函数記号付一階様相 μ 計算

岡本 圭史 † 木下 佳樹 †
Keishi OKAMOTO Yoshiki KINOSHITA

† 産業技術総合研究所 システム検証研究センター
keishi-okamoto@aist.go.jp yoshiki@m.aist.go.jp

先に第一著者が提示した一階様相 μ 計算 [7] は技術的な理由から函数記号を持たない．それでは実用上不便なので，本論文では，函数記号付一階様相 μ 計算を提示し，函数記号消去の技法を用いて，その体系を，同等ではあるが函数記号を持たない体系に変換する手法を示す．函数記号付一階様相 μ 計算では定数記号は状態によって解釈が変わるので，命令型プログラミング言語のプログラム変数にあたり，変数記号の付値の下での解釈値は，状態によらず一定なので，プログラムの定数にあたることになる．

1 序章

システム検証における検証項目の中には従来の (命題) 時相論理では記述できない重要なものがあり，そのような検証項目を記述するために，時相論理を一階へ拡張した論理，一階時相論理 [2, 4, 6] が考えられてきた．第一著者が提案した一階様相 μ 計算 [7] も，様相を時間の変化として解釈すれば一階時相論理になり，既存の一階時相論理は，自然な埋め込みにより，一階様相 μ 計算に埋め込まれる．さらに，一階様相 μ 計算は，既に様々な研究がなされている，一階述語論理，一階様相論理 [1]，命題様相 μ 計算 [5] の自然な拡張になっており，これらとの比較が容易である．これにより，一階述語論理などが持つ有用な性質を一階様相 μ 計算で再定式化することが容易になり，ひいては，それらが成り立つかどうかの確認が可能になる．

一階様相論理では，定数記号や函数記号の解釈を状態 (可能世界) によって変わりうるようにして，明けの明星と宵の明星が，天文学が十分進んでいない世界では別のモノを指し示すが，進んだ世界では，同じ金星を指し示す，といった状況を作りだしている．しかし，函数記号の解釈が状態によって変わるようでは，$\exists$ 除去や $\forall$ 導入の規則の健全性が得られず，さらには，函数記号消去も出来ないように見える．一階様相論理と命題様相 μ 計算を拡張した一階様相 μ 計算でも事情は同じである．そこで，第一著者は一階様相 μ 計算を函数記号を持たないものとして定式化し，健全性および帰納的に公理化不能であることを示した [4, 7]．

しかし，我々が一階様相 μ 計算を導入するのは，情報処理システムの検証に用いるためであり，そのためには，函数記号がなくては極めて不便である．そこで本論分では，函数記号付一階様相 μ 計算を提示し，一階述語論理での函数記号消去の技法を用いて，その体系を，同等ではあるが函数記号を持たない体系に変換する手法を示して，もともとの函数記号を持たない一階様相 μ 計算に関する結果が，函数記号を持つものにも適用できることを明確にした．

ここに提示する函数記号付一階様相 μ 計算では定数は状態によって解釈が変わるが，変数の付値の下での解釈値は，状態によらない．やや混乱を招くかもしれないが，命令型プログラムの言葉で述べると，定数記号がプログラム変数に，変数記号がプログラムの定数に，それぞれ対応する．

2 函数記号付一階様相 μ 計算

一階様相 μ 計算 [7] の語彙は函数記号を含んでいない．しかし，函数記号を用いた論理式による表現は馴染み深いものであり，理解しやすい．そこで本節では，語彙に函数記号を許すように一階様相 μ 計算を拡張し，この論理を函数記号付一階様相 μ 計算と呼ぶことにする．なお，一階述語論理と同様な定義 (項)，あるいは一階様相 μ 計算と同様な定義 (論理式，構造，付値) に関しては簡単に述べるに止める．

定義 1 函数記号付一階様相 μ 計算の語彙 σ は，一階様相 μ 計算の語彙 (ふたつの集合 $\mathcal{IV}$，$\mathcal{PV}$ と自然数 m 毎に用意される集合 $Pred_m$) と，自然数 n 毎に用意される集合 $Func_n$ からなる．

