情報論理の壺４

萩谷

算術の体系とその符号化

4.3

- 一階の算術（自然数に関する一階述語論理）
  - 定数記号 0 および関数記号 S( ) ＋ ×
  - 述語記号 = <
  - 以上に関する公理および数学的帰納法の公理
  - 例えば、公理を追加したHilbert流の
    一階述語論理の体系として形式化可能。
- 符号化
  - 論理式を符号化。
  - 証明を符号化。
  - 原始帰納的関数 proof(x,y)
    proof(x,y) = 0 if y は x の表す論理式の証明の符号
    proof(x,y) = 1 otherwise
  - 定理の符号の全体は帰納的に可算。

4.3(a)

算術の公理の例

- $\forall x \forall y (S(x)=S(y) \supset x=y)$
- $\forall x (\neg 0=S(x))$

- $\forall x (x+0=x)$
- $\forall x \forall y (x+S(y)=S(x+y))$
- $\forall x (x \times 0=0)$
- $\forall x \forall y (x \times S(y)=x \times y+x)$

算術の公理の例

4.3(a)

- $\forall x(0<S(x))$
- $\forall x(\neg x<0)$
- $\forall x\forall y(S(x)<S(y)\Leftrightarrow x<y)$

- $\forall x(x=x)$
- $\forall x\forall y(x=y\land A[x]\supset A[y])$

- $A[0]\land\forall x(A[x]\supset A[S(x)])\supset\forall xA[x]$

論理式の符号とは？

論理式の符号の例

4.3(c)

$\ulcorner t_1 = t_2 \urcorner = p(1, p(\ulcorner t_1 \urcorner, \ulcorner t_2 \urcorner))$

$\ulcorner t_1 < t_2 \urcorner = p(2, p(\ulcorner t_1 \urcorner, \ulcorner t_2 \urcorner))$

$\ulcorner \neg A \urcorner = p(3, \ulcorner A \urcorner)$

$\ulcorner A_1 \lor A_2 \urcorner = p(4, p(\ulcorner A_1 \urcorner, \ulcorner A_2 \urcorner))$

$\ulcorner \forall x\, A \urcorner = p(5\ , p(\ulcorner x \urcorner, \ulcorner A \urcorner))$

$\ulcorner x \urcorner = p(1,$ 変数xの文字コード$)$

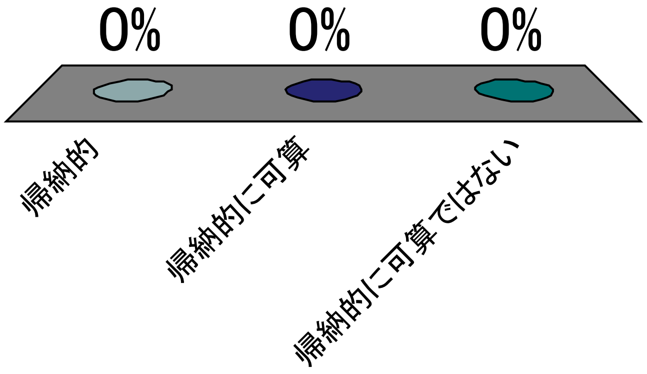
$\ulcorner 0 \urcorner = p(2, 0)$

$\ulcorner S(t) \urcorner = p(3, \ulcorner t \urcorner)$

…

公理の符号の全体は

1. 帰納的
2. 帰納的に可算
3. 帰納的に可算ではない



証明の符号とは？

4.3(c)

証明の符号

- 証明

  $A_1, A_2, \ldots, A_n$

  – 各 A_i は公理であるか、

  $A_j (j<i)$ に推論規則を適用したもの

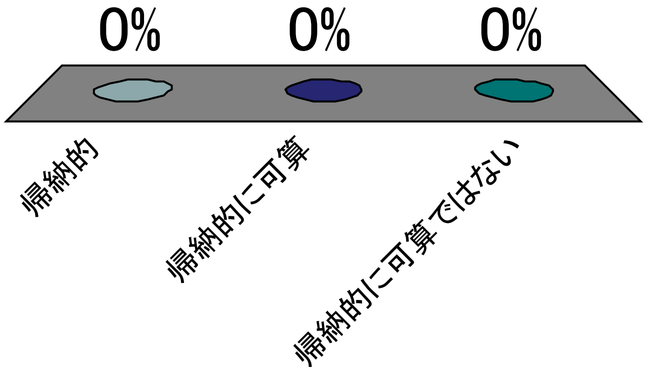
  – A_n の証明という。

- 証明の符号

  $p(\ulcorner A_1 \urcorner, p(\ulcorner A_2 \urcorner, \ldots p(\ulcorner A_n \urcorner, 0)\ldots))$

