revision には論理学と集合論の概念で単純に定義することができない経済性の原理が働いているために公準を必要としたのである.

ここで上記のような公準を満たす操作が定義できるかどうかを考えてみる必要がある. その前に Levi と Harper の Identity を導入して revision と contraction の公準の間の関係を示す.

まず Levi Identity は revision を contraciton と expansion の合成としてみる考えである. さらに精密には revision $K\dot{+}\phi$ を構築するためにまず K に $\neg\phi$ に関する contraction を用いて, それから $K\dot{-}\neg\phi$ に ϕ に関する expansion を用いる方法である. 形式的に Levi Identity は以下のように定義できる.

(Def $\dot{+}$) $\quad K\dot{+}\phi = (K\dot{-}\neg\phi) + \phi$

この定義が適切なものであることは次のことによってわかる.

定理 もし contraction $\dot{-}$ が $(K\dot{-}1)$ から $(K\dot{-}4)$ と $(K\dot{-}6)$ を満足するならば, (Def $\dot{+}$) から得られる revision $\dot{+}$ は $(K\dot{+}1) - (K\dot{+}6)$ を満足する. さらに $(K\dot{-}7)$ を満足するならば, $(K\dot{+}7)$ が (Def $\dot{+}$) から得られる revision によって満足される. さらに $(K\dot{-}8)$ を満足するならば, $(K\dot{+}8)$ が (Def $\dot{+}$) から得られる revision によって満足される.

$(K\dot{-}5)$ が使われないことに注意.

次に Harper Identity は revision によって contraction を定義する考えである. この考えは精密には ψ が contraction $K\dot{-}\phi$ において受理されるのは ψ が K と $K\dot{+}\neg\phi$ において受理される場合であり, そしてその場合に限りとするものである. 形式的に Harper Identity は次のように定義される.

(Def $\dot{-}$) $\quad K\dot{-}\phi = K \cap K\dot{+}\neg\phi$

この定義が適切なものであることは次のことによってわかる.

定理 もし revision $\dot{+}$ が $(K\dot{+}1)$ から $(K\dot{+}6)$ を満足するならば, (Def $\dot{-}$) から得られる contraction $\dot{-}$ は $(K\dot{-}1) - (K\dot{-}6)$ を満足する. さらに $(K\dot{+}7)$ を満足するならば, $(K\dot{-}7)$

が (Def $\dot{-}$) から得られる contracsion によって満足される. さらに $(K\dot{+}8)$ を満足するならば, $(K\dot{-}8)$ が (Def $\dot{-}$) から得られる contracsion によって満足される.

このようにして revision または contraction の操作を定義する場合, どちらか一方の操作を定義して公準を満たすかどうかの合理性を判定し, Levi または Harper の Identity を用いることで他方を定義すればよい. 以下の議論では contraction の定義について議論し, revision は Levi Identity を用いて得るものとする.

2.4 信念変更の操作の定義

この節では belief sets の contraction の操作についての明確なモデルを導入する. これに Levi Identity を用いれば revision の操作も同様にモデル化できる.

contraction $K\dot{-}\phi$ の定義をする際に AGM が基本的な理念としているのは先ほども述べた「情報の経済性」である. つまり情報というものはできる限り多ければ多いほどよいというもので, contraction のような信念を捨てる操作においてはできるだけ多くの知識が残るような操作を採用したい. そこで次の概念が有効になる. ある belief set K' が「ϕ の含意に失敗する K の極大部分集合」であるのは次のような場合, そしてその場合に限る.

(i) $K' \subseteq K$,

(ii) $\phi \notin Cn(K')$, そして

(iii) $K' \subset K'' \subseteq K$ となるような任意の K'' について, $\phi \in Cn(K'')$.

最後の箇条はもし K' が K からの文で expand された場合, それは ϕ を含意することを意味している. ϕ の含意に失敗する K の極大部分集合すべてを要素とする集合を $K\perp\phi$ で示す. この $K\perp\phi$ を用いて contraction を構築する問題を解決するいくつかの方法を考えることができる. まず最初に $K\dot{-}\phi$ を $K\perp\phi$ における極大部分集合のうちの一つとする方法である. 選択関数 γ を集合 A のうちの要素の一つを取り上げる関数 $\gamma(A)$ として定義すると $K\dot{-}\phi$ を以下のように定義できる.

(Maxichoice) $\vdash \phi$ でない場合 $K\dot{-}\phi = \gamma(K\perp\phi)$, それ以外の場合 $K\dot{-}\phi = K$.

このような contraction を Maxichoice contraction と呼ぶ.

Maxichoice contraction は AGM の公準のうち $(K\dot{-}1) - (K\dot{-}6)$ を満たすが, しかしこの contraction の操作はあまりにも「大きすぎる」. というのも, Levi Identity を用いて Maxichoice contraction から導いた revision $\dot{+}$ はあらゆる文 ψ について $\neg\phi \in K$ のとき $K\dot{+}\phi$ は $\psi \in K\dot{+}\phi$ または $\neg\psi \in K\dot{+}\phi$ になるという意味において極大になってしまう.

二つ目の方法は $K\dot{-}\phi$ を $K\perp\phi$ における極大部分集合すべての共通部分とする方法である. この方法によって $K\dot{-}\phi$ を以下のように定義できる.

(Meet) $K\perp\phi$ が空でない場合 $K\dot{-}\phi = \cap(K\perp\phi)$, それ以外の場合 $K\dot{-}\phi = K$.

このような contraction を full meet contraction と呼ぶ.

full meet contraction は AGM の公準のうち $(K\dot{-}1) - (K\dot{-}6)$ を満たすが, しかしこの contraction の操作はあまりにも「小さすぎる」. というのも, Levi Identity をもちいて full meet contraction から導いた revision $\dot{+}$ は $\neg\phi \in K$ のとき $K\dot{+}\phi = Cn(\{\phi\})$ になってしまい, ϕ の論理的帰結のみを結果として保持することになってしまう.

三つ目の方法は $K\dot{-}\phi$ の定義に $K\perp\phi$ における極大部分集合のうちのいくつかを選択して使う方法である. 選択関数 γ を集合 A の空でない部分集合を取り上げる関数 $\gamma(A)$ として定義すると $K\dot{-}\phi$ を以下のように定義できる.

(Partial Meet) $K\perp\phi$ が空でない場合 $K\dot{-}\phi = \cap\gamma(K\perp\phi)$, それ以外の場合 $K\dot{-}\phi = K$.

このような contraction を partial meet contraction と呼ぶ. partial meet contraction は以下のような定理を満たす.

定理 任意の belief set K について, $\dot{-}$ が partial meet contraction であるのは, $\dot{-}$ が公準 $(K\dot{-}1) - (K\dot{-}6)$ を満たす場合であり, そしてその場合に限る.

よって Levi Identity を用いることによって partial meet revision を以下のように定義することができる.

partial meet revision

$$K\dot{-}\phi = \cap\gamma(K\perp\neg\phi) + \phi$$

ここで選択関数に制約を加えることによって, どんな要素を選択する関数にするべきかを考慮する. $K\perp\phi$ の「もっとも良い」要素を γ が選択するという考えは $K\perp\phi$ における極大部分集合に良さを決定する順序が存在すると仮定することによってさらに精密にすることができる. $K\perp\phi$ 上のすべての要素の上に推移的で反射的な順序関係 $\leq$ が存在すると仮定する. $K\perp\phi$ が空でないとき, この関係を使って順序の上で最大となる要素を選択する選択関数を定義できる.

(Def γ) $\gamma(K\perp\phi) = \{K' \in K\perp\phi : \text{任意の } K'' \in K\perp\phi \text{ について } K'' \leq K'\}$.

(Def γ) によって与えられる選択関数 γ を使用した $\leq$ から定義される contraction を transitively relational partial meet contraction と呼ぶ. この方法によって定義した contraction に関して次の定理が成り立つ.

定理 任意の belief set K について,$\dot{-}$ が transitively relational partial meet contraction であるのは, $\dot{-}$ が公準 $(K\dot{-}1) - (K\dot{-}8)$ を満たす場合であり, そしてその場合に限る.

このようにしてすべての公準を満たす contraction は定義可能であることが判明した. なお, この公準は Katsuno and Mendelzon(1991) で意味論的な操作における公準に置き換えられたり, Hansson(1992) で recovery という通称で知られる公準 $(K\dot{-}6)$ が直観に合わないことを指摘されたり, そしてそれをふまえて Ferme(1998) で recovery を満足しない場合の contraction の定義の研究が行われたりしながら, 様々な問題点の指摘や改良が行われている.

第 3 章

AGMの問題点とepistemic relevance

この章では AGM の枠組みがそのままでは実装には使えないことを指摘し, 代わりに Nebel の epistemic relevance を用いた revision を導入する. しかしこの章の終りで, やはり Nebel の方法にも実装には使えない部分が存在することを指摘する. まず 3.1 節において AGM の実装上において困難となる問題点を述べる. 次に 3.2 節において Nebel が導入した epistemic relevance について述べる. 3.3 節において矛盾を検出するために導出原理を用いることを説明し, それによって, 3.4 節で epistemic relevance をもちいた revision の一応の実装としてのシステムは完成するが, 3.5 節における実験において以前として不備が残ることを検証する.

3.1 AGMの実装上の困難

AGM のした仕事の大きな成果はわずかな道具立てのみを用いて信念変更が満たすべき公準を組み立て, それを満たすような操作を定義したこと, そして Levi と Harper の Identity を用いて revision と contraction の公準の間の関係を調査したことにある. しかし, このような理論的な研究がそのままデータベースの更新という人工知能において重要視されている研究に結び付くかというと, そうではない. なぜならこのような理論は実装に当たっては困難な部分を伴うからである. 以下において, その困難を指摘していく.

まず問題になってくるのは epistemic state の平衡状態についての考え方である. 先述の通り AGM の枠組みにおける epistemic state の平衡状態とは (i) 受理された文の集合は無矛盾でなければならない, (ii) 受理された文の論理的帰結も受理されなければならない, ということである. ここで (i) は特に議論の余地はないが, 問題になるのは (ii) の方である. というのも, 論理的帰結を導く操作というものは原理的には可算無限個の文を産み出すもの

であり, 有限個の文しか実際の実装では扱えないからである. したがって最初の困難は以下の通りになる.

問題点 1　AGM の考えた平衡状態によって定義された belief set は論理的帰結の操作を伴うことによって可算無限個の要素を持つことになるが, 実装において可算無限の要素を扱うことは不可能である. しかも, 人間の認知活動においてもすべての論理的帰結を認識することはあり得ないので, AGM の枠組みにおいて現れる epistemic state とは一体何を扱ったものなのかわからなくなってしまう.

次に問題となる事柄は入力文 ϕ の否定の含意に失敗する belief set K の極大部分集合の求め方の問題である. 入力文の否定を演繹しないような K の極大部分集合を求めるために必要な $(K\perp\neg\phi)$ の要素を求める操作には論理的帰結の操作が不可欠になるのだが, これを実際に Gentzen の自然演繹を使って実装するとなると入力文の否定が演繹されるのが確認されるまでにどれだけのステップ数を要求することになるのだろうか. $(K\perp\neg\phi)$ を求める方法にこのような困難が生じることは partial meet revision の実行が不可能になることを意味する. したがって, ここでの問題は次のようになる.

問題点 2　部分集合が入力文の否定を演繹するかしないかをチェックするために論理的帰結の操作を実装しなければならないが, 実際に実装したとしてチェックにどのくらいのステップ数を必要とするのか.

そして K の部分集合を求めていく際に極大部分集合を最終的に見つけたいのだから, その方法として要素数の多い K の部分集合から順にチェックしていって, 入力文の否定を演繹しない部分集合のうち要素数最大のものを発見するまで操作を続けていくという方法がもっとも自然である. しかしこの方法では最悪の場合計算ステップ数が K の要素のべき乗回になってしまい, 要素数が増えれば指数的に計算時間は増大してしまう. したがって問題は以下のようになる.

問題点 3　AGM の「情報の数は多ければ多いほどよい」という経済的な原則はかえって計算時間の増大という別な経済的な原則を破壊してしまう. なぜなら, 入力文 ϕ の否定の含意に失敗する belief set$K$ の極大部分集合を求めるための計算時間は K 内の要素が増えるほど指数的に増大してしまうからである.