$\mathcal{IV}$ の要素 $(x, y, \ldots)$ を対象変数，$\mathcal{PV}$ の要素 $(X, Y, \ldots)$ を命題変数と呼ぶ．また $Pred_m$ の要素 $(P, Q, \ldots)$ を m-変数述語記号，$Func_m$ の要素

$(f, g, \ldots)$ を m-変数函数記号とそれぞれ呼ぶ．特に 0-変数函数記号を $c, d, \ldots$ で表し，定数記号と呼ぶことにする．なお，特に断らない限り，語彙 σ を固定して定義を進める．

函数記号付一階様相 μ 計算の σ-項を，一階述語論理同様，帰納的に定義する．すなわち，対象変数と定数記号は σ-項，また f が n-変数函数記号で $\tau_1, \ldots, \tau_n$ が σ-項のとき $f(\tau_1, \ldots, \tau_n)$ は σ-項とする．

函数記号付一階様相 μ 計算の σ-原始論理式を，一階述語論理と同様に定義し，σ-論理式を，一階様相 μ 計算の σ-論理式の定義を拡張して，次のように定義する．すなわち，σ-原始論理式と命題変数は σ-論理式，φ と ψ が σ-論理式ならば $\neg\varphi, \varphi \vee \psi, \forall x.\, \varphi, \Box\varphi$ は σ-論理式，φ が σ-論理式で命題変数 X の φ における出現すべてが肯定的なとき，$\mu X.\, \varphi$ は σ- 論理式とする．また $\varphi \wedge \psi$，$\exists x.\, \varphi$，$\Diamond\varphi$，$\nu X.\, \varphi$，$\varphi \equiv \psi$ なども通常通りの省略形として用いる．

次に，一階様相 μ 計算の σ-構造の解釈函数の定義を拡張して，函数記号付一階様相 μ 計算の σ-構造を定義する．すなわち，函数記号付一階様相 μ 計算の σ-構造は，集合 S と D，S 上の二項関係 R，σ の m-変数述語記号または n-変数函数記号と，状態を受け取り，それぞれ D 上の m-変数述語と n-変数函数を返す函数 I からなる四つ組み $\langle S, R, D, I\rangle$ とする．このとき，D を領域，I を解釈函数と呼び，S の要素を状態 (または可能世界) と呼ぶ．

続いて，自由変数を解釈する函数を定義する．σ-構造 $\langle S, R, D, I\rangle$ に対する函数記号付一階様相 μ 計算の付値は，函数 $v\colon \mathcal{IV} \to D$ と $V\colon \mathcal{PV} \to \mathcal{P}(S)$ の組 (v, V) とする．

一階様相 μ 計算は，対象変数だけを σ-項に持つ，特別な函数記号付一階様相 μ 計算であり，対象変数以外の σ-項の解釈を決める仕組みを持たない．そこで，σ-項を解釈する函数を新たに定義する．

定義 2 $\mathcal{A} = \langle S, R, D, I\rangle$ を構造，(v, V) を付値とする．σ-項全体と S から D への函数 d_v^I を，σ-項の構成に関する帰納法で次のように定義する．

- τ が対象変数記号 x のとき $d_v^I(x, s) \stackrel{\text{def}}{=} v(x)$
- τ が定数記号 c のとき $d_v^I(c, s) \stackrel{\text{def}}{=} I(c, s)$
- τ が $f(\tau_1, \ldots, \tau_n)$ の形の σ-項のとき
  $d_v^I(\tau, s) \stackrel{\text{def}}{=} I(f, s)(d_v^I(\tau_1, s), \ldots, d_v^I(\tau_n, s))$

σ-項 τ と状態 s に対する，函数 d_v^I の値 $d_v^I(\tau, s)$ を τ の s における v と I に関する解釈，あるいは単に τ の s における解釈と呼ぶ．

注意 3 定義から明らかなように，函数記号と述語記号の解釈は状態に依存して変化し，自由変数の解釈は状態に依存しないで定まる．

最後に，函数記号付一階様相 μ 計算における σ-論理式 φ の意味を，一階様相 μ 計算同様，φ が成り立つ状態の集合として定義する．しかし，函数記号を含む σ-論理式の意味を定めるために，一階様相 μ 計算での定義を拡張して定義する．