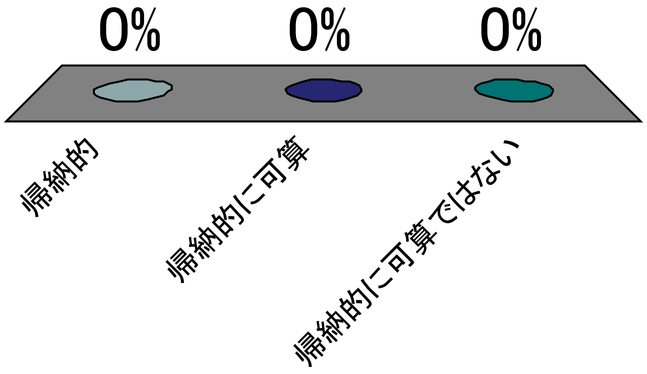
証明の符号の全体は

1. 帰納的
2. 帰納的に可算
3. 帰納的に可算ではない



定理の符号の全体は

1. 帰納的
2. 帰納的に可算
3. 帰納的に可算ではない



なぜ、定理の符号の全体は帰納的に可算か？

算術の体系とその符号化

4.3

- 一階の算術（自然数に関する一階述語論理）
  - 定数記号 0 および関数記号 S( ) ＋ ×
  - 述語記号 = <
  - 以上に関する公理および数学的帰納法の公理
  - 例えば、公理を追加したHilbert流の
    一階述語論理の体系として形式化可能。
- 符号化
  - 論理式を符号化。
  - 証明を符号化。
  - 原始帰納的関数 proof(x,y)
    proof(x,y) = 0 if y は x の表す論理式の証明の符号
    proof(x,y) = 1 otherwise
  - 定理の符号の全体は帰納的に可算。

では、定理の符号の全体は帰納的か？

表現可能性

4.3(b)

- 関数が演繹体系の「中」で計算できること。
- 関数 $f(x_1,\ldots,x_n)$ が表現可能であるとは、
  ある論理式 $A[x_1,\ldots,x_n,y]$ が存在して、
  - $f(x_1,\ldots,x_n)=y \Rightarrow A[num(x_1),\ldots,num(x_n),num(y)]$ が
    定理として証明できる。

    $\overline{x_i}$

    - $num(x_i)$ は $S(\ldots S(0)..)$ という項を表す。S は x_i 回。
  - $x_1,\ldots,x_n$ に対して y の値がユニークに決まることを
    定理として証明できる。
- 一階の算術において、
  任意の原始帰納的関数は表現可能（Gödel）。

表現可能性の別の定義

4.3(b)

- $f(x_1,\ldots,x_n)=y$ ならば

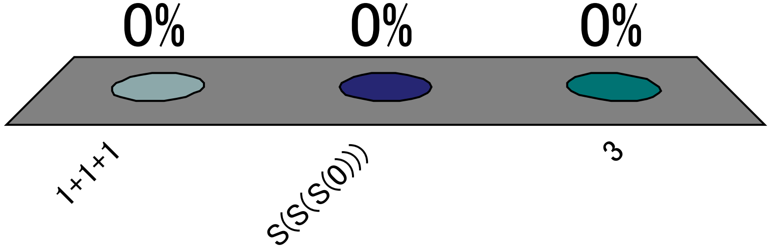
  $$\forall \mathbf{y}.\ A[\mathrm{num}(x_1),\ldots,\mathrm{num}(x_n), \mathbf{y}]$$

