Vol. 34 No. 4 情報処理学会論文誌 Apr. 1993

時間的特性を考慮した並列プロセスの形式的記述

佐藤一郎† 所 真理雄†

並列計算では計算の動作内容だけでなく同期待ち時間や実行時間などの時間的特性が，計算の効率や正当性を与える上で重要な要素となる．しかし，従来の並列計算モデルではこうした時間的特性を明示的に表現することができなかった．そこで本論文では既存のプロセス計算の体系に時間経過と時間に依存した動作の表現能力を拡張することによって，時間的特性が表現可能な並列計算のための形式系を構築する．この形式系はプロセス計算の有用な特性を保持しながら，同時に並列プロセス間の通信や同期操作などの相互作用における動作内容とその時間性の両方を明示的に表現できる形式系である．本論文ではさらに，この形式系に基づく証明技法として時間性を考慮した二つのプロセスの等価性，時間的強等価と時間的観測等価を与える．これらは二つのプロセスが動作内容的にも時間的にも等価であることを調べるもので，特に時間的観測等価は内部的な動作を隠蔽して外部的相互作用だけに基づいて等価性を判定することができる．本論文ではこれらの等価性の代数的特性を調べ，さらに時間的強等価に基づく有限プロセスのための健全かつ完全な公理系を与える．最後に，線形接続されたマルチプロセッサ間の通信をこの形式系により記述し，その動作および時間性を解析する例を与える．

A Formal Description for Parallel Processes with Time Properties

Ichiro Satoh† and Mario Tokoro†

We investigate a formal model for reasoning about time properties in parallel computing. We develop a timed process calculus which is an extension of Milner's CCS with two notions for time: the passage of time and behaviors dependent on time. It cannot represent only functional results in parallel computing but also various time properties, such as execution time, synchronization delay time, and timeout handling. We define two timed equivalences based on CCS's bisimulation. These equivalences can analyze whether two processes are equivalent in both their functional behavior and time properties. We derive their algebraic laws for reasoning about processes with time properties and define a sound and complete equational proof system for finite processes based on the equivalence. An example is shown in order to demonstrate the utility of our calculus.

1. はじめに

並列計算を考える上で計算に含まれる時間性を考慮することは重要である．

特に並列プロセス間の通信や同期操作時に生じる同期待ち時間は，一部のプロセッサのアイドリングを引き起こし，全体の計算時間にも大きく影響することから事前に解析しておくことが重要となる．本論文では並列計算におけるプロセス間の同期通信と同期待ち時間に着目し，これらを形式的に扱う体系を考える．

従来，並列計算に含まれる時間性の表現形式として，時相論理体系や Timed Petri Net などが用いられてきたが，これらは並列動作体間の通信の表現性が十分とはいえない．ところが近年，通信プロトコルの記述などを目的に，CCS[6] や CSP[4] などの既存のプロセス計算に時間概念を導入した体系が幾つか提案されている[12),10)]．プロセス計算は通信に基づいて並列計算を抽象化する理論的な体系のため，上記の表現形式として優れた特性が期待できるが，既存の時間拡張体系には時間表現性や通常のプロセス計算とその親和性に関して以下に述べるような問題がある．

多くのプロセス計算の時間拡張体系[3),5),14)]は，事象の実行可能性を所定時間だけ遅延させる操作を導入することによって時間性を表現している．しかし，この方法では逆に既に実行可能になった事象を時間経過に従って再び実行不可能にすることができないので，タイムアウト処理のように事象の時間的制限をもつ動作が表現できないという問題点がある．一方，時間による事象の実行可能性の制限が可能な体系 Temporal

† 慶應義塾大学理工学研究科計算機科学専攻
Department of Computer Science, Faculty of Science and Technology, Keio University

CCS[7),8)], ATP[9)], TPCCS[2)] が存在するが，これらには次のような問題がある．TPCCS ではタイムアウト処理と類似した選択演算子を CCS に導入して事象の時間制限を表現するが，この演算子ではタイムアウト前の動作に当たるプロセスにおいて時間が経過しないので，時間的な動作をもつプロセスのためのタイムアウト処理が表現できない．これに対して，Temporal CCS は CCS の選択演算子 (+) を拡張した時間経過による選択演算子を CCS に導入することにより，そして，ATP は ACP[1)] に類似した体系に時間的な選択演算子を導入することにより，時間経過による事象の制限と TPCCS の問題を解決している．しかし，Temporal CCS と ATP では実行可能な通信の実行開始タイミングが非決定的であるために，通信のための同期待ち時間が正確に解析できない問題がある．また，Temporal CCS, ATP, TPCCS は観測性に基づくプロセスの等価性の概念がなく，仕様記述や検証などで不要な状態数が増大し複雑になるという問題点がある．

本論文では，プロセス計算の中でも並列計算の仕様記述や検証に広く用いられている Robin Milner の CCS[6)] を取り上げ，これをもとにタイムアウトなどの時間制限が表現可能で，さらに観測性に基づく抽象化が可能な形式系 RtCCS (Real-time CCS) を提案し，さらにその形式系のための証明技法を与える．

本論文では，次章で CCS への拡張の概要を示し，3章で RtCCS を定義し，4章で RtCCS プロセスの等価関係を与える．5章で等価則からなる公理系を与え，続く6章で記述例を示し，最後の章で結論と課題について述べる．

2. 時間概念の拡張

RtCCS では，以下に概説する時間的概念を CCS に拡張することにより，CCS の有用な特性を最大限に残しながら時間的表現性を実現する．

時間経過の表現

プロセス計算の計算は事象の実行から構成され，その事象は原子的で不可分であるという特徴がある．われわれの形式系ではプロセス計算の特性を保持するために，時間経過も事象により表現し，さらに通常の事象の実行時間をゼロとして事象の原子性を保持する．