定義 4 $\mathcal{A} = \langle S, R, D, I\rangle$ を σ-構造，(v, V) を付値とする．函数記号付一階様相 μ 計算の σ-論理式 φ の $\mathcal{A}$ における (v, V) に関する意味 $[\![\varphi]\!]^{\mathcal{A}}_{(v,V)} \in \mathcal{P}(S)$ を，以下のように帰納的に定義する．

- $[\![P(\tau_1, \ldots, \tau_n)]\!]^{\mathcal{A}}_{(v,V)}$
  $\stackrel{\text{def}}{=} \{s \in S \mid (d_v^I(\tau_1, s), \ldots, d_v^I(\tau_n, s)) \in I(P, s)\}$
- $[\![X]\!]^{\mathcal{A}}_{(v,V)} \stackrel{\text{def}}{=} V(X)$
- $[\![\neg\varphi]\!]^{\mathcal{A}}_{(v,V)} \stackrel{\text{def}}{=} S \setminus [\![\varphi]\!]^{\mathcal{A}}_{(v,V)}$
- $[\![\varphi \vee \psi]\!]^{\mathcal{A}}_{(v,V)} \stackrel{\text{def}}{=} [\![\varphi]\!]^{\mathcal{A}}_{(v,V)} \cup [\![\psi]\!]^{\mathcal{A}}_{(v,V)}$
- $[\![\forall x.\, \varphi]\!]^{\mathcal{A}}_{(v,V)} \stackrel{\text{def}}{=} \bigcap\{[\![\varphi]\!]^{\mathcal{A}}_{(v[x\mapsto d],V)} \mid d \in D\}$
- $[\![\Box\varphi]\!]^{\mathcal{A}}_{(v,V)}$
  $\stackrel{\text{def}}{=} \{\, s \in S \mid (\forall)s'.\, s\ R\ s' \Rightarrow s' \in [\![\varphi]\!]^{\mathcal{A}}_{(v,V)} \,\}$
- $[\![\mu X.\, \varphi]\!]^{\mathcal{A}}_{(v,V)}$
  $\stackrel{\text{def}}{=} \bigcap\{\, T \subseteq S \mid [\![\varphi]\!]^{\mathcal{A}}_{(v,V[X\mapsto T])} \subseteq T \,\}$

$\mathcal{A} = \langle S, R, D, I\rangle$ を σ-構造，φ を σ-論理式とする．任意の付値 (v, V) に対し $[\![\varphi]\!]^{\mathcal{A}}_{(v,V)} = S$ となるとき，$\mathcal{A} \models \varphi$ と書く．さらに，任意の σ-構造 $\mathcal{B}$ に対し $\mathcal{B} \models \varphi$ となるとき，$\models \varphi$ と書く．

以降，特に断らない限り，σ を省略して，項，原始論理式，論理式や構造と呼び，さらに，論理式を単に式と呼ぶことにする．

3 函数記号消去

一階述語論理においては，函数記号を「消去」できることがよく知られている ([3] 等の教科書を参照)．この節では，函数記号付一階様相 μ 計算における函数記号消去定理を一階述語論理におけるのと同様の手法を用いて証明する．我々が与える証明は，同時

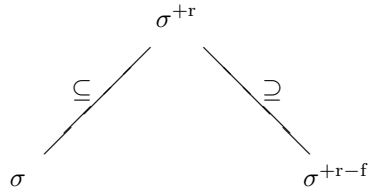

図 1: 語彙の関係

に一階様相論理の函数記号消去定理の証明にもなっている．

ある語彙 σ の各 n-変数函数記号 f に対し，新しい $n+1$-変数述語記号 P_f を σ に加えた語彙を $\sigma^{+\mathrm{r}}$ と書き，$\sigma^{+\mathrm{r}}$ から函数記号をすべて取り除いた語彙を $\sigma^{+\mathrm{r}-\mathrm{f}}$ と書く．σ-式や $\sigma^{+\mathrm{r}-\mathrm{f}}$-式はすべて $\sigma^{+\mathrm{r}}$-式でもあることに注意 (図 1 参照)．