最後に transitively relational partial meet contraction の定義において, 極大部分集合間に順序をつけるという考え方が問題である. 極大部分集合は $(K\perp\neg\phi)$ を求める操作が完了

してはじめて生成されるものであるが, この時一体どういう基準で $(K\perp\neg\phi)$ 内の要素間で順序を決定するのかが不明である. したがって次のような問題が生じる.

問題点 4　あらかじめ極大部分集合間に順序を定めておくことはできない. したがって transitively relational partial meet contraction とは異なる仕組みで選択関数 γ の仕組みを考える必要がある.

なお Doyle(1991) においては AGM の枠組みにおいて以上の４つの問題点のうち問題点 1,3 を議論しているが Doyle においては論理的帰結の操作に問題があることを指摘しながらも結局操作 Cn を用いており, しかも問題点 4 を解消することを考えていない.

3.2　epistemic relevance による方法

以上のような問題点を解消するために, ここで Nebel が提案した epistemic relevance を導入する. これは知識ベース内の信念間に順序をつけることによって信念変更後に知識ベースに保持しておくべき知識を選択しようというものである.

まず Nebel が用いる形式的な枠組みについて説明する. 言語に関しては AGM と同じで通常の論理結合子を持った言語 L を想定する. $\top$,$\perp$ の導入も同じように行う. 論理的帰結を導く操作 Cn も同様に定義する. ただし, epistemic state を表す知識ベースに関しては論理的に閉じていることを必ずしも要求しない. このような文の恣意的な集合のことを信念ベース (belief base) と呼ぶことにする. これまで扱ってきた belief sets を $K, L, M, ...$ で表すことにし, belief base は $A, B, C, ...$ で表すことにする. belief base は論理的帰結の操作によって閉じている必要がないので有限個の要素で表現が可能である. したがって問題点 1 の解決が可能となる.

次に異なった文の間に優先順位をつける方法について議論する. このような順位は変更することができる知識と保護されるべき知識の間に区別をつけるのに適している. これを活用して選択関数を極大部分集合間できめることができないという問題点 4 を文の間の順序によって選択関数を構成するという手法を使って解消する.

文の間に異なった優先順位をつけるという考えは belief base C の要素上に, $\phi \preceq \psi$ と書く, 極大の要素を持った完全擬順序 (complete preorder) を導入することによって形式化される. ここで完全擬順序とはすべての $\phi, \psi \in C$ において $\phi \preceq \psi$ または $\psi \preceq \phi$ が成り立つ

ような反射的で推移的な関係のことである. $\phi \preceq \psi$ かつ $\psi \not\preceq \phi$ が成り立つ場合, $\phi \prec \psi$ とも書く. さらに少なくとも一つ極大要素 ϕ が存在する, つまり $\phi \prec \psi$ となるような要素 ψ が存在しない要素 ϕ が存在するものとする. この関係を epistemic relevance ordering と呼ぶ. $\phi \simeq \psi$ と書く, 同値関係を次のように導入する.

$\phi \simeq \psi$ であるのは $\phi \preceq \psi$ かつ $\psi \preceq \phi$ が成り立つ場合, そしてその場合に限る.

同値類を $\bar{\chi}$ で表し, これを C の epistemic relevance の等級と呼ぶ. 同値類の集合 $C/\simeq$ は $\bar{C}$ によって示される. 擬順序は完全なので, $\preceq$ は $\bar{C}$ 上の線形順序である. さらに, この擬順序は極大要素を含むので線形順序には極大の等級が存在する.

epistemic relevance を順序として持つ belief base は優先づけられたベース (prioritized base) と呼ばれる. その belief base が有限である場合, 我々は C の epistemic relevance の n 等級を示す表記法 $C_1, ..., C_n$ を使う. ここでこの表記法では C_1 がもっとも高い順位を持っているとする

epistemic relevance を使って, $C \Downarrow \phi$ で表記される, C から ϕ の優先づけられた削除 (prioritized removal) は C の部分集合の集合 S として定義される. それぞれの要素 $B \in S$ は epistemic relevance のすべての等級の部分集合からなる和集合である.

$$B = \bigcup\{B_{\bar{\chi}}\}_{\bar{\chi} \in \bar{C}}, \text{ ここで } B_{\bar{\chi}} \subseteq \bar{\chi}$$

形式的に, $B \in (C \Downarrow \phi)$ であるのは次のような場合であり, そしてその場合に限り成り立つ.

1. $B = \bigcup_{\bar{\chi} \in \bar{C}} B_{\bar{\chi}}$
2. すべての $\bar{\chi} \in \bar{C}$ について, $B_{\bar{\chi}} \subseteq \bar{\chi}$, そして
3. すべての $\bar{\chi} \in \bar{C}$ について, $B_{\bar{\chi}}$ は $\bigcup_{\bar{\psi} \succeq \bar{\chi}} B_{\bar{\psi}} \not\vdash \phi$ となるような $\bar{\chi}$ の部分集合の間で集合が包摂する要素の数が極大になるものである.

直観的に, $C \Downarrow \phi$ の要素は epistemic relevance の最大の等級から ϕ を含意しない極大部分集合を選択し, それから ϕ を含意しないような次に重要な等級の極大部分集合を加え, 以下同じことを繰り返して構築される. しかしながら, $C \Downarrow \phi$ の要素を構築することについてのこの直観は一般的な場合においては失敗する. 我々は epistemic relevance に制限を

加えていないので, epistemic relevance の等級の無限に上っていく鎖が存在する場合が起こりうる. しかし, またこの場合においても上記の条件を満足する B の要素の存在はツォルンの補題によって保証される.

prioritized removal の操作は与えられた命題を含意しないあるベースの極大部分集合の部分集合を選択する.

命題 ベース C と relevance ordering $\preceq$ があたえられて, すべての ϕ について

$$(C \Downarrow \phi) \subseteq (C \perp \phi)$$

このように $\Downarrow$ を $\perp$ の選択関数として使用することによって問題点 4 を解消することができる. また等級を分けることによって極大部分集合を求める手続きを一部軽減できるので問題点 3 の解消にもなる.$C \Downarrow \phi$ と belief base を使って prioritized base revision $\hat{+}$ を定義すると以下のようになる.

(prioritized base revision) $C\hat{+}\phi = \left(\bigcap_{B \in (C \Downarrow \neg\phi)} B\right) + \phi$

もし C が belief set ならば, 上記の partial meet contraction と Levi Identity を用いた revision の定義と同じになる.

以上のようなベースを用いた修正がどのように作業するか, 次の例を見てみることにする. ある人物があなたに彼が泳ぎにビーチへ行ってきたと告げ, あなたは日が照っていたのを観察しているとする. さらにあなたは日が照っているときにビーチへ泳ぎに行ったならば日焼けをするだろうと固く信じている. もしあなたがその人物が日焼けしていないのを発見したならば, 解決しなければならない矛盾が発生する. そこで次の命題を仮定する.

$$b = \text{“going to the beach for swimming”},$$
$$s = \text{“the sun is shining”},$$
$$t = \text{“sun tan”},$$

この状況は prioritized base C によって形式的にモデル化され得る.

$$C_1 = ((b \wedge s) \rightarrow t),$$

$$C_2 = s,$$

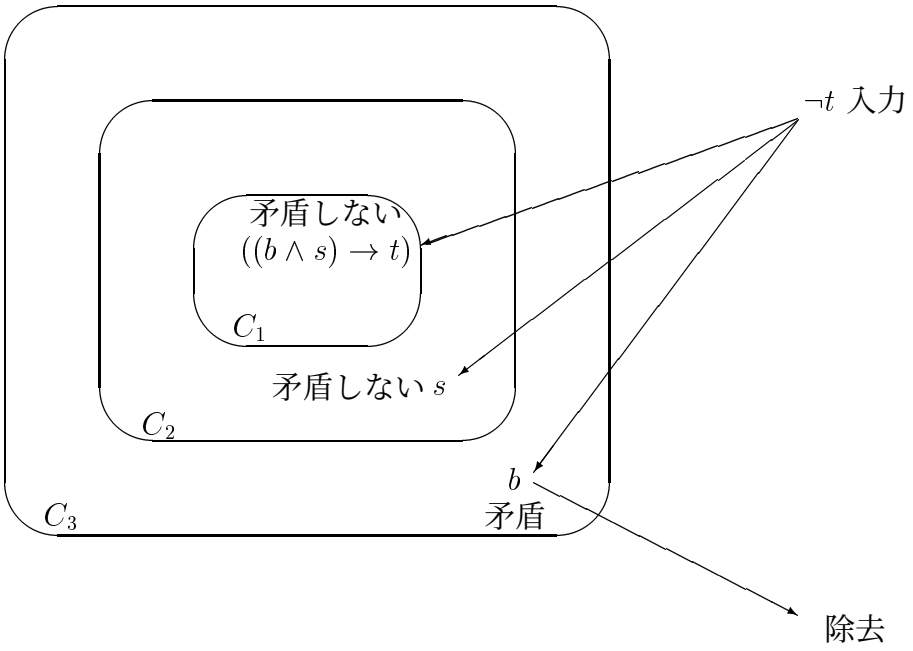


図 3.1: prioritized base revision の例

$$C_3 = b,$$

$$C = C_1 \cup C_2 \cup C_3$$

この belief base から t が導出され得る. もし我々があとで $\neg t$ を観察したならば, 図 3.1 のようなやり方でこの belief base は以下のように修正される.

$$\begin{aligned} C\hat{\oplus}\phi &= \bigcap(C \Downarrow t) + \neg t \\ &= \bigcap\{((b \wedge s) \to t), s\} + \neg t \\ &= \{((b \wedge s) \leftarrow t), s, \neg t\} \end{aligned}$$

このように, 我々は b は嘘であると結論づけるだろう.

このような知識間に順序をつけるという考えは Gardenfors and Makinson(1988) にも存在していたが, 彼らの epistemic entrenchment は以下のような公準を要求していた.

(EE1) もし $\phi \leq \psi$ かつ $\psi \leq \chi$ ならば, $\phi \leq \chi$

(EE2) もし $\phi \vdash \psi$ ならば, $\phi \leq \psi$
(EE3) 任意の ϕ と ψ について, $\phi \leq \phi \wedge \psi$ または $\psi \leq \phi \wedge \psi$
(EE4) $K \neq K_{\perp}$ であるとき, $\phi \notin K$ であるのは, 任意の ψ について, $\phi \leq \psi$ が成り立つ場合, そしてその場合に限り成り立つ
(EE5) もしすべての ψ について $\psi \leq \phi$ ならば, $\vdash \phi$

この公準に対して contraction の操作に以下のような条件を加えたとする.

($C \leq$) $\phi \leq \psi$ であるのは $\phi \notin K\dot{-}\phi \wedge \psi$ または $\vdash \phi \wedge \psi$ の場合, そしてその場合に限り成り立つ.

この条件を満たすとき, 順序が (EE1) から (EE5) を満足することと ($K\dot{-}1$) から ($K\dot{-}8$) を満足することは同値になる. しかし以上のようなやり方は実装向きではない. 以上のような公準に基づいて知識ベース内の知識間の順序を決めなければならないのは難しい上に, 条件 ($C \leq$) を満たす操作の定義が不明である.

3.3 矛盾の検出法

前節において epistemic relevance の導入によって多くの実装に当たっての問題点が解消されたが, しかし prioritized removal の定義の 3 の個所において論理的帰結を導く $\vdash$ を想定しているのでやはり問題点 2 の部分集合が入力文の否定を演繹するかしないかのチェックを論理的帰結を導く自然演繹法のような手間のかからない方法をどうやって実装するかという問題が残されている. そこで「部分集合が入力文の否定を演繹するかしないか」ではなく「部分集合内の文全体と入力文は矛盾するかしないか」というチェックに方法を変えて考えてみる. つまり入力文と部分集合が矛盾すれば入力文の否定を部分集合が演繹することが背理法よりわかる.

このことを確認するために導出原理を用いることにする. 導出原理は節集合が充足不可能であることを効率よく判断するための方法である. これまで扱ってきたすべての論理式の集合は節集合に変換できるので節集合が充足不可能であることと論理式の集合が充足不可能は同値なので, つまり論理式の集合が矛盾を演繹することと同値になる. そこで以下において導出原理を使った矛盾の検出法をのべる.