  $$\Leftrightarrow \mathbf{y}=\mathrm{num}(y)$$

  が証明可能
- たとえば、$x_1+ x_2=y$ という論理式は、足し算という関数を表現している。
  - 体系に関数記号 + が含まれるので当たり前。

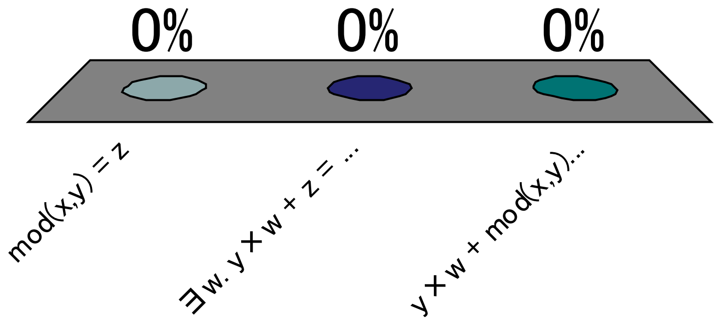
num(3) とは

1. 1+1+1
2. S(S(S(0)))
3. 3



mod(x,y) を表現する論理式 Mod[x,y,z] は

1. mod(x,y) = z
2. ∃w. y×w + z = x ∧ z<y
3. y×w + mod(x,y) = x ∧ mod(x,y)<y



不完全性

4.3(c)

ここでは健全性は
当然のこととして
仮定している。

- 算術の標準的な解釈を考える。
  - 閉じた論理式は真か偽かのどちらか。
- ここでいう完全性とは、真の論理式がすべて定理であること。
- 完全性（と健全性）が成り立つと、閉じた論理式 A に対して、

  A が真 ⇔ A が定理

  A が偽 ⇔ A が定理でない ⇔ ¬A が定理

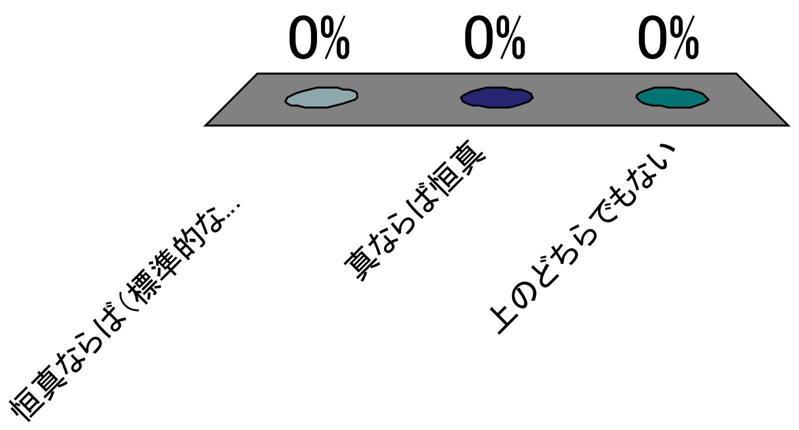
  従って、A が定理かどうかは判定可能になってしまう。
  - 定理の符号の全体は帰納的になってしまう。
- しかし、定理の符号の全体は帰納的でない。
  - T[e,x,y,z] を Kleeneの述語を表現する論理式とする。
  - K = { x | ∃y. T[num(x),num(x),y,0] は定理 }
  - 定理の符号の全体が帰納的とすると K も帰納的。
- 従って、真であるにも関わらず
  定理でない論理式が存在する。

K = { x | ∃y. T[num(x),num(x),y,0] は定理 }

- K = { x | T(x,x,y)=0 を満たす y が存在する } （Kの定義）
- x∈K とすると、
  - T(x,x,y)=0 を満たす y が存在。
  - T の表現可能性から、
    ∀z. T[num(x),num(x),num(y),z] ⇔ z=0 は定理。
  - T[num(x),num(x),num(y),0] は定理。
  - ∃y. T[num(x),num(x),y,0] は定理。
- ∃y. T[num(x),num(x),y,0] は定理であるとすると、
  - 算術の健全性から、∃y. T[num(x),num(x),y,0] は真。
  - T[num(x),num(x),num(y),0] が真となる y が存在。
  - ここで T(x,x,y)=1 と仮定すると、表現可能性から、
    ∀z. T[num(x),num(x),num(y),z] ⇔ z=S(0) は定理。
  - これも真のはずだが、T[num(x),num(x),num(y),0] と矛盾。
  - したがって、T(x,x,y)=0
  - したがって、x∈K