$\sigma^{+\mathrm{r}}$-構造 $\mathcal{A}^{+\mathrm{r}}$ で，σ の全ての函数記号 f に対し

$$\mathcal{A}^{+\mathrm{r}} \models f(\bar{x}) = y \equiv P_f(\bar{x}, y)$$

が成り立つものを許される構造と呼ぶことにする．

定義 5 (等価な論理式) σ- 式 θ と $\sigma^{+\mathrm{r}-\mathrm{f}}$-式 χ が等価だとは，任意の許される構造 $\mathcal{A}^{+\mathrm{r}}$ に対し，$\mathcal{A}^{+\mathrm{r}} \models \theta \equiv \chi$ が成り立つことである．

このとき，σ-式 φ の函数記号を消去することを，φ と等価な $\sigma^{+\mathrm{r}-\mathrm{f}}$-式 $\varphi^{+\mathrm{r}-\mathrm{f}}$ を構成することと決める．

以下では，入れ子消去補題 (補題 6) が函数記号付一階様相 μ 計算において成り立つことを示し，これを用いて，函数記号消去定理 (定理 8) を証明する．

以下の原始式を入れ子でない原始式と呼ぶ．

- $x = y \quad (x, y \in \mathcal{IV})$
- $f(x_1, \ldots, x_m) = y$
  $(f \in \mathit{Func}_m, x_1, \ldots, x_m, y \in \mathcal{IV})$
- $P(x_1, \ldots, x_n)$
  $(P \in \mathit{Pred}_n, x_1, \ldots, x_n \in \mathcal{IV})$

さらに，全ての部分原始式が入れ子でない式を入れ子でない式と呼ぶ．また，σ-式 θ と χ が $\models \theta \equiv \chi$ を満たすとき，θ と χ は論理的に同値であるという．

補題 6 (入れ子消去補題) 任意の σ-式 φ に対し，φ と論理的に同値な入れ子でない σ-式 $\hat{\varphi}$ を構成できる．

証明 まずは，σ-式 φ を受け取り，入れ子でない σ-式を返す函数 Ef を，以下のように帰納的に定義する．

以下 τ_i は項を表す．また $\bar{\tau} = \langle \tau_1, \ldots, \tau_n \rangle$ と対象変数 $u_1, \ldots, u_n$ に対し，$t(\bar{\tau}) \stackrel{\mathrm{def}}{=} \{\, i \mid 1 \le i \le n, \tau_i \notin \mathcal{IV} \,\}$，$\bar{u}_{t(\bar{\tau})} \stackrel{\mathrm{def}}{=} \langle u_i \rangle_{i \in t(\bar{\tau})}$ とし，$\bar{\tau}'$ は $\bar{\tau}$ の要素 τ_i のうち，$i \in t(\bar{\tau})$ となるものを u_i に置き換えたものとする．なお P は等号以外の述語記号を表す．

1. φ が入れ子でない原始式とき $\mathrm{En}(\varphi) \stackrel{\mathrm{def}}{=} \varphi$
2. φ が $\tau_1 = \tau_2$ の形の原始式で $\tau_2 \notin \mathcal{IV}$ のとき $\mathrm{En}(\varphi) \stackrel{\mathrm{def}}{=} \exists u.\, \mathrm{En}(\tau_1 = u) \wedge \mathrm{En}(\tau_2 = u)$
3. φ が $P(\bar{\tau})$ の形の原始式で $t(\bar{\tau}) \neq \emptyset$ のとき $\mathrm{En}(\varphi) \stackrel{\mathrm{def}}{=} \exists \bar{u}_{t(\bar{\tau})}.\, \bigwedge_{t(\bar{\tau})} \mathrm{En}(\tau_i = u_i) \wedge P(\bar{\tau}')$
4. φ が $f(\bar{\tau}) = y$ の形の原始式で $t(\bar{\tau}) \neq \emptyset$ のとき $\mathrm{En}(\varphi) \stackrel{\mathrm{def}}{=} \exists \bar{u}_{t(\bar{\tau})}.\, \bigwedge_{t(\bar{\tau})} \mathrm{En}(\tau_i = u_i) \wedge f(\bar{\tau}') = y$
5. φ が $\neg\varphi_1$, $\varphi_1 \vee \varphi_2$, $\forall x.\, \varphi_1$, $\Box\varphi_1$, $\mu X.\, \varphi_1$ の形の式とき，$\mathrm{En}(\varphi)$ をそれぞれ $\neg\mathrm{En}(\varphi_1)$, $\mathrm{En}(\varphi_1) \vee \mathrm{En}(\varphi_2)$, $\forall x.\, \mathrm{En}(\varphi_1)$, $\Box\mathrm{En}(\varphi_1)$, $\mu X.\, \mathrm{En}(\varphi_1)$ とする．