まず部分集合内と入力文の各論理式を論理積標準形に変換する. 論理積標準形とは $C_1 \wedge ... \wedge C_m$ の形をした式のことでここで各 C_i は $L_1 \vee ... \vee L_n$ の形をした節 (clause) と呼ばれ

る論理式である. 各 L_j は原子式ないしは原子式の否定でリテラル (literal) と呼ぶ. 各論理式は必ず同値の論理積標準形に変換できることが知られている. 次に各論理式から得られた節集合 $\{C_1, ..., C_m\}$ の和集合をとって論理式全体の節集合をつくる. この集合が充足不能であることを調べることによって矛盾を検出する. そこで導出原理を使用するのだが, それをここで説明する.

正リテラル L に対して, $\neg L$ を L の補リテラル (complement literal) と呼び, 一方また, L を $\neg L$ の補リテラルと呼ぶ. C_1,C_2 を節とする. C_1 がリテラル L_1 を含み,C_2 が L_1 の補リテラル L_2 を含むとする. C_1 から L_1 を除いた節と C_2 から L_2 を除いた節の論理和を取り, さらに重複して現れるリテラルを一つにまとめた節を C_1 と C_2 の導出形 (resolvent) と呼ぶ. またこの操作を導出 (resolution) と呼ぶ. 互いに補リテラルとなる節, たとえば A と $\neg A$ に導出を適用するとリテラルのない節が生成されるが, これを $\square$ と記述し, 空節 (empty clause) と呼ばれる. ここで以下のことが言える.

命題 節 C_1 と C_2 の導出形は, $\{C_1, C_2\}$ の論理的帰結である.

今もし節集合 Γ から空節 $\square$ が導出されたならば, $\square$ はある A と $\neg A$ から導出されたはずであるから, もし Γ が充足可能だとすると A と $\neg A$ の両方が充足可能だということになってしまうが, これはおかしい. したがって Γ から空節 $\square$ が導出されたならば, Γ は充足不能である. 節集合 Γ が充足不能ならば, この節集合と同値となる元々の論理式の集合も充足不能となり, したがって入力文と部分集合との矛盾が示せたことになる.

これを使って以前に使った「泳ぐためにビーチへ行った」と証言する人物の話を例にして実装における revision の方法を見てみることにする. まず例のように「日焼けをしていない」という情報が与えられる前の知識ベースに epistemic relevance$((b \wedge s) \rightarrow t) \succeq s \succeq b$ が与えられている, したがって以前のような prioritized base になっているとする. そして矛盾のチェックのために節集合に変換する. そうなると prioritized base は以下のようになる.

$$C_1 = (\neg b \vee \neg s \vee t),$$

$$C_2 = s,$$

$$C_3 = b,$$

$$C = C_1 \cup C_2 \cup C_3$$

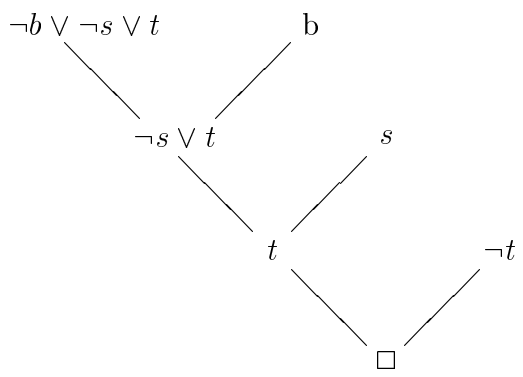


図 3.2: 空節の導出の例

これだけでは空節は導かれないので矛盾していない. しかしここに $\neg t$ が入力されると図 3.2 のように空節が導かれるので矛盾が発生する.

そこで revision が実行されなければならないことになる. revision の実行は prioritzed base の優先順位の高い順に行われるのでまず $(\neg b \vee \neg s \vee t)$ と $\neg t$ は空節を導かない, つまり矛盾しないので $(\neg b \vee \neg s \vee t)$ は保持の候補になる. そして s は $(\neg b \vee \neg s \vee t)$ と $\neg t$ に矛盾しないので s は $(\neg b \vee \neg s \vee t)$ と同様に保持の候補になる. 最後に b を加えると s, $(\neg b \vee \neg s \vee t)$ と $\neg t$ に空節が導かれてしまって矛盾するので排除されるのは b になる. これは矛盾が発生しない極大無矛盾集合なので最終的に知識ベースとして保持されるのは

$$\begin{aligned} C\hat{\oplus}\neg t &= \{(\neg b \vee \neg s \vee t), s, \neg t\} \\ &= \{((b \wedge s) \rightarrow t), s, b\} \end{aligned}$$

となる.

3.4 実装について

以上の議論を通じて実装の準備ができた. その方法を以下にまとめる.

- 知識ベースとしては belief set のような可算無限個の要素を含むような集合ではなく, belief base を使って有限個の集合で収まるようにする. 有限個に限定することによって知識ベースとしての実装が可能になる.

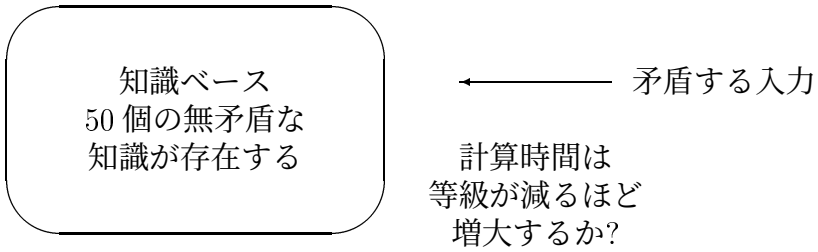


図 3.3: 実験

- 知識間に epistemic relevance である完全擬順序を割り当てる. 実装に当たってはいちいち知識間ごとにこの順序を割り当てるのはめんどうなので各知識に数値を割り当て, その大小で持って順序をつけることにする.
- 入力文の否定が知識ベースから導出されるかどうかのチェックは入力文と知識ベースから矛盾が導かれるかどうかのチェックに変えて, そのチェックに導出原理を用いる. これによって論理的帰結の操作を実装しなくても $C \Downarrow \neg\phi$ の操作が実行可能になる.

以上の方法で prioritized base revision を Java を言語として実装を行った. 以下の 3 つのファイルを作成した.

- 操作の実行に必要な BeliefRevApp.java
- revision の操作について記述してある EpistemicRelevance.java
- 知識ベースの管理に用いる KnowledgeBase.java

3.5 実装に基づく実験と問題点

以上のような実装を行ってみたものの, 依然として問題は残ってしまう. それはもしも epistemic relevance の等級において同じ等級内に同値となる多くの要素が存在していた場合, 計算時間はどの程度かかってしまうのかということである. 知識が様々な等級に分かれていたとしても, もしも同値類であるそれぞれの等級内に多くの要素が存在していた場合, やはりそれは計算の指数的増大を招く恐れがある. そのことを調べるために実験を行っ

表 3.1: Prioritized Base Revision の計算時間

同値となる要素数	同値類 $\overline{\chi}$ の数	計算時間 (ミリ秒)
25	2	1 週間以上
10	5	449438.0
5	10	17887.0
2	25	2487.0
1	50	1806.0

てみる.

実験

まず 50 個の無矛盾な論理式の入力を知識ベースに加えておく. そして 51 番目の入力として知識ベース中のある 1 つの知識を除いてほかのすべての知識と矛盾するようにする. すると, その 50 個の中で同値となる要素の数が増えて, 等級の数が減ってくると計算時間はどうなるか (図 3.3)?

この実験結果は表 3.1 のようになる. 50 個の要素をいくつかの等級となる同値類 $\overline{\chi}$ ごとに同じ数づつ分けて実行してある. すると一つの等級内の要素数が増えて等級の数が減るほど, いかに同じ数だけの知識を相手にした実験だとしても計算時間が指数的に増大していくのがわかる. したがって, 以下やはり計算時間の問題をどうするか, 考慮する必要がある.

第 4章

知識の順序の根拠とモデル化

これまでの章において epistemic entrenchment や epistemic relevance などの知識に順序をつけることによって信念の変更後にどの知識を優先して残しておくかという基準をもうける方法が現れた. しかし, これらの順序を定める基準はいったい何であるのか. この章においては Cantwell(1998) を通じて議論する. 4.1 節において Cantwell の与えた知識の優先順序の根拠を述べる. 4.2 節ではその根拠であるところの情報源の信頼度に基づいた意味論的な信念の更新方法について説明する. 4.3 節では Cantwell の方法の問題点を考察し, Cantwell が考案することができなかった情報源の信頼度の決定方法について 4.4 節で述べる. 4.5 節では前章の最後に現れた計算時間の増大の解消法を提案し, 4.6 節において情報源の信頼度の決定法とそして知識の優先関係の決定法を数理的にモデル化する.

4.1 Cantwell の提案

Cantwell(1988) の指摘するところではこの信念変更に関する研究においては, そのほとんどがいくつかの信念を他の信念よりも重要なものであると仮定し, 信念を変更するとき, そのような概念を考慮にいれて考えている. このような仮定はある形式的な言語における文として考えられるか, 言語のある解釈における命題として考えられる. この時, 信念の相対的な重要さは

(i) 言語の単一な文

(ii) そのような文の集合

(iii) 可能世界のような意味論的概念

のいずれかの関係によって捕らえられる. (i) の代表としては上記で説明した Nebel(1992), Gardenfors and Makinson(1988), (ii) としては AGM(1985), Doyle(1991), (iii) としては

Katuno and Mendelzon(1991), Friedman and Halpern(1997), Meyer et al.(2000) がある. 興味深いのはこのような概念が信念変更以外の非単調推論などのジャンルでも研究されていることである.

Cantwell が主張するには, このように知識に順序関係をつける研究は多いにもかかわらず, そのような関係がどこから生じるのかということについて説明する研究がほとんど存在していない. だがそういった研究が存在していないのはむしろ当たり前のことのように思える. なぜならたとえ知識や信念の関係といった尺度が与えられたときに, そのような尺度がどうして生じたのかという疑問が存在しなくても, そのような尺度だけをとっても十分に複雑で興味深いものであることがいえるからである. しかし, 文や可能世界間の関係はそれだけでは抽象的な定義でしかなく, したがってどのようにそれらが生じうるのかということを考えなければいったいそれがどういう意味があって関係付けられるのか分からなくなってしまう. この問題を信念変更のさらに一般的な問題への単なる応用のレベルでのみ存在するものとして考えることを放棄するのは正しい態度ではない. Cantwell が合理的な信念変更の説明はどのようにしてエージェントが彼らの持っている信念を変更すべきか, またはすべきでないのかという説明が与えられるまで完全なものとはならないと主張している理由がここにある.

しかし, 信念の評価が産み出されるメカニズムには多くの要因が含まれているので, 直に研究するのは極端に難しい. これがなぜ Cantwell がこの問題のとても小さな部分しか意図して見ていないのかということの理由となっている. しかしたとえ小さくても, Cantwell が選んだ以下の事柄は信念の評価にとって中心的な事柄である. その事柄とは「どうやって我々は信頼度に基づいて情報源からの情報を評価しているのか」ということである. そこで Cantwell は文の上での評価方法から情報源の上での評価方法へと移行する. ここでこのような考え方が持っている問題点を Cantwell は二つあげている.

1. ここで我々が全面的に依存しようとしている概念は評価の基準となるいくつかの概念のうちのほんの小さな段階のひとつに過ぎない.

2. どうやって我々は情報源を評価する基準を生じうるのか, その方法が全く説明されていない.

しかし, 彼はここで生じた考え方は調査を正当化するのに十分なほど興味深いことであると考えており, 一番目の問題点に関しては本論文でも文の間の評価の根拠に関して「文の間の優先順位を決定するのは情報源の信頼度である」という考えに限定したアプローチ

を採用して議論していくことにする. しかし二番目の情報源を評価する基準についてはある回答を与えることによって議論できると本論文では考える.

4.2 Cantwellの方法

Cantwell のとった情報源の信頼度から知識の優先順位を決定する方法は信念の評価基準の 3 つの分類のうち 3 番目の可能世界を考える方法に近い. 彼の考えた方法は以下の通りである.