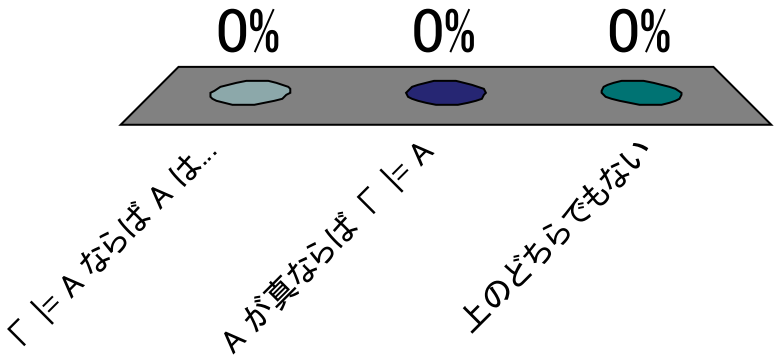
恒真と真の関係は？

1. 恒真ならば(標準的な解釈で)真
2. 真ならば恒真
3. 上のどちらでもない



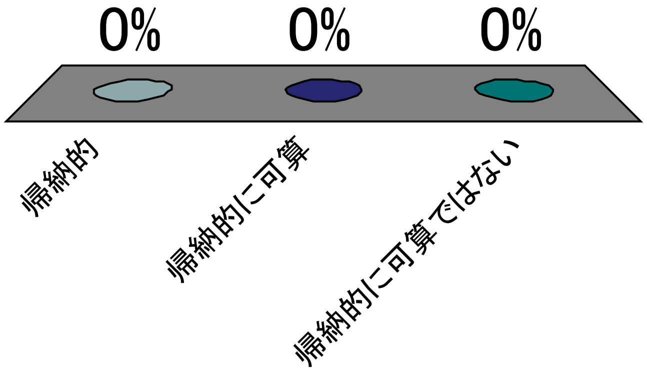
Γを算術の公理の全体として

1. Γ |= A ならば A は(標準的な解釈で)真
2. A が真ならば Γ |= A
3. 上のどちらでもない



真な論理式の符号の全体は

1. 帰納的
2. 帰納的に可算
3. 帰納的に可算ではない



不完全性より

- Γ を算術の公理の全体とすると、
  A が算術の定理 ⇔ Γ |- A ⇔ Γ |= A
  （強い完全性）
- 標準的な解釈は Γ を充足する。
- 標準的な解釈で真だが証明できない論理式 A が存在するとは、
  A は真だが Γ |= A でない
- Γ を充足するが A を偽にする解釈が存在。

4.3(d)

Gödelの不完全性

- Gödelは証明可能性の概念のみを用いて、
  不完全性を定式化した。
- 証明できない論理式を具体的に構成した。
- 閉じた論理式 G で、
  G も $\neg$G も証明できないものが存在する。
  - G をGödel文という。
  - Proof[x,y,z] が proof(x,y) を表現しているとき、
    G は $\neg\exists$y. Proof[g,y,0] と同値。g は G の符号。
  - 算術の体系がω無矛盾ならば、
    G も $\neg$G も証明できない。

正確には num(g)

4.3(d)

G の構成

- subst**x**(y,z)
  - A[**x**] の符号 y と自然数 z に対して
    A[num(z)] の符号を対応させる原始帰納的関数
  - Subst**x**[**y**,**z**,**w**] は subst**x**(y,z) を表現する論理式
- 以下では特に A[**y**] を次のように定義。
  - ∃**w.** Subst**x**[**y**,**y**,**w**]∧¬(∃p. Proof[**w**,p,0])
- A[**x**] の符号を z とする。
- A[num(z)] を G とおく。

4.3(d)

すると…

$G \Leftrightarrow A[\mathrm{num}(z)]$

$\Leftrightarrow \exists \mathbf{w}.\ \mathrm{Subst}\mathbf{x}[\mathrm{num}(z),\mathrm{num}(z),\mathbf{w}] \land$

$\neg(\exists p.\ \mathrm{Proof}[\mathbf{w},p,0])$

$\Leftrightarrow \exists \mathbf{w}.\ \mathbf{w}=\mathrm{num}(g) \land \neg(\exists p.\ \mathrm{Proof}[\mathbf{w},p,0])$

$\Leftrightarrow \neg(\exists p.\ \mathrm{Proof}[\mathrm{num}(g),p,0])$

- g は G の符号
- これらの同値性を
  算術の定理として証明できる。

表現可能性の別の定義

4.3(b)