ただし，2，3，4 で新たに導入される対象変数 u は，引数の式に現れていないものとする．(2 と 4 を 3 の特別な場合として扱い，規則の数を減らすことも出来るが，その場合，値となる式は冗長になる)

$\mathrm{En}(\varphi)$ の計算中に 2，3，5 が適用される回数は，φ に含まれる原始式の個数以下であるので，計算が終了することは，式の複雑さ関する帰納法により示される．さらに $\mathrm{En}(\varphi)$ は明らかに入れ子でない式である．

次に，任意の式 φ に対し，φ と $\mathrm{En}(\varphi)$ が論理的に同値であることを示す．これを示すには，1 から 5 で En の引数と値の式が論理的に同値であることを，式の複雑さ関する帰納法で示せばよい．(2 から 5 の各左辺の (En の) 引数の式は，それぞれの右辺に現れる引数の式より複雑であることに注意) なお，1 と 5 に関しては明らかに成り立ち，4 はより一般的な 3 から導かれるので，2 と 3 のみ示す．

以下 $\mathcal{A} = \langle S, R, D, I \rangle$ を構造，s を状態，(v, V) を付値，各項 τ_j は対象変数 u_i を含まないとする．

$$\begin{aligned}
& s \in [\![\tau_1 = \tau_2]\!]^{\mathcal{A}}_{(v,V)} \\
\Leftrightarrow\ & d^I_v(\tau_1, s) = d^I_v(\tau_2, s) \\
\Leftrightarrow\ & \text{ある } d \text{ に対し } d^I_v(\tau_1, s) = d, d^I_v(\tau_2, s) = d \\
\Leftrightarrow\ & \text{ある } d \text{ に対し } s \in [\![\tau_1 = u]\!]^{\mathcal{A}}_{(v[d/u],V)} \\
& \quad \text{かつ } s \in [\![\tau_2 = u]\!]^{\mathcal{A}}_{(v[d/u],V)} \\
\Leftrightarrow\ & \text{ある } d \text{ に対し } s \in [\![\mathrm{En}(\tau_1 = u)]\!]^{\mathcal{A}}_{(v[d/u],V)} \\
& \quad \text{かつ } s \in [\![\mathrm{En}(\tau_2 = u)]\!]^{\mathcal{A}}_{(v[d/y],V)} \\
\Leftrightarrow\ & s \in [\![\exists u.\, \mathrm{En}(\tau_1 = u) \wedge \mathrm{En}(\tau_2 = u)]\!]^{\mathcal{A}}_{(v,V)}
\end{aligned}$$

以下 $\langle d_v^I(\tau_1, s), \ldots, d_v^I(\tau_n, s)\rangle$ を $d_v^I(\bar{\tau}, s)$ と略記し，$t(\bar{\tau}) = \{\, i(1), \ldots, i(k) \,\}$ に対し $v[d_{i(1)}/u_{i(1)}] \ldots [d_{i(k)}/u_{i(k)}]$ を $v[d_i/u_i]_{t(\bar{\tau})}$ と略記する．

$$
\begin{aligned}
& s \in [\![P(\bar{\tau})]\!]^{\mathcal{A}}_{(v,V)} \\
\Leftrightarrow\ & \text{ある } \{\, d_i \in D \mid i \in t(\bar{\tau}) \,\} \text{ に対し} \\
& s \in [\![\bigwedge_{t(\bar{\tau})} \tau_i = u_i \wedge P(\bar{\tau}')]\!]^{\mathcal{A}}_{(v',V)} \\
\Leftrightarrow\ & \text{ある } \{\, d_i \in D \mid i \in t(\bar{\tau}) \,\} \text{ に対し} \\
& s \in [\![\bigwedge_{t(\bar{\tau})} \mathrm{En}(\tau_i = u_i) \wedge P(\bar{\tau}')]\!]^{\mathcal{A}}_{(v',V)} \\
\Leftrightarrow\ & s \in [\![\exists \bar{u}_{t(\bar{\tau})}.\ \bigwedge_{t(\bar{\tau})} (\tau_i = u_i) \wedge P(\bar{\tau}')]\!]^{\mathcal{A}}_{(v,V)}
\end{aligned}
$$

