『コンピュータサイエンスにおける様相論理』第 1 版第 1 刷の正誤表と補遺　（2025 年 7 月現在）

.....

【正誤表】

38 ページ，5 行目.

（誤） $M, (\Gamma \Rightarrow \Delta) \nvdash \xi$

（正） $M, (\Gamma \Rightarrow \Delta) \not\models \xi$

99 ページ，11 行目（式 (4.6) の直後）.

（誤） x_0 が μ

（正） $x_0 \in \mathrm{Inf}(P)$ が μ

110 ページ，下から 11 行目.

（誤） $\{s \in S \mid s$ は

（正） $\{s \in S \mid (x, s)$ は

134 ページ，図 6.1 のプログラム 4 行目.

（誤） `v2 := v3;`

（正） `v2 := v3`

135 ページ，下から 5 行目（脚注含まず）.

（誤）v2 の値だけある

（正）v2 に関する条件だけである

147 ページ，1 行目，5 行目，11 行目，16 行目.

（誤） $A\{\pi\}B$

（正） $\{A\}\pi\{B\}$

.....

【補遺】

定理 3.4.4（**CTL** の有限モデル性）についての注意

本書で示した状態数は $\mathrm{O}(2^{2^{\mathrm{Lh}(\varphi)}})$ であるが，補遺の最後に追加した参考文献 [31] 第 16 章の定理 6.14 では $\mathrm{O}(2^{\mathrm{Lh}(\varphi)})$ が示されている．

3.6 節における EU の長さについて

3.6 節では EX は $\neg$AX$\neg$ の省略形なので，$\mathrm{Sub}^+(p\mathsf{EU}q) = \{\neg\mathsf{AX}\neg(p\mathsf{EU}q),\ \mathsf{AX}\neg(p\mathsf{EU}q),\ \neg(p\mathsf{EU}q),\ p\mathsf{EU}q,\ p,\ q\}$ である（$\{\mathsf{EX}(p\mathsf{EU}q),\ p\mathsf{EU}q,\ p,\ q\}$ ではない）．そのため，$|\mathrm{Sub}^+(\xi)| \leq \mathrm{Lh}(\xi)$ を保証するためには EU の長さは 2 ではなく 4 にする必要がある（つまり $\mathrm{Lh}(\mathsf{EX}(\alpha\mathsf{EU}\beta)) = \mathrm{Lh}(\alpha) + \mathrm{Lh}(\beta) + 4$）．ただし 3.6 節の最後の段落（p.78 の一番下）で説明しているように，$\neg$ はここで作るモデルの状態数に影響しないので，有限モデル性の文面では EU の長さは 2 としてよい．

ホーア論理の相対完全性の証明

定理 6.3.9 の後半，すなわちホーア論理の相対完全性を，本文と少し異なる方針で少し詳細に証明する．

任意のプログラム π と任意の表明 B に対して次の (I)(II) が成り立つことを，π に関する帰納法で示す．

(I) ある表明 A_0 が存在して次の (1)(2) が成り立つ（この A_0 を「π と B に対する最弱前条件」と呼ぶ）．

(1) $\mathbf{Hoare} \vdash \{A_0\}\pi\{B\}$.

(2) 任意の変数解釈 J, I に対して次が成り立つ．

$J, I \not\models A_0$ ならば，ある J' に対して $J \overset{\pi}{\mapsto} J'$ かつ $J', I \not\models B$.

(II) 任意の表明 A に対して次が成り立つ．

$\models \{A\}\pi\{B\}$ ならば $\mathbf{Hoare} \vdash \{A\}\pi\{B\}$.

この (II) が求める完全性である．ところで（I $\Rightarrow$ II）が以下のように証明できる．

(I) と $\models \{A\}\pi\{B\}$ を仮定する．まず $\models A \to A_0$ であることを示す．もし $\not\models A \to A_0$ ならば，ある J, I に対して $J, I \models A$ かつ $J, I \not\models A_0$ となっている．しかしこれは，(I) の (2) および仮定 $\models \{A\}\pi\{B\}$ と矛盾する．したがって $\models A \to A_0$ である．すると (I) の (1) と帰結規則によって $\mathbf{Hoare} \vdash \{A\}\pi\{B\}$ となる．

このことから実際には (I) だけを示せばよいことになる．以下では π に関する帰納法で，任意の B に対して (I) が成り立つことを示す．

【π が $\mathtt{v}i := t$ という代入文のとき】 $B(t/\mathtt{v}i)$ が求める A_0 になっている．

【π が複合文のとき】 簡単のために $\pi = \mathtt{begin}\ \pi_1; \pi_2\ \mathtt{end}$ とする（3 個以上でも同様にできる）．このとき帰納法の仮定により π_2 と B に対する最弱前条件 X が存在し，さらに π_1 と X に対する最弱前条件 Y が存在する．すると Y が $\mathtt{begin}\ \pi_1; \pi_2\ \mathtt{end}$ と B に対する最弱前条件になっていることは簡単に確認できる．