ある情報源 w_1 が我々に ϕ であることを知らせたとき, その情報源は我々に ϕ が保持しないような状態にある世界の可能性を排除するように告げている. つまり, これは $\neg\phi$ が保持するようなあらゆる可能な状態を削除するように告げているのと同じである. ここで, w_0 という別な情報源から $\neg\phi$ という情報が前もって告げられていたとすると w_1 の情報は存在する可能な状態をすべて排除してしまうことになるので, これをどうやって調整するのかという問題が発生してしまう.

ここでオーバーライドの原理というものが考えられる. もし我々が情報源 w_0 が基礎となっている可能状態の集合を削除することを決定し, それと同じだけの信頼度を持っている別の情報源 w_1 が我々に可能状態の異なった集合を削除するように告げてきたら, 我々は w_1 の告知も w_0 の時と同様に気を使うべきである. もし上記のような場合において, 我々がすべての可能状態を削除することがわかったら, 我々はこの戦略だと失敗するので, w_0 と w_1 に等しい不信感を持って, どちらの告知にも気をつけないことにする.

上記の戦略を採用して信念変更の操作を定義する. まず基本的な概念を以下のように導入する.

U	世界の可能状態の空でない集合
V	言語におけるそれぞれの命題文 p について, $V(p) \subseteq U$ であるような評価関数
$M = (U, V)$	言語についてのモデル

そして論理式 ϕ の真理集合 (truth-set) または内包 (intension) である $\|\phi\|_M$ は標準的な作法によって定義する.

$$\|p\|_M = V(p)$$

$$\|\phi \wedge \psi\|_M = \|\phi\|_M \cap \|\psi\|_M$$

$$\|\neg\phi\|_M = U - \|\phi\|_M$$

ここで Γ は論理式の集合で,

$$\|\Gamma\|_M = \bigcap\{\|\phi\|_M : \phi \in \Gamma\}$$

が成り立つ.

X が状態の集合であるときはいつでも, 我々は $Th(X)$ を X におけるあらゆる状態において真であるような文の集合を示すものとする.

$$Th(X) = \{\phi : X \subseteq \|\phi\|_M\}$$

任意の状態 $u \in U$ について, $R_l(u)$ を u を削除する情報源の集合を示すものとする.

$$R_l(u) = \{w \in W : u \notin \|I(w)\|_M\}$$

$R_l(u)$ は情報源の集合なので, したがって信頼度の関係を状態間の順序を導くために使うことができるようになる. たとえば, $R_l(u) < R_l(v)$ は u を削除する情報源の集合は v を削除する情報源の集合よりも結合した信頼度が低いことを意味している.

$\mathcal{R}$ を情報状態 l を基盤として削除される状態の集合とする. 我々は前もって $\mathcal{R}$ においてどの状態が存在すべきかをいうことができないが, $\mathcal{R}$ が満たすべきたしかな性質なら知ることができる.

定義 $\mathcal{R}$ が l に関して削除可能クラス (possible rejectance class) であるのは, 次のような場合であり, そして次のような場合に限る.

1. $U \not\subseteq \mathcal{R}$
2. $\forall u, v \in U$,if $u \in \mathcal{R}$ and $R_l(v) \not< R_l(u)$, then$v \in \mathcal{R}$

つまり, あらゆる状態が削除可能だというわけではなく, かつ, もし u が削除されて v が r よりも信頼できないならば,v も同様に削除される.

このように我々は削除される (rejected) 状態の概念を定義した. 我々はここから受理される (accepted) べき状態が何であるかということについて何かいえるのだろうか. ここで Cantwell は次のような少し弱い受理の解釈を提案する. つまり, 状態 u が受理されるのは u がまだ削除されていない場合である. この解釈を使って受理された状態の集合 $\mathcal{A}$ を次のように定義する.

定義 $\mathcal{A}$ が l に関して受理可能クラス (possible acceptance class) であるのは, 次のような場合であり, そして次のような場合に限る.
$\mathcal{A} = U - \mathcal{R}$

削除のための上記の二つの性質は対応して受理のための二つの性質を生じる.

定理 上記の定義より次が成り立つ.

1. $A \neq \emptyset$
2. $\forall u, v \in U$,if $u \in \mathcal{A}$ and $R_l(u) \not< R_l(v)$, then$v \in \mathcal{A}$

あと, 受理可能クラスは包含関係について線形順序であることもわかっている. ここから文の間の順序関係,Cantwell がいうところの支持関係 (support relation) に当たるものが定義できる. 任意の二つの状態 u と v について, もし $R_l(u) \leq R_l(v)$, つまり v を削除する情報源が u を削除する情報源と少なくとも同じだけの結合した信頼度を持つならば, 状態 u は状態 v に少なくとも同じだけの支持を持っているということは道理にかなう. さらにいうと, このような情報が手元に与えられたとして, 状態 u は状態 v と少なくとも同じだけの正しい蓋然性があることになる. つまり以下のようになる.

- $v \leq_{prob} u$ であるのは $R_l(u) \leq R_l(v)$ である場合, そしてその場合に限る.

さらに文の支持関係を以下のように定義することができる.

$\phi \preceq_l \psi$ であるのは, 次のような場合であり, そして次のような場合に限る.
(i) $\forall u \in \|\phi\|_M \exists v \in \|\psi\|_M, R_l(v) \leq R_l(u)$ かつ
(ii) $\forall v \in \|\neg\phi\|_M \exists u \in \|\neg\psi\|_M, R_l(v) \leq R_l(u)$

つまり ψ が ϕ と少なくとも同じくらいに支持されるのは, (i) すべての ϕ 状態について少なくとも同じくらいの支持がある ψ 状態が存在し, かつ (ii) すべての非 ψ 状態について少なくとも同じだけの支持がある非 ϕ 状態が存在する場合である. この関係は ϕ と ψ の状態の相対的な価値についてのみでなく非 ϕ と非 ψ の状態の相対的な価値も説明している. 以上のことから受理可能クラスと支持関係の間に次のことが成り立っている.

1. もし $\phi \in Th(\mathcal{A})$ ならば, $\neg\phi \prec_l \phi$.
2. もし $\phi \in Th(\mathcal{A})$ かつ $\phi \prec_l \psi$ ならば, $\psi \in Th(\mathcal{A})$.

ここで例をあげて実際にどのようにして信念変更が行われるのかを確かめてみる. ここでどのような信念が受理されるべきかということについて, 以下の規則に従うものとする.

(Gulibility) もし $\neg\phi \prec_l \phi$ ならば, $\phi \in A_l$

本当はこの規則には問題があることを Cantwell は指摘しているのだが, ここではそれには触れない. またそれぞれの情報源はその信頼度を反映した数と同一視され, 情報源の集合の信頼度は単にここの情報源の信頼度の合計と同じ物とする.

例 I を以下のような情報状態とする:

$$w_0 : \phi \quad w_1 : \neg\phi \quad w_2 : \psi$$

ここで $w_0 \sim w_1 \sim w_2 \sim 1$ とする.

$\neg\phi$	ϕ	
$w_0 = 1$	$w_1 = 1$	ψ
$w_0 + w_2 = 2$	$w_1 + w_2 = 2$	$\neg\psi$

ここで我々は $U = \{\phi \wedge \psi, \phi \wedge \neg\psi, \neg\phi \wedge \psi, \neg\phi \wedge \neg\psi\}$ とする. 明白に $R_l(\phi \wedge \psi) \sim R_l(\neg\phi \wedge \psi) < R_l(\phi \wedge \neg\psi) \sim R_l(\neg\phi \wedge \neg\psi)$ である. このようにすると二つの削除クラス $\mathcal{R}_0 = \emptyset$ と $\mathcal{R}_1 = \{\phi \wedge \neg\psi, \neg\phi \wedge \neg\psi\}$ が存在し, そしてそれゆえ二つの受理クラス $\mathcal{A}_0 = U$ と $\mathcal{A}_1 = U - \mathcal{R}_1 = \{\phi \wedge \psi, \neg\phi \wedge \psi\}$ が存在する. Gulibility によれば我々は $\mathcal{A}_1$ を選択する. 受理された文の集合は $Th(\mathcal{A}_1) = Cn(\psi)$ となる.

以上のような方法を用いることで情報源の信頼度からどの文を価値あるものと見なすかということを決定し, そこから受理されるべき文の集合を見つけることができる.

4.3 Cantwellの問題点

以上のようにして情報源の信頼度から受理される文の集合を決定する仮定を定義したのだが, この Cantwell の意味論的な方法には問題がある. まず, もしも情報源が「p であり, かつ $\neg p$ である」のような自己矛盾を起こす発言をした場合, それに対応する可能な状態が存在しないので計算することができなくなってしまうことである. 次にそれぞれの状態, つまり情報源の集合の信頼度を決定するために単純な例としてそれぞれの情報源の信頼度の合計を用いるという方法をとっているが, 実際にはそのような結合した信頼度を決

定するのは非常に難しい.

そしてなによりも Cantwell 自信が問題点として指摘していることは話題に応じて単一な情報源の信頼度が変化してしまうことである. たとえば, 時間の経過は「ビルは風邪を引いている」といったたぐいの情報が, 一年も前に起こったことなら無効化してしまう. Cantwell は異なった情報源の信頼度を評価する方法のダイナミクスもまた重要な問題だと思っている. これは先に説明した情報源の信頼度を決定する方法が Cantwell においては存在しないという問題とつながってくる.

ところでどうして Cantwell は Nebel や AGM のようなシンタックスの立場からのアプローチではなくセマンティクスの立場からのアプローチをとったのか. Cantwell はシンタックス用の信頼度からの文の順序関係の決定方法を定義しているのだが, しかしある問題が発生していると Cantwell は主張しているためにシンタックスからのアプローチを破棄している.

ではその問題とは何であるかを見るために Cantwell 流のシンタックスの方法を見ることにする. まず Cantwell が考えているところの「支持」の概念についてのべる. 「支持」の概念は異なった読み方によって幅広い変化を与えることができるが, Cantwell はここではっきりとした考え方をしている. つまり, 支持とは次のような規則を持つ概念である: もし文 ϕ がその否定より支持を持つならば, それは重要な受理の代表となる. 先に見た規則 Gulibility はこの規則に支えられている.

まず文 ϕ がある情報状態 I から支持を得るためのいくつかの異なったあり方を見ることにする. もっとも明らかな場合はある証言者 w が実際に ϕ を証言している場合である:これを $\phi \in I(w)$ で表す. この場合において, ϕ は直接的な支持 (direct support) を I に持っているという. この概念の自然な拡張は ϕ が直接的に支持されているときはいつでもその論理的帰結も直接的に支持されているということである. 我々は文の集合からの論理的帰結についても同じように情報源からの直接的な支持を得ていると考える.

このような直接的な支持だけが支持の唯一の形態だというわけではない. もし $\phi \in I(w_1)$ かつ $\phi \to \psi \in I(w_2)$ ならば, ψ は $\phi, \phi \to \psi \vdash \psi$ によって間接的な支持 (indirect support) を持っているということができる. それをいうための別の方法は ψ について I において情報の鎖 (chain of information) が存在すると考えるものである. 我々はそれを定義するため

に次の概念を使う.

定義 任意の論理式の集合Γについて$\Gamma \Rightarrow \phi$であるのはつぎの場合であり, つぎの場合に限る

$$(i) \quad \Gamma \vdash \phi$$
$$(ii) \quad \Gamma \nvdash \bot$$
$$(iii) \quad \forall \psi \in \Gamma, \Gamma - \{\psi\} \nvdash \phi$$

つまり$\Gamma \Rightarrow \phi$であるのはΓがϕを含意する極小無矛盾集合である場合, そしてその場合に限る.

定義 Cがϕについての情報の鎖であるのは, Iに関して$C \subseteq I(w_1) \cup ... \cup I(w_n)$かつ$C \Rightarrow \phi$であるような証言者$w_1, ..., w_n$が存在する場合, そしてその場合に限る.

ある情報状態においてϕについての異なった情報の鎖がいくつも存在するかもしれない. ϕについてのIにおける情報の鎖の集合を$\mathcal{C}^{\phi}$で表す. ただし, 二つの鎖は矛盾した情報を主張しているかも知れないことに注意.

情報の鎖の扱いはCantwellによると古い歴史を持っているそうだが, 二つの考えが顕著である.