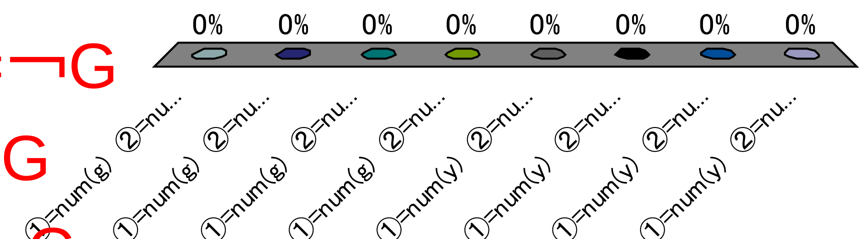
- $f(x_1,\ldots,x_n)=y$ ならば

  $$\forall \mathbf{y}.\ A[\mathrm{num}(x_1),\ldots,\mathrm{num}(x_n), \mathbf{y}] \Leftrightarrow \mathbf{y}=\mathrm{num}(y)$$

  が証明可能
- たとえば、$x_1+ x_2=y$ という論理式は、足し算という関数を表現している。
  - 体系に関数記号 + が含まれるので当たり前。

G が証明できるとすると、
G の証明の符号を y として、
Proof[①,②,0] が証明できる。
すると、∃p. Proof[①,p,0] は証明できる。
同値性より、③も証明できる。矛盾！

1. ①=num(g) ②=num(g) ③=G
2. ①=num(g) ②=num(g) ③=¬G
3. ①=num(g) ②=num(y) ③=G
4. ①=num(g) ②=num(y) ③=¬G
5. ①=num(y) ②=num(g) ③=G
6. ①=num(y) ②=num(g) ③=¬G
7. ①=num(y) ②=num(y) ③=G
8. ①=num(y) ②=num(y) ③¬G



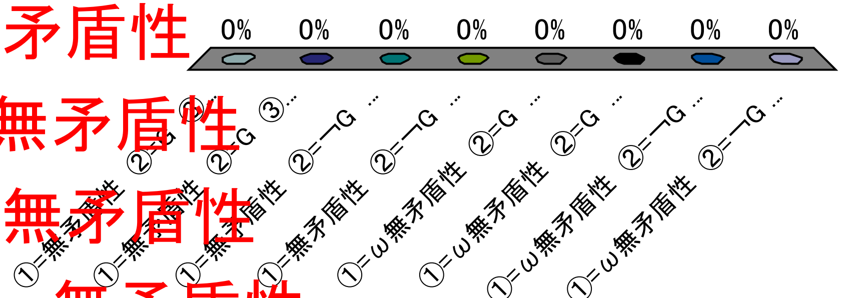
ω無矛盾

4.3(d)

- 以下のような論理式 A[**x**] が存在するとき演繹体系はω矛盾するという。
  - ∃**x**A[**x**] は証明できる。
  - 任意の y に対して ¬A[num(y)] が証明できる。
- このような論理式が存在しないとき演繹体系はω無矛盾であるという。

¬G が証明できるとすると、同値性より、
∃p. Proof[num(g),p,0] が証明できる。
①より②は証明できないはずなので、
任意の y に対して、proof(g,y)=1 であるから、
¬Proof[num(g),num(y),0] が証明できる。
これは、③に反する。

1. ①=無矛盾性 ②=G ③=無矛盾性
2. ①=無矛盾性 ②=G ③=ω無矛盾性
3. ①=無矛盾性 ②=¬G ③=無矛盾性
4. ①=無矛盾性 ②=¬G ③=ω無矛盾性
5. ①=ω無矛盾性 ②=G ③=無矛盾性
6. ①=ω無矛盾性 ②=G ③=ω無矛盾性
7. ①=ω無矛盾性 ②=¬G ③=無矛盾性
8. ①=ω無矛盾性 ②=¬G ③=ω無矛盾性



Gödelの不完全性

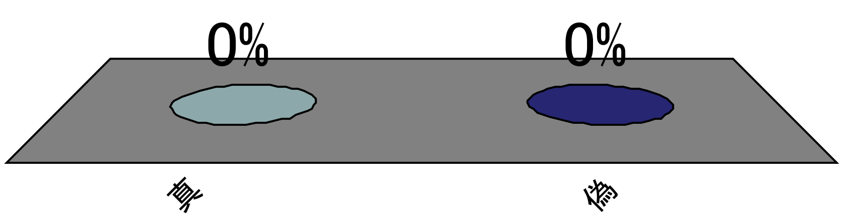
4.3(d)

- Gödelは証明可能性の概念のみを用いて、不完全性を定式化した。
- 証明できない論理式を具体的に構成した。
- 閉じた論理式 G で、G も ¬G も証明できないものが存在する。
  - G をGödel文という。
  - Proof[x,y,z] が proof(x,y) を表現しているとき、G は ¬∃y. Proof[g,y,0] と同値。g は G の符号。
  - 算術の体系がω無矛盾ならば、G も ¬G も証明できない。