【π が $\mathtt{if}\ C\ \mathtt{then}\ \pi_1\ \mathtt{else}\ \pi_2$ のとき】帰納法の仮定により π_1 と B に対する最弱前条件 A_1, およびと π_2 と B に対する最弱前条件 A_2 が存在する．すなわち次の 4 つが成り立つ．(1-1) $\mathbf{Hoare} \vdash \{A_1\}\pi_1\{B\}$. (1-2) $(^{\forall}J, I)(J, I \not\models A_1 \Rightarrow (^{\exists}J')(J \overset{\pi_1}{\mapsto} J'$ かつ $J', I \not\models B))$. (2-1) $\mathbf{Hoare} \vdash \{A_2\}\pi_2\{B\}$. (2-2) $(^{\forall}J, I)(J, I \not\models A_2 \Rightarrow (^{\exists}J')(J \overset{\pi_2}{\mapsto} J'$ かつ $J', I \not\models B))$. このとき $A_0 = (C \to A_1) \wedge (\neg C \to A_2)$ とすれば，これが $\mathtt{if}\ C\ \mathtt{then}\ \pi_1\ \mathtt{else}\ \pi_2$ と B に対する最弱前条件になっている．A_0 が最弱前条件の条件 (1) を満たすことは，(1-1), (2-1) と $\models (C \wedge A_0) \to A_1$ および $\models (\neg C \wedge A_0) \to A_2$ という事実と，帰結規則と if 文規則で言える．A_0 が最弱前条件の条件 (2) を満たすことは，$J, I \not\models A_0$ ならば $J, I \models C \wedge \neg A_1$ または $J, I \models \neg C \wedge \neg A_2$ であることと，(1-2), (2-2) および if 文の動作の定義から言える．

【π が $\mathtt{while}\ C\ \mathtt{do}\ \pi'$ のとき】 簡単のためこのプログラム中には変数がただひとつ（$\mathtt{v}$ と書く）しか使われていないとする（変数が複数あっても同様にできる）．このとき次が求める A_0 である．

$$(\forall k \geq 0)\forall x_0 \forall x_1 \cdots \forall x_k$$
$$\left[\left(\begin{array}{l}\mathtt{v} = x_0 \wedge \\ (\forall i)\Big((0 \leq i < k) \to C(x_i/\mathtt{v}) \wedge \text{“}(\mathtt{v} = x_i) \overset{\pi'}{\mapsto} (\mathtt{v} = x_{i+1})\text{”}\Big)\end{array}\right) \to C(x_k/\mathtt{v}) \vee B(x_k/\mathtt{v})\right]$$

ただしここでは以下の事実を使用している．

- 可変個数の表明変数を用いた記法 $\forall x_0 \forall x_1 \cdots \forall x_k$ は表明言語（算術の一階述語言語）を逸脱しているが，任意の長さの整数列をひとつの自然数で表現する技法（ゲーデル数化とも言う）を用いれ

ば，固定した表明変数で同じ意味を表現できる．

- “$(\mathtt{v} = x_i) \overset{\pi'}{\mapsto} (\mathtt{v} = x_{i+1})$” は「変数 v の値が x_i のときにプログラム π' を実行開始すると，v の値が x_{i+1} になって停止する」を表す表明であり，算術の一階述語言語で記述することができる．

これらの詳細は参考文献 [29,2,3] やゲーデルの不完全性定理を解説している教科書を見てほしい．A_0 の直観的な意味は次の通りである：「変数 v の現在の値を x_0 としてその後 $x_1, \ldots, x_k$ とどのように変化しても，それがプログラム $\mathtt{while}\ C\ \mathtt{do}\ \pi'$ の実行（の途中までまたは最後まで）に従った変化であるならば，$\mathtt{v} = x_k$ のときに条件 C または B が成り立つ」．以下では A_0 が $\mathtt{while}\ C\ \mathtt{do}\ \pi'$ と B に対する最弱前条件の条件 (1) と (2) を満たすことを示す．(1)：まず A_0 の内容をよく考えれば $\models \{C \wedge A_0\}\pi'\{A_0\}$ であることが言える．したがって帰納法の仮定 (II) により $\mathbf{Hoare} \vdash \{C \wedge A_0\}\pi'\{A_0\}$ となり，while 文規則で $\mathbf{Hoare} \vdash \{A_0\}\ \mathtt{while}\ C\ \mathtt{do}\ \pi'\ \{\neg C \wedge A_0\}$ を得る．一方 $\models \neg C \wedge A_0 \to B$ も言える（A_0 で $k = 0, x_0 = \mathtt{v}$ とすればよい）．したがって帰結規則により $\mathbf{Hoare} \vdash \{A_0\}\ \mathtt{while}\ C\ \mathtt{do}\ \pi'\ \{B\}$ が得られる．(2)：$J, I \not\models A_0$ であるとする．これはすなわち「プログラム $\mathtt{while}\ C\ \mathtt{do}\ \pi'$ の実行に従った変数 v の値の遷移 $x_0, x_1, \ldots, x_k$ が存在して，$\mathtt{v} = x_k$ のときに C も B も成り立たない」ということである．すなわち $J \overset{\pi}{\mapsto} J'$ かつ $J', I \not\models B$ となる J' が存在する．

参考文献の追加

[31] 広瀬健，野崎昭弘，小林孝次郎（監訳）．**形式的モデルと意味論（コンピュータ基礎理論ハンドブック II）**．丸善，1994．（J. van Leeuwen (ed.), *Formal Models and Semantics (Handbook of Theoretical Computer Science, Volume B)*, Elsevier 1990 の翻訳）．

[32] 徳山豪，小林直樹（編集），**理論計算機科学事典**．朝倉書店，2022．

文献 [31] では **CTL** とその周辺の論理が第 16 章で，**PDL** が第 14 章で，ホーア論理が第 15 章で，それぞれ詳しく説明されている．

文献 [32] の第 8 章ではさまざまな種類のモデル検査が説明されている．ホーア論理も説明されている．