1. 証拠の鎖の強さはそのもっとも弱いリンク (鎖の任意の要素のこと) の強さよりも決して大きくならず, そして実際には, 弱くなる.

2. 独立した証拠の鎖の強さは付加的である.

この説明は「強さ」と「独立」という定義されていない概念を使っているために曖昧に見えるが, もし我々が「強さ」を確率的概念として扱うならば, (1) を動機づけるのを見るのはたやすいことである:もし事象eを建設するために独立した事象e_1とe_2を建設しなければならないならば, eの確率はe_1とe_2の確率の生産物となる. これらが確実に起こるわけではない事象であるならば, つまり$P(e_1) < 1$かつ$P(e_2) < 1$ならば, $P(e) = P(e_1) \times P(e_2) < P(e_1)$. これが (1) が情報の鎖の倍増 (multiplicative) 効果と呼ばれている所以である.

しかし, ここではこのような確率的なやり方はとらずに, ϕについての情報の鎖の基礎の上に直接的にlにおけるϕについての支持の概念を定義したい. 基礎入力は文についての直接的な支持によって与えられる.

$$DS_l(\phi) = \{w : \exists \psi_1, ..., \psi_n \in I(w), \psi_1, ..., \psi_n \vdash \psi\}$$

そして l における ϕ についての全体的な指示は次のように与えられるだろう.

$$S_l(\phi) = \bigcup \{X_C : C \in \mathcal{C}^\phi\}$$

ここで X_C は $min_{\leq}(\{DS_l(\psi) : \psi \in C\})$ のいくつかの要素である $S_l(\phi)$ は ϕ についての可能ないくつかの情報の鎖の結合力を表している. この概念は全く鎖の倍増的な観点を正当化していないことに注意. ここでは鎖によって与えられる支持は鎖のもっとも弱い要素の支持と等しくなる. これは必ずしも数値でもって作業するわけでなく, 代数的な操作を一般的に使うわけでもないので, これは粗い近似に見えるかも知れないと Cantwell はのべている. しかしむしろ問題になるのはそういうことではなく, ここから述べることである. 以上のような支持の概念を基礎として関係 $\preceq_l$ を定義することができるかどうか. Cantwell は以下のようなやり方では十分ではないと主張する.

$\phi \prec_l \psi$ であるのは $S_l(\phi) \leq S_l(\psi)$ である場合, そしてその場合に限る.

これが十分ではない理由として, Cantwell は上記の例をとって問題点を指摘しようとする.

例 1 I を以下のような情報状態とする:

$$w_0 : \phi \quad w_1 : \neg\phi \quad w_2 : \psi$$

ここで $w_0 \sim w_1 \sim w_2$ とする. ϕ と ψ は論理的に独立であると仮定する. そのとき $\psi \approx_l \phi$ である. $\neg\psi$ が全く支持されていない一方で $\neg\phi$ が ϕ と同じくらい支持されているという事実があるにも関わらず, ψ が ϕ と同じくらい支持を持っていると考慮されている.

以上のようなことが問題だというのは少し分かりにくいが, それはこのやり方が元々の支持の概念の定義「もし文 ϕ がその否定より支持を持つならば, それは受理の重要な代表となる」にとって都合が悪いようにできているからだとするとわかりやすくなる. $\neg\psi$ が支持されていずに ψ が支持されているのだから ψ は受理の対象となってもいいはずである. ところがその受理の対象となるべき ψ と同じくらいの支持を ϕ と $\neg\phi$ が持ってしまってい

るために, 矛盾した文も受理の対象になってしまうのかということになってしまう. そこで別な関係 $\preceq_l$ の定義の仕方を考えてみる.

$\phi \prec_l \psi$ であるのは $S_l(\phi) \leq S_l(\psi)$ かつ $S_l(\neg\phi) \leq S_l(\neg\psi)$ である場合, そしてその場合に限る.

ここで少なくとも二つの異なった支持の概念を用いている. $S_l(\phi)$ は ϕ についての正の支持と呼ぶことができ, $S_l(\neg\phi)$ は ϕ についての負の支持と呼ぶことができる. そして関係 $\preceq_l$ は正の支持と負の支持によって比較に関する統合的な支持の関係となる. ところがこの方法にも不備があると Cantwell は主張する. それは次のような例において問題が出てきてしまう.

例 2 I を以下のような情報状態とする:

$$w_0 : \phi \qquad w_1 : \phi \to \psi$$
$$w_2 : \neg\phi \qquad w_3 : \chi$$
$$w_4 : \chi \to \neg\psi$$

ここで $w_3 \sim w_4 < w_0 < w_1 \sim w_2$ とする. ここで ψ についての鎖における二つのリンクはそれぞれ $\neg\psi$ についての鎖における二つのリンクのそれぞれよりも直接的に支持されている. したがって以前に定義した統合的な支持の概念によると, ψ は $\neg\psi$ よりも支持を持っている. しかし w_2 のために我々は ψ についての鎖における ϕ リンクに疑問を持ち, この場合 ψ はその支持の多くを失わなければならない.

ところがその一方で χ に対する $\neg\chi$ は支持がないので $\neg\psi$ がその支持を失うことはない. このようにして Cantwell はシンタックスに基づく支持の概念の定義がうまく行かなかったためにセマンティクスを支持の概念の基盤とした.

しかし Cantwell の以上のような議論には穴があるように思われる. その穴とは以下の二点である.

1. Cantwell は支持の概念を「もし文 ϕ がその否定より支持を持つならば, それは受理の重要な代表となる」と定義しているが, そのように考えなければならない必然性はない. むしろ Nebel のように知識の順序は恣意的な基準で決められる方が自由で

あるように思われる. また上の例題ではすでに矛盾が発生している情報状態を用いているが AGM や Nebel では知識ベースと入力の間の矛盾は認めても, 知識ベースそれ自体の矛盾は認めない. よって AGM の epistemic entrenchment にせよ Nebel の epistemic relevance にせよ信念変更の操作の際に文 ϕ とその否定が比較の対象にはなり得ない. 二つの文のどちらか一方は知識ベースにおいて受理されていないからである. したがって最初の例のような問題は Nebel および AGM には発生しない.

2. Cantwell は鎖の概念を導入することによって明示的にのべられていない情報に対して間接的な支持が働くということにしたが, これを考えなくてはならない理由は Cantwell は epistemic state の論理的帰結はやはり epistemic state になることを想定している, つまり暗に base ではなく belief set を考えているからである. しかし信念変更を set ではなく base にしてしまえば論理的帰結を考えずに済むので. 鎖の概念を導入する必要がなくなる. したがって二つ目の例のような問題は Nebel においては発生しない.

以上の点によって Cantwell の方法の代わりに Nebel の epistemic relevance を知識の優先順位として用いることはある程度の正当性があると本論文では考える. そのことにどの程度の正当性があるかは最終的にシステムを実装して確認をとらなければならないがとりあえずは Nebel の epistemic relevance を使って情報源の信頼度を考慮した信念の更新を行うシステムというものを考えて実装に応用することにする.

4.4 情報源の信頼度の評価方法

前節において Cantwell が自らの説の困難として話題に応じて単一な情報源の信頼度が変化してしまうことを説明できないことをあげていたことをのべた. そこで実際に信頼度が変化するようなモデルを, つまり信頼度を決定する根拠となるようなモデルを考慮する必要がある.

ここでそのモデルは「知識ベースの更新の作業の結果得られた知識に対して正しい情報を提供した情報源の信頼度は上昇し, 誤った情報を提供した情報源の信頼度は下降する.」という考えに基づいて与えられるものとする. つまりエージェントが信念を修正する操作を行ってみて知識ベースに残るような知識を提供した情報源は正しい知識を与えたものと見なして信頼度を増し, その逆に知識ベースから捨てられるような知識を提供した情報源は間違った知識を与えたものと見なして信頼度を下げるということにする. このような

やり方は Cantwell や AGM のように belief set を用いて知識ベース内のある知識の論理的帰結をも知識ベース内にいれるものとして考えると, そのような間接的な支持を持つ知識が捨てられる場合, いったいその知識を得るために用いた元の知識の提供者となった情報源のうちどの情報源の信頼度を下げれば良いのかという問題が発生するので, base を用いた Nebel の考え方がもっとも適している.

ところでこのやり方はやはり問題を単純化して行っていることに注意したい. というのも, 先ほどの Cantwell があげた例「ビルは風邪を引いている」といった情報が一年後になって間違った情報になったところでこの情報の提供者の信頼度が下がってしまうとか, 情報源の信頼度が低くなったからこの情報の価値が低くなったということはあり得ない. これは「ビルは風邪を引いている」という情報が時間が経つにつれて真ではなくなっていくような種類の情報であるということをエージェント自身が背景知識として持っているからである. つまり情報源が提供した情報の中でも文脈によっては間違っていたとしてもそれがその情報源の信頼度を下げる理由にはならない場合や情報源の信頼度が高くても様々な要因から見てその情報が優先順位の低いものであると見なす場合が存在する. これは知識の優先順位は情報源の信頼度のみによって必ずしも決まるわけではないということの例証にもなるが, これは終章で述べることにして本論文ではこのような単純化された方法を用いることにする.

4.5 計算時間の問題の解決

前章の最後に計算時間の問題を Nebel では解消しきれていないことを述べた. この節ではこの問題を前節に述べた情報源の信頼度の決定の問題と合わせて議論をしてみることにする.

まず人間が解決するのに時間がかかりすぎる問題に直面した際の解決法を見る. 人間が何か物事を考えて時間が経っても結論が出ない場合に考えることを放棄してしまうことがある. これは人間が経済性の観点から見て正確な知識よりも時間を節約することを重要視する場合があることを示している. このような事例にしたがってもしもある入力によって生じる修正があまりにも時間がかかりすぎるものであったら, 修正を放棄してそのような問題を生じた入力のほうを破棄するという考えが生まれる.

このような考え方にどの程度の正当性があるのか. これを実際に Nebel の方法で照らし合わせて考えてみる. Nebel の方法で計算時間がかかる場合というのは一つの epistemic

relevance の等級内の要素数が多いときである. したがってもしあまり epistemic relevance において重要でない等級に要素が多く存在するばあい, 重要でない知識が多いために外部入力を捨てることになってあまり説得力がなくなってしまう.

ただし, もし重要である等級に要素が多く存在する場合, この考え方はかなりの正当性を持つ. つまり重要な等級の修正に時間がかかるということは重要な等級の信念の大部分を削除することにつながってくる. このようなことは不合理であるので逆に入力となる情報の方を切り捨てた方が良いということになる.

以上のような考察から理解できることは知識の等級が高いほど計算に時間がかかるような入力は疑わしいということであり, したがって知識が重要でない等級においては計算時間が多くてもかまわないということになる. よって等級に応じて入力の側を切り捨てることになる計算時間の設定は変える必要がある.

ただし, このやり方は知識の正確さを犠牲にして時間短縮を計るという方法なので, このやり方がどれだけ実装の実行の段階において影響をしてくるのか, あとで実験で確かめてみる必要がある.

4.6 モデル化

この節では以上の議論から情報源の信頼度を変更する仕組みのモデル化を行う. まず情報源の信頼度と知識間の順序関係, つまり epistemic relevance との対応関係についての定義であるが, ここでは Cantwell の最初の案「$\phi \prec_l \psi$ であるのは $S_l(\phi) \leq S_l(\psi)$ である場合, そしてその場合に限る.」をそのまま採用することにする. ただしここで用いている順序関係は base 内の知識に関する順序なので実際の $S_l(\phi)$ の定義のような間接的支持の鎖は存在しないことにする. よってここでの $S_l(\phi)$ の定義は

$$S_l(\phi) = \bigcup\{w : \phi \in I(w)\}$$

となる.

次にこれまでと同様にある添字つきで各情報源を w_i で表記し, その情報源が提供する知識 ϕ は情報状態 I によって与えられるものとする. つまりこれを $\phi \in I(w_i)$ によって表記する. これを持って各入力における修正ステップ数 j での情報源 w_i の信頼度 $w_{i,j}$ を以下

のようにする.

$$w_{i,j+1} = x_{i,j} w_{i,j}$$

$$x_{i,j+1} = x_{i,j} + c r_i w_{i,j}$$

ここで $x_{i,j}$ は添字に対応する各情報源の各ステップ数における重みであり, r_i は更新の結果決まる学習信号, c は定数である. ただし入力によって矛盾が生じない場合はすべての学習信号を 1 とし, c の値は矛盾発生時よりも小さくして学習信号の影響を小さくするものとする. ここで矛盾が発生したときの r_i を決める方法を前章までの議論をふまえて考案する.