正確には num(g)

G は

1. 真
2. 偽



第二不完全性

4.3(e)

- G は ¬(∃p. Proof[num(g),p,0]) と同値。
- 演繹体系が無矛盾

  ⇔ 証明できない論理式が存在する

  ⇔ 特定の論理式が証明できない
- G は算術が無矛盾であることを意味する。
- G が証明できない

  ⇔ 算術の無矛盾性が算術の中で証明できない

Presburger算術

4.4(a)

- 構文論
  - 定数記号 0 と 1
  - アリティ2の関数記号 + と -
    - 定数倍は表現可能（5x := x+x+x+x+x）
  - アリティ2の述語記号 <
    - 等号は定義可能（$x=y := x<y+1 \wedge y<x+1$）
    - 定数で割り切れるは表現可能（$5|y := \exists x.\ y=5x$）
- 意味論
  - 領域は整数の全体とする。
    - 不等号を用いて自然数に限定することは可能。
    - 関数記号と述語記号は標準的に解釈。

限定子除去

4.4(a)

例：

$\exists x.\ A[x]$ where $A[x] := x<y+y \land z < x \land 3|x$

$A_{-\infty}[x] :=$ 真 $\land$ 偽 $\land\ 3|x$

$\exists x.\ A[x] \Leftrightarrow A_{-\infty}[1] \lor A_{-\infty}[2] \lor A_{-\infty}[3] \lor$

$A[z+1] \lor A[z+2] \lor A[z+3]$

- A[x] を満たすいくらで小さい x が存在する場合
  - x < t という条件は真、t < x という条件は偽。
- A[x] を満たす最小の x が存在する場合
  - t < x といういずれかの条件をぎりぎりで満たしているはず。

4.4(a)

$$
\begin{aligned}
\exists x.\ A[x] \Leftrightarrow\ & A_{-\infty}[1] \lor A_{-\infty}[2] \lor A_{-\infty}[3] \lor \\
& A[z+1] \lor A[z+2] \lor A[z+3] \\
\Leftrightarrow\ & A[z+1] \lor A[z+2] \lor A[z+3] \\
\Leftrightarrow\ & (z+1<y+y \land z<z+1 \land 3 \mid z+1) \lor \\
& (z+2<y+y \land z<z+2 \land 3 \mid z+2) \lor \\
& (z+3<y+y \land z<z+3 \land 3 \mid z+3)
\end{aligned}
$$

限定子除去

4.4(a)

- 最も内側の限定子(量化子)に着目。
- ∃x. A[x] とする。
  - ∀ならば否定を考える。
  - A[x] は限定子を含まない。
- 否定を内側に押し込んで削除する。
- 原子論理式は $x < t$ か $t < x$ か $d|x+t$
  - t には x は現れない。
  - dx は x で置き換えて $d|x$ という条件を追加。

限定子除去

4.4(a)

- $d|x+t$ という条件の d の最小公倍数を δ
- $t < x$ という条件の t の全体を D
- $A_{-\infty}[x]$ は $x < t$ という条件を真、
  $t < x$ という条件を偽に変えたもの。
- $\exists x.\ A[x]$ は以下に同値

$$\bigvee_{j=1,\ldots,\delta} A_{-\infty}[j] \vee \bigvee_{j=1,\ldots,\delta} \bigvee_{t \in D} A_{-\infty}[t+j]$$

- 与えられた論理式が「真」かどうかは、
  判定可能。

述語論理の決定不能性と決定可能な部分体系

- 一般の一階述語論理の論理式が与えられたとき、恒真かどうかは決定不能。
- 関数記号がなく（定数記号はあってもよい）、述語記号が単項（アリティが１）に限るとき、論理式が恒真かどうかは決定可能。
  – 等号（アリティが2）が入っていてもよい。
- その他、論理式が恒真かどうかが決定可能。
  – 一階述語論理のガード付きフラグメント
  – 後継者のみの単項二階論理

一階述語論理

- 「完全」な演繹体系が存在する。
  - 恒真ならば必ず証明できる。
  - 定理（の符号）の全体は帰納的に可算なので、恒真な論理式（の符号）の全体は帰納的に可算。
  - しかし、一般には帰納的にはならない。
- 論理式を限定すると、恒真か否か判定できる。
  - 例：単項述語論理