ただし $v' = v[d_i/u_i]_{t(\bar{\tau})}$ とする．

以上から，任意の式 φ に対し，φ と $\mathrm{En}(\varphi)$ は論理的に同値であるので，$\hat{\varphi} = \mathrm{En}(\varphi)$ とすればよい．□

例 7 (入れ子消去の例) P は一変数述語記号，c は定数記号とする．式 $\mathrm{En}(P(c) \supset \Box P(c))$ は，入れ子消去補題により，次の入れ子でない式となる．

$$(\exists u.\, c = u \wedge P(u)) \supset \Box \exists v.\, c = v \wedge P(v)$$

一階述語論理の場合，入れ子消去補題により変形された式は，論理的に同値な冠頭標準形にさらに変形できるが，函数記号付一階様相 μ 計算には様相記号 □ があるため，一階述語論理のような変形はできない．例えば，上の例の場合

$$\exists u, v.\, c = u \wedge P(u) \supset \Box(c = v \wedge P(v))$$

とは論理的に同値ではない．

次に，一階述語論理同様，入れ子消去補題の帰納的定義を拡張して函数記号消去定理を示す．

定理 8 (函数記号消去定理) 任意の σ-式 φ に対し，等価な $\sigma^{+\mathrm{r}-\mathrm{f}}$-式 $\varphi^{+\mathrm{r}-\mathrm{f}}$ を構成できる．

証明 まずは，入れ子消去補題同様，σ-式 φ を受け取り，$\sigma^{+\mathrm{r}-\mathrm{f}}$-式を返す函数 Ef を帰納的に定義する．($\sigma^{+\mathrm{r}-\mathrm{f}}$-式は函数記号を含まないことに注意) Ef の定義は，En の定義に次の 4' を加えたものとする．

4' φ が $f(\bar{\tau}) = y$ の形の原始式で，$t(\bar{\tau}) = \emptyset$ のとき $\mathrm{Ef}(\varphi) \stackrel{\mathrm{def}}{=} P_f(\bar{\tau}, y)$

入れ子消去補題の En 同様，任意の σ-式 φ に対し，$\mathrm{Ef}(\varphi)$ の計算は終了し，$\mathrm{Ef}(\varphi)$ は $\sigma^{+\mathrm{r}-\mathrm{f}}$-式である．

続いて，任意の式 φ に対し，φ と $\mathrm{Ef}(\varphi)$ が等価であることを示す．これには，任意の許される構造 $A^{+\mathrm{r}}$ と付値 (v, V) に対し，$[\![\varphi]\!]^{\mathcal{A}^{+\mathrm{r}}}_{(v,V)} = [\![\mathrm{Ef}(\varphi)]\!]^{\mathcal{A}^{+\mathrm{r}}}_{(v,V)}$ であることを示せばよいが，論理的に同値な式は等価でもあるので，入れ子消去補題の証明に加えて

$$[\![f(\bar{x}) = y]\!]^{\mathcal{A}^{+\mathrm{r}}}_{(v,V)} = [\![P_f(\bar{x}, y)]\!]^{\mathcal{A}^{+\mathrm{r}}}_{(v,V)}$$

を示せば十分である．しかし，これは許される構造 $\mathcal{A}^{+\mathrm{r}}$ の定義から成り立つ．よって，$\varphi^{+\mathrm{r}-\mathrm{f}} = \mathrm{Ef}(\varphi)$ とすればよい． □

例 9 (函数記号消去の例) 記号の定義は例 7 と同じとする．函数記号消去定理により，式 $\mathrm{Ef}(P(c) \supset \Box P(c))$ は，次の函数記号を含まない式となる．

$$(\exists u.\, P_c(u) \wedge P(u)) \supset \Box \exists v.\, P_c(v) \wedge P(v)$$