まず更新の作業にあまり計算時間を使わない場合を考察する. 更新の結果削除される知識は入力ではなく知識ベース内の知識のいくつかである. この場合複数の知識が同一の情報源を持っていることがあり, その情報源が提供した知識のうちいくつかは保持されいくつかは削除されるとき, 単純に情報源が信頼できるものなのかできないものなのかを決定することはできない. よってここでは保持される知識が削除される知識を越える場合, つまり元々保持されていた知識のうちの 2 文の 1 以上が更新後も保持される場合その知識を提供した情報源は信頼度をあげることにし, そうでない場合信頼度を下げることにする.

このような考え方をモデル化するとまずステップ数 j での情報源 w_i の提供した知識によって構成される情報状態 I を $\phi \in I(w_{i,j})$ によって表記する. すると学習信号 r_i は以下のようになる.

$$r_i = \begin{cases} 1 & (|I(w_{i,j+1})| \geq |I(w_{i,j})|/2 \text{ である場合}) \\ -1 & (\text{それ以外}) \end{cases}$$

そして更新の作業に計算時間を使う場合を考察する. この場合削除される可能性のあるのは入力のみなので入力を提供した情報源の信頼度はもし入力が削除された場合には下がることにする. このとき入力以外の情報を提供した情報源の信頼度はもし入力が削除された場合には上がることにする. ということで, ここで必要とされるモデル化はどれだけの時間が経てば更新作業を打ち切って入力が削除されるのかを明確にすることである.

ここで計算時間が指数的に増大するのは epistemic relevance の各等級における知識が多くなる場合であることを振り返ると, 簡単にある一定時間が立てば入力を捨ててしまっても良いとは思えない. 知識が多くなるほど削除される可能性のある知識が増えるので計算

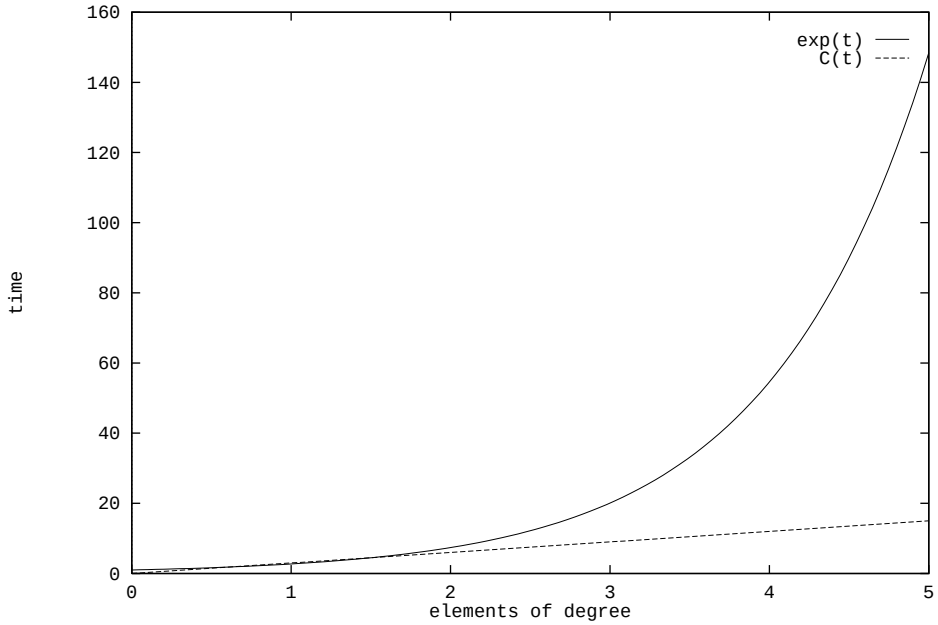


図 4.1: 計算時間の増大と制限時間の設定 (1)

時間が増すのは仕方のないことだからである. したがって指数的に計算時間が大きくなるのは困るが, 知識が多くなるほど計算時間の限界というのは大きくした方が良い. そこで図 4.1 のように計算時間が各等級における知識の数に対して計算時間が指数的に増大するときに更新の作業を停止する計算時間の限界を線形に増大するものとして考えることにする.

そして先ほど述べた通り等級によって制限時間を変えるのが望ましい. 知識の等級が高くなるほどこの制限時間によって更新の作業に限界を設定するという説の根拠は大きくなるので, その場合は制限時間は更新の対象となっている等級より低い等級の制限時間よりも短くて良い. 逆にいうと等級が低くなるほどこの節の根拠は小さくなるので制限時間は増大するものと考える. すると図 4.2 のように等級の上の方から順に C_1, C_2, C_3 とすると以下のように制限時間の設定はわかれる.

このような考え方をモデル化するとまず epistemic relevance の k 等級におけるステップ数 j での更新の計算時間 $T_{j,k}$ によって, そして epistemic relevance の各等級を C_k によって表記する. そして epistemic relevance の順序を情報源の信頼度の数値の順序と対応づけることにしたので各等級 C_k における順序に対応する信頼度を $Sl(C_k)$ とする. ここで情報源

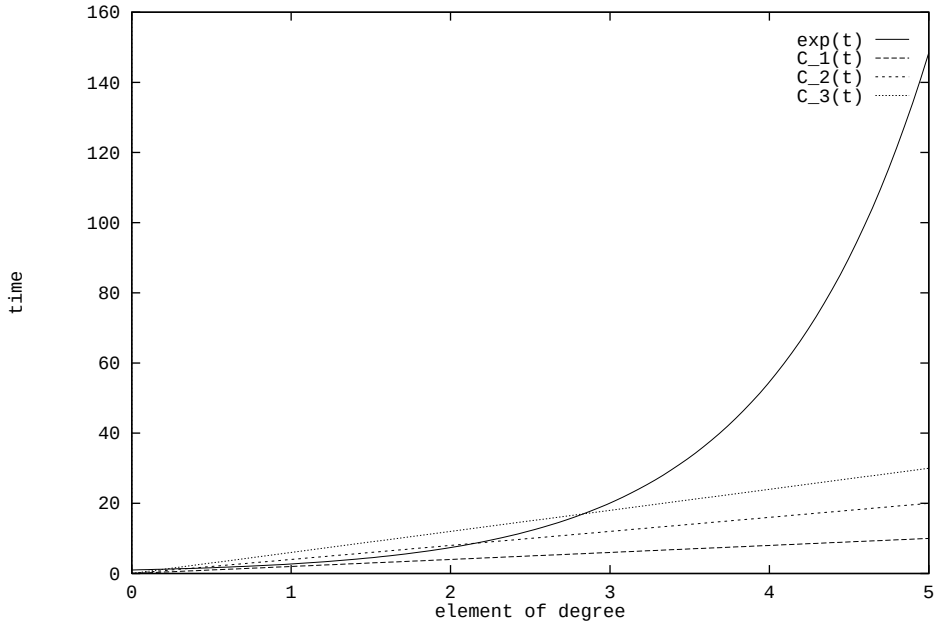


図 4.2: 計算時間の増大と制限時間の設定 (2)

w_i が入力の提供者であることを $Input(i)$ で表し, $a(ms/個数)$ を定数とすると学習信号 r_i は以下のようになる.

$$r_i = \begin{cases} 1 & (Input(i) \text{でないか, またはすべての } k \text{ について } T_j < (a/Sl(C_k)) * |C_k| \text{ の場合}) \\ -1 & (otherwise) \end{cases}$$

以上, 2 つの場合に分けて考えた学習信号 r_i をまとめると以下のようになる.

$$r_i = \begin{cases} 1 & (|I(w_{i,j+1})| \geq |I(w_{i,j})|/2 \text{であり, かつ } Input(i) \text{でない} \\ & \text{かまたはすべての } k \text{ について } T_j < (a/Sl(C_k)) * |C_k| \text{ の場合}) \\ -1 & (otherwise) \end{cases}$$

以上によって学習信号のモデル化とそれに基づく情報源の信頼度の決定方法, そしてそれに対応する epistemic relevance の順序の決定をモデル化した. このようにして決定された epistemic relevance を Prioritized Base Revision と組み合わせることで情報源に対する信頼度を変更していくシステムができる. そしてこれまでの実装を用いれば以上のことをその中に組み込むことは容易に可能である. その実装のプログラムとして前章で述べた EpistemicRelevance.java の代わりに ResolvingConflict.java というファイルを作成した.

第 5 章

実験

この章では前章までで述べたモデルの実装に基づいて, どれだけの信頼が持てるシステムなのか, またパラメーターの変動によってどれだけその信頼が保てるシステムなのかを検証するための実験を行うことにする. まず 6.1 節においてこのシステムが正しい知識を提供する情報源を見分けることがどうかをテストする実験方法を述べることにする. 6.2 節ではその実験の結果を示し, 最後に 6.3 節では結果からどういったことがこのシステムについていえるのか, その考察を行う.

5.1 実験方法

これまでの章によって情報源の信頼度を更新の結果から自律的に変更していくシステムの実装が可能となった. だがこのシステムはどの程度まで信頼するに足るものなのか, 試行を中止して計算時間の短縮を計ってしまうとどの程度の影響が出るのかどうかを検証してみなくてはならない. そこでこのシステムに関する実験を行ってみる.

まず正しい情報をそれぞれ異なった確率で提供する 4 つの情報源を想定する. それらの確率は固定であるとする. そして実際に正しい情報を提供する確率の高い情報源から順番にシステムに情報を提供していき, もっとも確率の低い情報源が情報を提供し終わったら, また確率の高い情報源から情報を提供することを繰り返していく. そしてシステムの側の情報源の信頼度の初期設定を等しくしておくことにして以上の操作を繰り返していくうちにどうなっていくのかということを確かめる. ここで正しい情報として何を想定するかということであるが各情報源は命題 A か B の二つの命題に関する言及のみをすることにし, 発言する可能性のある論理式は以下の 14 個であるとする.

1. $A \wedge B$
2. $\neg A \wedge B$
3. $\neg A \wedge \neg B$
4. $A \wedge \neg B$
5. A
6. B
7. $\neg B$
8. $\neg A$
9. $(A \vee \neg B) \wedge (\neg A \vee B)$
10. $(\neg A \vee \neg B) \wedge (A \vee B)$
11. $A \vee B$
12. $\neg A \vee B$
13. $A \vee \neg B$
14. $\neg A \vee \neg B$

このうち正しい文は 1,2,4,5,6,10,11 であり, 間違っている文は 3,7,8,9,12,13,14 であるとする. この基準は $\neg A \wedge \neg B$ を明らかに間違っている文と見なし, この文を正しい文と見なす可能性のない文を明らかに正しい文と見なすところにある. したがってここでは間違っている可能性のある文にはシステムはなるべく寛容な態度をとらず排除してくれることを期待している.

そして４つの情報源が正しい情報を提供する確率をそれぞれ $0.9, 0.75, 0.25, 0.1$ とし, 各情報源の信頼度ははじめはすべて 0.5 で設定されているものとし, 更新の実行回数は 200 回として議論する.

表 5.1: $a = 1000$ の結果

	agent1	agent2	agent3	agent4
1	1.113857939093245	0.2631358867038881	0.07833699028563992	0.0594166421908234
2	4.7844690476122	0.5489729880379306	0.17004391178685402	0.2420989215863491
3	1.643025718164268	0.4473973516279492	0.7123459279172242	0.4019630993588448
4	1.2744605153377289	0.5113540667934778	0.1495840058478837	0.24776140798606372
5	20.517179981099314	1.1403384658504319	0.8294147346595966	0.4304359620891735
6	0.6491383067074025	0.18793704449612286	0.17923318239855193	0.18829966706337994
7	7.222272849556334	0.28215497001591716	0.4613144791612412	0.16617351434261216
8	1.7962316523461093	0.1554693300163111	0.1652699555954593	0.09298557323513959
9	7.383645877195135	0.22442937988408157	0.1943498264524724	0.17341498731935137
10	0.9348847638631097	0.3259996478861357	0.10537910042881306	0.10582683630377433

5.2 実験結果

まず重み付けを決定する定数 c は矛盾が発生するときは $c = 0.001$ とし, 発生しないときは $c = 0.0007$ とするのが信頼度の上下動がもっとも安定する値であることがわかった. この値を動かすと情報源の信頼度が大きくなり過ぎて測定できなくなったり, すべての情報源の信頼度はゼロに向かって収束していくようになってしまう. よって以下の実験では定数 c は以上の値のまま行うことにした.