本節の終わりに，函数記号消去定理の応用について述べる．今 φ を σ-式，$\mathcal{A}$ を σ-構造とする．このとき $\mathcal{A}^{+\mathrm{r}}$ を $\mathcal{A}$ の $\sigma^{+\mathrm{r}}$-構造への拡張で，かつ許される構造であるものとする．(実は，任意の σ-構造に対し，このような構造はただひとつに定まる) さらに $\mathcal{A}^{+\mathrm{r}}$ の $\sigma^{+\mathrm{r}-\mathrm{f}}$-構造への制限 $\mathcal{A}^{+\mathrm{r}-\mathrm{f}}$ を考える．このとき，任意の付値 (v, V) に対し，次が成り立つ．

$$[\![\varphi]\!]^{\mathcal{A}}_{(v,V)} = [\![\varphi^{+\mathrm{r}-\mathrm{f}}]\!]^{\mathcal{A}^{+\mathrm{r}-\mathrm{f}}}_{(v,V)}$$

(これは $\mathcal{A} \models \varphi \Longleftrightarrow \mathcal{A}^{+\mathrm{r}-\mathrm{f}} \models \varphi^{+\mathrm{r}-\mathrm{f}}$ と同値)

この事実から，一階様相 μ 計算の構造と式の組 $(\mathcal{A}^{+\mathrm{r}-\mathrm{f}}, \varphi^{+\mathrm{r}-\mathrm{f}})$ は函数記号一階様相 μ 計算の構造と式の組 $(\mathcal{A}, \varphi)$ の「抽象化」であることが分かる．これは，函数記号付一階様相 μ 計算でのモデル検査を，一階様相 μ 計算でのモデル検査に変換することができることを意味する．

4 函数記号付一階様相 μ 計算における体系の例

本節ではまず，函数記号付一階様相 μ 計算の上に自然数論の形式的理論を導入し，これを用いて，二項関係の推移的閉包を定義する公理を例示する．なお，前節で示したように，これらの公理は函数記号消去した式で表すこともできる．最後に，推移的閉包を用いた検証項目の記述例を例示する．

函数記号付一階様相 μ 計算の上の形式的理論 $\mathrm{Th}(\mathbb{N})$ を次のように定める．$\mathrm{Th}(\mathbb{N})$ の語彙 $\sigma(\mathbb{N})$ は定数記号 `0`，一変数函数記号 `suc`，一変数述語記号

$\mathtt{Nat}$ からなるとし，$\mathrm{Th}(\mathbb{N})$ の公理は以下に示す式とする．

1. $\mathtt{Nat}(\mathtt{0})$
2. $\forall n.\, \mathtt{Nat}(n) \supset \mathtt{Nat}(\mathtt{suc}(n))$
3. $\forall n.\, \mathtt{Nat}(n) \supset \mathtt{0} \neq \mathtt{suc}(n)$
4. $\forall m, n.\, \mathtt{Nat}(m) \supset \mathtt{Nat}(n) \supset \mathtt{suc}(m) = \mathtt{suc}(n) \supset m = n$
5. 任意の論理式 $\varphi[x]$ について，$\varphi[\mathtt{0}] \wedge (\forall m.\, \mathtt{Nat}(m) \supset \varphi[m] \supset \varphi[\mathtt{suc}(m)]) \supset (\forall n.\, \mathtt{Nat}(n) \supset \varphi[n])$

さらに，語彙 $\sigma(\mathbb{N})$ に二変数述語記号 $\mathtt{E}$, $\mathtt{E}^*$ と 三変数述語記号 $\mathtt{E}'$ を加えたものを $\sigma(\mathrm{TC})$ とし，$\mathrm{Th}(\mathbb{N})$ に次の公理を加えたものを $\mathrm{Th}(\mathrm{TC})$ とする．

1. $\forall x, y.\, \mathtt{E}'(\mathtt{0}, x, y) \equiv \mathtt{E}(x, y)$
2. $\forall x, y, n.\mathtt{Nat}(n) \supset \mathtt{E}'(\mathtt{suc}(n), x, y) \equiv (\exists z.\, \mathtt{E}'(n, x, z) \wedge \mathtt{E}(z, y))$
3. $\forall x, y.\, \mathtt{E}^*(x, y) \equiv (\exists n.\, \mathtt{Nat}(n) \wedge \mathtt{E}'(n, x, y))$