次に計算時間の増大を押さえる傾きの定数 $a(ms/個数)$ をいろいろ操作してそれぞれ 10 回試行する実験をした. 表 5.1 から 5.5 にそれぞれ $a = 1000, 500, 100, 50, 30$ の値についての更新実行後の各情報源の信頼度を表にした.

表 5.2: $a = 500$ の結果

	agent1	agent2	agent3	agent4
1	4.456183973267708	0.19560195306039566	0.16895342426061227	0.13018710127393504
2	23.150250238320528	0.23481103437011328	0.1991376310762167	0.13449114327345277
3	1.1823936163479691	0.7485455171916747	0.2119548909830919	0.6839230520128851
4	15.241501633379537	0.9787584137353912	0.3685615891905524	0.275906581596014
5	370.37798351668494	0.6174090569239568	0.3757634957692323	0.5319244839588154
6	39.43710579147299	0.5777705408612313	0.4843030000045543	0.6247236727219487
7	2.5551693106720994	0.5652732650518866	1.5573400781027325	0.37023734312143747
8	0.5678125314197646	0.08449364192139434	0.05441978477602495	0.19754382769306575
9	1.4253213067418737	0.694378632131025	0.17238692289778804	0.2440948334721415
10	13.863314087984607	0.15130269882417l9	0.1986145333640376	0.24151037183327564

表 5.3: $a = 100$ の結果

	agent1	agent2	agent3	agent4
1	7.60387137970046	0.13360680391696234	0.07385354532272734	0.19486160331884805
2	3.1418608100010337	0.13102450214169525	0.2325448586752769	0.09335827129759314
3	9.743011537110482	0.5558892059685684	0.3408954520368662	0.36942740260117074
4	0.15120863491466072	0.12421045665119497	0.12736288024414308	0.47123469563364767
5	0.9796694156072395	0.25169235132978807	0.15596687961394265	0.13943832988505359
6	1.5021642800783954	0.6739601221282921	0.21445222912898146	0.3451346163430737
7	2.168126177498434	0.3106237183091212	0.2841555780440137	0.2534486005439002
8	185.67746644167755	0.28247385889153975	0.5087035649097623	0.2217488180992336
9	428.94500765780623	1.620334923556838	0.558912021512175	0.5016970750979147
10	1.2662808635204845	0.1464348928118199	0.14347571450808552	0.17846167349035402

表 5.4: $a = 50$ の結果

	agent1	agent2	agent3	agent4
1	2.1307682176475105	2.1603180998399014	1.513829959128234	21.80394559437523
2	5.978099507261897	0.43917591301986836	0.19975597165865489	0.3322324485129482
3	3.3266211976386066	0.43968437998545495	0.22908899834483426	0.34384244571186784
4	0.44362448255486087	0.16131430717681522	0.0716965267773628	0.0409590271833898
5	3.6906786302007193	0.22409899303913852	0.15452183926058524	0.19380190945976922
6	0.6364989318018064	0.30063735105071315	0.19000016714742216	0.4305822312238047
7	1.0113252529528491	0.1248010182835115	0.31764791232207945	0.16399426671729794
8	12.91790747286655	0.39306137909558925	0.2545453504638405	3.0667134503056466
9	24.523904447286917	0.20984015954154367	0.1959707220601711	0.3147787170346579
10	0.8769909022037814	0.2017843346096077	0.13045239164486974	0.34625080943677233

表 5.5: $a = 30$ の結果

	agent1	agent2	agent3	agent4
1	3.5183885672733712	2.5566631020368185	0.8701103013185574	0.7115265146999649
2	0.7854451329185882	0.4639505237864656	0.2157542135810349	1.2852656362528005
3	0.9326182815907638	0.2224995921448364	0.13931924708327448	1.1748114641594536
4	0.32791373196140655	0.31325042772373524	0.23039411461399717	0.6754845043522516
5	2.139058425388146	0.36719574704515345	0.22054976911314186	0.2491048234966452
6	107.4808838457019	1.1092043908700029	0.6845921865192672	0.39980289779371886
7	2.7367528672346686	2.572032260843943	2.222214031821687	0.33212399653065716
8	0.25964024455877116	0.21254102880150766	0.24091453922741646	0.30523755710313694
9	3.1734101067308025	0.3670226967011134	0.6705344840636431	0.4868484745846299
10	2.406449377327572	0.14779387115326062	0.1488893208896456	0.11880649168699225

表 5.6: パターン分類 (10 回試行)

a=	パターン A	パターン B	パターン C	パターン D
1000	10	8	6	4
500	10	8	5	2
100	9	8	5	3
50	9	6	4	1
30	6	4	4	3

以上のデータからわかることは $a = 1000$ や $a = 500$ の状態の時にはもっとも情報源の信頼度の高いエージェントが agent1 であることを 10 回の試行すべてに対して正しく判定しており, しかもその大半の場合において agent1 だけがほかのエージェントに比べて異常に信頼度が高くなっていることである. しかし $a = 100$ に下がって更新作業の制限時間を短くするにつれて agent1 がもっとも信頼できるエージェントであるとの判定が困難になり, $a = 30$ まで下げると確実さはほとんどなくなっている. ちなみに表にはしていないが $a = 10$ まで下げると情報源の信頼度の上下動がランダムなものと化して最終的な測定値が無限に達してしまうなどで測定が不能となってしまう.

表 5.6 は以上の 10 回の試行に対して示した結果をいくつかのパターンに分類してその回数を整理したものである. それぞれのパターンは以下の通りである.

パターン A　agent1 がもっとも高い信頼度を獲得した場合.
パターン B　agent1 がほかのエージェントに比べて一人勝ちをした場合. ここで一人勝ちとは二番目に信頼度の高いエージェントと比べても一つ以上けたの多い数値を獲得した時を指す.
パターン C　agent1 がもっとも高い信頼度を獲得し, agent2 が次に高い信頼度を獲得した場合.
パターン D　agent1,2,3,4 の順に正しい情報を提供する確率の高い順番通に並んだ場合.

この表を見てみると実際に a の値が下がって計算時間の制限が短くなるにつれて agent1 がもっとも信頼できるエージェントであると正しく判断できる傾向は弱くなっていくことがわかる. また, このシステムでは agent1 がもっとも信頼できる場合はほとんどが一人勝

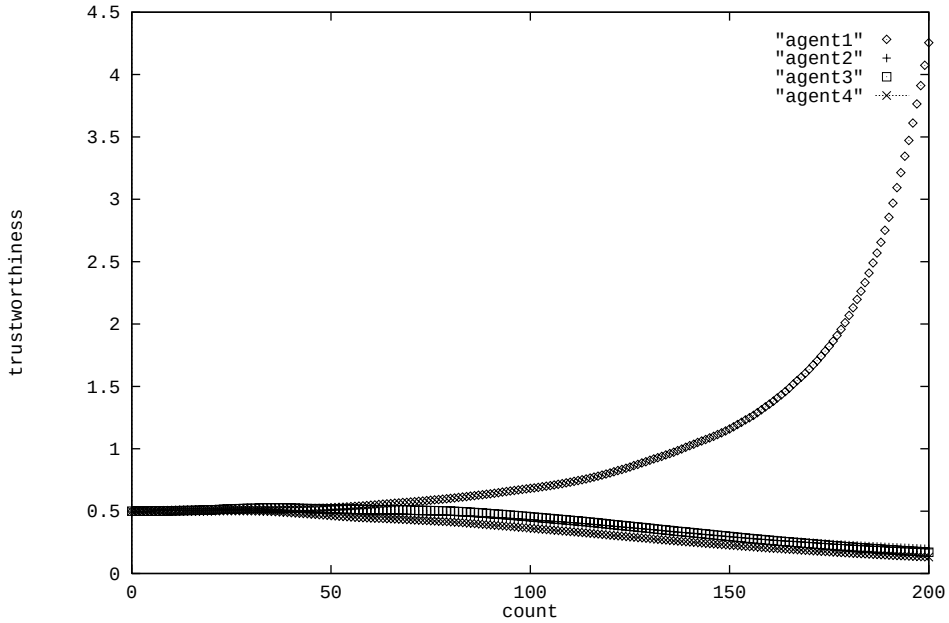


図 5.1: a=500, 試行 1 回目の過程

ちの状態にはなるが, agent1 が一番正しいことは判定できてもそれ以外のエージェントがどういう順番で正しい情報を提供する確率が高いかということの判定は a の変動にはほとんど関係せず, きちんと判定できる可能性は低いことがわかる.

上記のような実行結果ではなく実行過程の方をグラフにしてみるとおおむね agent1 が一人勝ちを納める場合のグラフは図 5.1 のようになり agent1 の増加が極端になるのがわかる. 一人勝ちを納めない場合の図 5.2 では途中まではランダムな状態に見えるが最終的には残り 3 つがゼロに収束していって agent1 だけが増加していくようになる. だが興味深いのは $a = 50$ のように制限時間が短いときに途中までは一人勝ちになりそうないきおいであるのに, 途中で減少傾向を見せるグラフの図 5.3 が存在することである. こうなる理由は制限時間の短さが agent1 がもっとも正しいという判断を掻き乱してしまうためだと推測される. 最後に示したグラフの図 5.4 は agent1 が一番正しいと判定されなかった試行のグラフであり, あまりたいした特性は示さないランダムなものとなる.