このとき $\mathrm{Th}(\mathrm{TC})$ のモデル $\langle S, R, D, I\rangle$ と状態 s に対し，$\langle I(\mathtt{Nat}, s), I(\mathtt{0}, s), I(\mathtt{suc}, s)\rangle$ が $\mathrm{Th}(\mathbb{N})$ の標準モデルと一階述語論理の $\{\mathtt{0}, \mathtt{suc}\}$-構造として同型であれば，$\mathtt{E}^*$ は s で $\mathtt{E}$ の推移的閉包になり，そうでない場合は，超準的な推移的閉包になる．

続いて，上で定義した語彙 $\sigma(\mathrm{TC})$ と形式的理論 $\mathrm{Th}(\mathrm{TC})$ を用いた検証項目の記述例を示す．コンピュータネットワークにおいて，要素 (ノード) 間の到達可能性を知ることは重要であり，ある状態 (時刻) における到達可能性は transitive closure logic で記述される．しかし，コンピュータネットワークの接続状況は，故障や災害により時刻と共に変化する．したがって，ある状態における到達可能性よりも「どの要素にも，(時間関係 R で) いつか必ず，(接続関係 E で) 到達できる」という，transitive closure logic と時相論理を併せた論理でしか記述できない性質が重要になる．

そこで，要素間の (直接的) 接続を二変数述語記号 $\mathtt{E}$ を用いて表すことにする．すると，要素間の到達可能性 (間接的接続) は $\mathtt{E}^*$ を用いて表される．また，一階様相 μ 計算では述語記号の解釈は状態に依存して変化するので，接続状況の時間による変化は，$\mathtt{E}$ の解釈の状態毎の変化として表される．

このとき，$\mathcal{A}$ を $\mathrm{Th}(\mathrm{TC})$ の標準的なモデル，(v, V) を付値とすると，下式

$$s_0 \in [\![\forall x.\, \mu X.\, T^*(x_0, x) \vee \Box X]\!]^{\mathcal{A}}_{(v,V)}$$

は「モデル $\mathcal{A}$ の状態 s_0 において，要素 $v(x_0)$ から，どの要素にも，(時間関係で) いつか必ず，(接続関係で) 到達できる」ことを表す．(自由変数 x_0 の解釈は状態に依存しないで決まることに注意)

5 結論

(函数記号付) 一階様相 μ 計算は，一階述語論理の拡張であるが，帰納的に公理化不可能であり，他にもコンパクト性定理が成り立たないなど，一階述語論理と異なる点も多い．したがって，一階述語論理で成り立つ性質が，一階様相 μ 計算でも成り立つかどうかは，必ずしも明らかではない．本論文では，函数記号消去が，函数記号付一階様相 μ 計算においても，一階述語論理に対するのと同様に成り立つことを示した．

他方，函数記号消去は一種の (モデル論的) 抽象化であり，抽象化はモデル検査などで広く利用されている重要な手法である．函数記号消去以外の (モデル論的) 抽象化には量化記号消去があり，これが一階様相 μ 計算のある種の形式的理論で成り立つかどうかは知られていない．

参考文献

[1] Melvin Fitting and Richard L. Mendelsohn, *First-order Modal Logic*, Synthese Library/volume 277, (1999), Kluwer Academic Publications

[2] David Harel, Dexter Kozen, Jerzy Tiuryn, *DYNAMIC LOGIC*, Foundations of Computing Series, (2000), The MIT Press

[3] Wilfrid Hodges, *Model Theory*, Encyclopedia of Mathematics and its Applications Volume 42 (1993) Cambridge

[4] Ian Hodkinson, Frank Wolter and Michael Zakharyaschev, *Decidable fragments of first-order temporal logics*, Annals of Pure and Applied Logic 106 (2000) 85-134

[5] Dexter Kozen, *Results on the Propositional μ-calculus*, Theoretical Computer Science 27 (1983) 333-354

[6] Zohar Manna, Amir Pnueli, *The Temporal Logic of Reactive and Concurrent Systems :Specification*, (1991) Springer-Verlag

[7] Keishi Okamoto, *A First-Order Extension of Modal μ-calculus*, Programming Science Technical Report, AIST/CVS (2006), http://unit.aist.go.jp/cvs/tr-data/PS06-003.pdf