5.3 考察

上記の実験よりこのシステムはいくつかの情報源が存在した場合もっとも高い確率で正

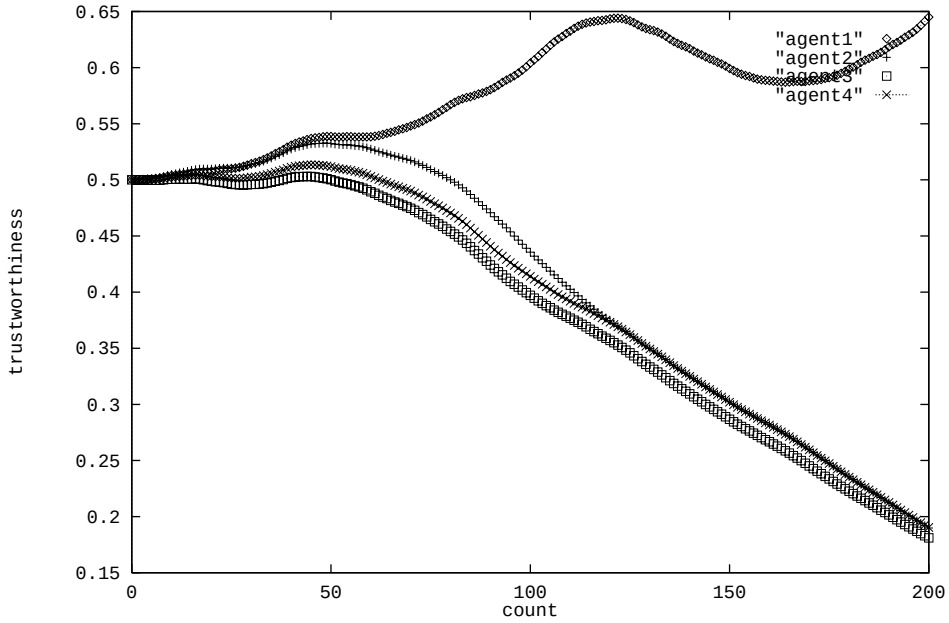


図 5.2: a=1000, 試行 6 回目の過程

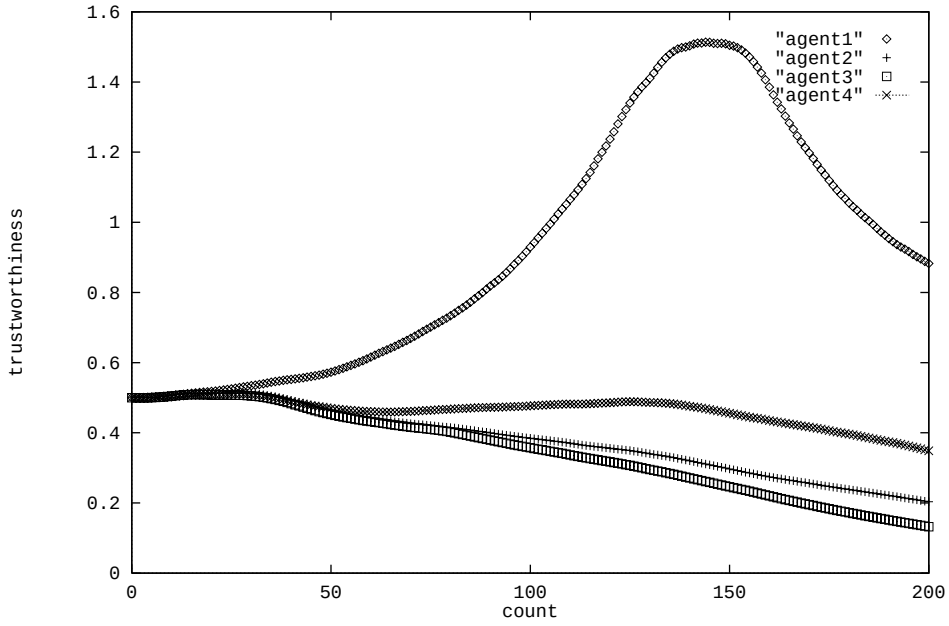


図 5.3: a=50, 試行 10 回目の過程

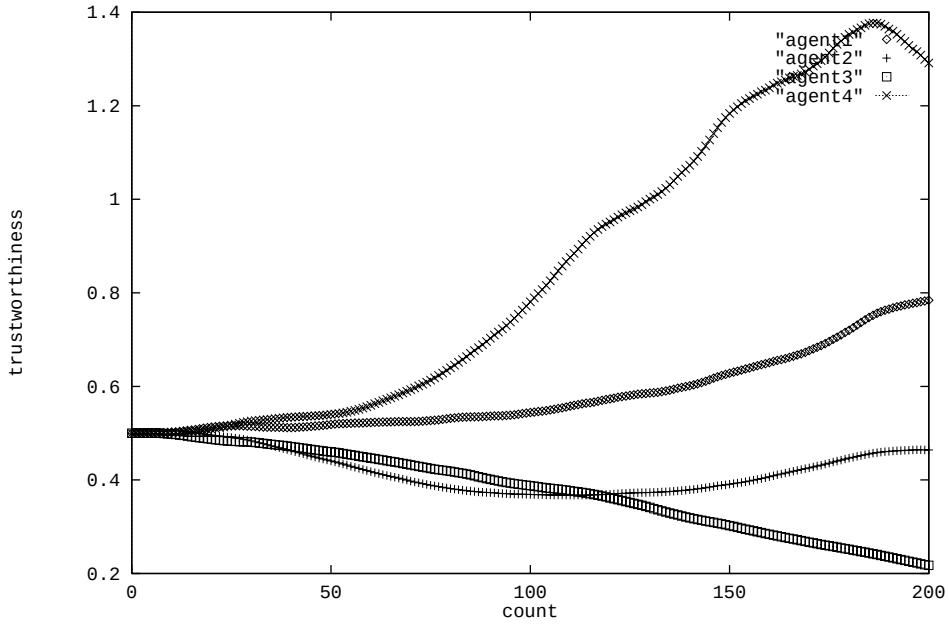


図 5.4: a=30, 試行 2 回目の過程

しい情報を提供する情報源に対してのみ高い信頼度をおいてそれ以外の情報源がどういう順番で正しい情報源を提供する確率が高いのかという考慮はほとんどしないようなシステムであることがわかった. したがって Cantwell が提唱した意味論的なモデルの代わりに情報源の信頼度に Nebel の epistemic relevance を対応させることでシンタックスの立場で知識の優先順位の根拠を与えるやり方は完全にとはいえないにしてももっとも正しい情報源がどれであるのかを判定する限りにおいて正当な考え方であることが示せた.

また正確な思考をすることよりも実行時間を節約するために途中で実行を中止する制限時間をもうけるというやり方は一番正しい情報源が誰であるかを判定できるというこのシステムの特性をある程度損なってしまい, あまりにその時間が短すぎる場合は時間の節約を急ぐあまりかえって全くでたらめな判断をしてしまうようなシステムになってしまうことが示せた.

以上のようなことをふまえてこのシステムは元々認識状態のモデルについての考察からの産物だったので, このシステムの認知的な側面を考察すると, このシステムは情報源が複数存在して誰が正しいのか判断に迷う場合, 一番正しいことをいっているように判断できる情報源だけを重宝してあとの情報源は無視を決め込むことで, 情報の判断に関して少

しでも疑わしい存在の提供した情報には危険をおかしてまで関わらないという安全策をとる認識の在り方をしている. また解決を急ぐあまり判断に時間を取らないと本当は間違っている割合が大きい存在を信じてしまい失敗するような構造になっていて, これは人間の認識の在り方と良く似ている.

第 6 章

おわりに

本論文ではまず belief revision の基本的な枠組みとなっている AGM の方法を導入し, belief revision がそのままの理論では人工知能において関心事となっているデータベースの更新の問題に使えるような実装にはならないことを論じた. そしてその解消のために Nebel の epistemic relevance を導入した方法と導出原理を用いて一応暫定的な形での実装に成功したが, 実験で示した通り計算時間の増大の問題は以前として残る結果となった.

そして Cantwell の情報源の信頼度が知識の優先順位を決定するという考えを採用し, Cantwell の意味論的なやり方でなく Nebel の epistemic relevance にこの考えを適応することを宣言した. そして Cantwell が答えることができなかった知識の優先順序の決定方法に関して「更新の結果得られた知識に対して正しい情報を提供した情報源の信頼度は上昇し, 誤った情報を提供した情報源の信頼度は下降する. 」という回答を与え, 実際にその仕組みをモデル化した. また同時に, 計算時間の増大の問題に対して途中で計算の作業を中止する制限時間をもうける方法を提案した.

以上のモデル化に基づいた実装を用いて, そのような実装がどの程度信頼できるシステムであるのかを評価する実験を行った. その結果は情報源が複数存在した場合, もっとも高い確率で正しい情報を提供する情報源に対して非常に高い信頼度を与え, 残りの情報源の信頼度はそれに比べて低くなるものとなった. そして更新実行の制限時間を減らしていくと, だんだんどれが一番信頼がおける情報源なのかという判断に関して不正確なものになっていった.

本論文の以上のような結果から残された課題としてはまず情報源の信頼度の変更結果

が正しい情報を提供する確率の高い順番にきちんとならばなかったことがあげられる. 一番正しい情報源は正しく判定できるが, それ以外のことに関しては完全な正確さを与えることができなかったことが反省される.

また「ビルは風邪を引いている」のような例の場合, こういった知識は一年間もの時間が経つとその優先順位を著しく低下させるが, これはその情報を提供した情報源の信頼度が低下するためではなく, それが時間が経つと信頼できなくなるようなたぐいの知識であることをエージェントが背景知識として持っているためであると考えられる. したがって知識の優先順位をきめる根拠は Cantwell が述べたように情報源の信頼度のみで決定されるべきものではなく様々な要因によって成り立つものと考えられるのであり, 背景知識などの観点を取り入れた本論文とは別な立場からのアプローチも考えられる.

そして本論文で取り入れることのできなかったテーマとして Doyle(1991) であげられた「AGM の知識の経済性の観点は必ずしもそれのみが経済性として妥当というわけではない」という観点でのアプローチがある. この Doyle の考え方では更新作業後の知識の量は必ずしも極大である必要性はなく, 計算論的なコストやモラルの原理などの観点からの経済性によっては AGM の原則は破られても良いということになる. 本論文での計算時間に制限時間を置くということの正当性はこの Doyle によってある程度保証されるのだが, 具体的に計算時間に応じてどの程度の知識を残すべきかという考えをとらずに入力の方だけを捨てて残った知識はすべて保有されるという観点をとったために, この Doyle の考え方に忠実にしたがった観点からはどのような更新のシステムが構築され得るのかという考察を課題がまだ残っている.

参考文献

[1] Alchourron, C. E., Gärdenfors, P., Makinson, D., On the Logic of Theory Change: Partial Meet Contraction and Revision Functions, in Journal of Symbolic Logic, Vol. 50, No. 2, pp510–530, 1985.

[2] Nebel, B., Syntax based approaches to belief revision, in Belief Revision, Gärdenfors, P., ed., Cambridge: Cambridge University Press, pp52–58, 1992.

[3] Cantwell, J., Resolving Conflicting Information, in Journal of Logic, Language, and Information, Vol. 7, pp191–220, 1998.

[4] Doyle, J., Rational Belief Revision (Preliminary Report), in Principles of Knowledge Representation and Reasoning: Proceedings of the Second International Conference KR'91, J.Allen, R.Fikes, and E.Sandewall, ed., CA: Morgan Kaufmann, pp163–174 1991.

[5] Gärdenfors, P., Knowledge in Flux: Modeling the Dynamics of Epistemic States, Cambridge, MA: The MIT Press, 1988.

[6] Hansson, S. O., A Textbook of Belief Dynamics: Theory Change and Database Updating, Dordrecht: Kluwer Academic Publishers, 1998a.

[7] Hansson, S. O., Editorial: Belief Revision Theory Today, in Journal of Logic, Language, and Information, Vol. 7, pp123–126, 1998b.

[8] Katsuno, H., Mendelzon, A.O., On the Difference between Updating a Knowledge Base and Revising it, in Belief Revision, Gärdenfors, P., ed., Cambridge: Cambridge University Press, pp183–203, 1992.

[9] Nebel, B., Belief revision and default reasoning: Syntax-based approaches, in Principles of Knowledge Representation and Reasoning: Proceedings of the Second International Conference KR'91, J.Allen, R.Fikes, and E.Sandewall, ed., CA: Morgan Kaufmann, pp417–428 1991.

[10] Li, J., A note on partial meet package contraction, in Journal of Logic, Language, and Information, Vol. 7, pp139–143, 1998.

[11] Olsson, E. J., Making beliefs coherent. The subtraction and addition strategies, in Journal of Logic, Language, and Information, Vol. 7, pp143–163, 1998.

[12] Witteveen, C., Belief Revision in Truth Maintenance, in Logic, Action, and Information: Essays on Logic in Philosophy and Artificlal Intelligence, Fuhrmann, A., and Rott, H., ed., Berlin, New York: Walter de Gruyter, pp447–470, 1996.

[13] Friedman, N., Halpern, J. Y., Modeling belief in dynamic systems, Part I: Foundations, in Artificial Intelligence, Vol. 95, pp257–316, 1997.

[14] Meyer, T. A., Labuschagne, W. A., Heidema, J., Infobase Change: A First Approximation, in Journal of Logic, Language, and Information, Vol. 9, pp353–377, 2000.

[15] Gärdenfors, P., Makinson, D., Revisions of knowledge systems using epistemic entrenchment, in Proceedings of the Second Conference on Theoretical Aspects of Reasoning about Knowledge, Y. Vardi, ed., Los Altos: Morgan Kaufmann, pp83–95 1988.

[16] Doyle, J., A truth maintenance system, in Artificial Intelligence, Vol. 12, pp231–272, 1979.

[17] Fagin, R., Ullman, J. D., Vardi, M. Y., On the semantics of updates in datebases, in Preceedings of Second ACM SIGACT-SIGMOD, pp352–365, 1983.

[18] Katsuno, H., Mendelzon, A.O., Propositional knowledge base revision and minimal change, in Artificial Intelligence, Vol. 52, pp263–294, 1991.

[19] Hansson, S.O., A dyadic representation of belief, in Belief Revision, Gärdenfors, P., ed., Cambridge: Cambridge University Press, pp89–121, 1992.

[20] Ferme, E. L., On the Logic of Theory Change: Contraction without Recovery, in Journal of Logic, Language, and Information, Vol. 7, pp127–137, 1998.