ここでは，その時間の経過を表す事象を時刻印事象と呼び，$\surd$ と記述する．時刻印事象はすべてプロセスで同時に実行される事象で，一回の実行が単位時間の時間経過に相当する．その結果，RtCCS の時間は離散的な時間となる．

時間依存した動作の表現

システムによっては，遅延操作やタイムアウト操作のように時間経過量によって動作内容が変化する処理をもっている．われわれは遅延操作だけでなくタイムアウト操作のように時間による事象の実行可能性の制限を可能にするために，タイムアウト処理に類似した意味をもつ特別な二項演算子を導入する．ここではこれを時限演算子と呼び，$\langle , \rangle_t$ と記述する．例えば，時限演算子 $\langle P, Q\rangle_t$ では $\langle P, Q\rangle_t$ となってからタイムアウトするまでの時間を，プロセス P はタイムアウト前の動作を，Q はタイムアウト後の動作を表す．これは**図1**のように遷移し，P が t 単位時間内に通信または内部事象 (α) が実行できないときは Q となり P の実行可能性は失われることになる．このとき，この時限演算子は TPCCS[2)] の時間演算子と異なり，時間経過がタイムアウト前動作に対応するプロセス P にも伝わることから，時間的な動作をもつプロセスのタイムアウトが表現できる．

内部事象の実行優先

前述の時限演算子は，直前の事象を実行してからの時間経過量によって通信可能な事象や内部事象の実行可能性を制限するが，これは通信や内部事象の実行タイミングにより以降の動作内容が変化することを意味する．このことは，内部事象を無視して取り扱う観測等価性において，無視されたはずの内部事象の実行タイミングによって以降の動作内容が変化することになり，CCS の観測性に関する特性の多くを保持できなくなる．特に，CCS における実行可能な事象は，他の事象が選択実行されるとき以外は実行可能性が失われることがないという CCS の性質に影響するので，この実行可能性の制限は CCS との親和性に関しても重大な問題となる．そこで，「実行可能な通信と内部事象は次の時刻印演算子 $\surd$ が実行される前に実行され

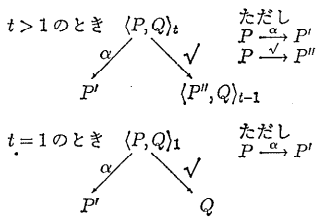


図 1 時限演算子 $\langle P, Q\rangle_t$ の遷移
Fig. 1 The transitions of $\langle P, Q\rangle_t$.

る」という制限を設けることにより，これらの問題を解決する．また，この制限により，実行可能な同期通信における不要な遅延が禁止されることにより同期通信のための最小同期待ち時間の正確な測定や，後述するように観測等価性においては内部遷移の無限実行[13]の判定が可能になる．

3. RtCCS の定義

RtCCS では以下に定義する動作式によりプロセスを記述する．ここでは動作式の記法と構文，さらにその意味論を与える．

3.1 動 作 式

まず，定義に用いる事象名およびその集合の記号を与える．RtCCS の事象には，入力事象と出力事象からなる通信事象と，外部的に観測されない事象である内部事象と時刻印事象がある．

定義 3.1（事象）

1. 入力事象名の集合を $\mathcal{A}$ とし，その要素を $a, b, \cdots$ と記述する．
2. 出力事象名の集合を $\overline{\mathcal{A}}$ とし，その要素を $\bar{a}, \bar{b}, \cdots$ と記述する．ただし $\bar{\bar{a}} \equiv a$ である．
3. 内部事象を τ，また時刻印事象を $\surd$ と記述する．
4. 通信事象名の集合を $\mathcal{L} \equiv \mathcal{A} \cup \overline{\mathcal{A}}$ とし，$l, l', \cdots$ を $\mathcal{L}$ 上の要素とする．
5. 通信事象名と内部事象の集合を $Act \equiv \mathcal{L} \cup \{\tau\}$ とし，$\alpha, \beta, \cdots$ を Act 上の要素とする．
6. すべての事象の集合を $Act_\surd \equiv Act \cup \{\surd\}$ とし，$\mu, \nu, \cdots$ を $Act_\surd$ 上の要素とする ■

定義 3.2（動作式） 以下のような構文規則を持つ式の集合を動作式集合 $\mathcal{E}$ とし，その要素を $E, E_1, E_2, \cdots$ と記す．

$$
\begin{array}{ll}
E ::= \mathbf{0} & \text{(停止プロセス)} \\
\quad | \ X & \text{(プロセス変数)} \\
\quad | \ \alpha.E & \text{(逐次合成)} \\
\quad | \ E_1 + E_2 & \text{(選択合成)} \\
\quad | \ E_1 | E_2 & \text{(並列合成)} \\
\quad | \ E[f] & \text{(事象名変更)} \\
\quad | \ E \backslash L & \text{(事象制限)} \\
\quad | \ \mathbf{rec}X : E & \text{(再帰)} \\
\quad | \ \langle E_1, E_2 \rangle_t & \text{(時限合成)}
\end{array}
$$

ここで，f を事象名の変更関数 $f : Act_\surd \to Act_\surd$，ただし，任意の f において $f(\overline{l}) = \overline{f(l)}$，$f(\tau) = \tau$，$f(\surd) = \surd$ とする，L は $\mathcal{L}$ の部分集合，$\mathbf{rec}X : E$ の E 中に X がある場合はその X はガードされているとする☆．

さらに，自由変数を含まない動作式の集合を $\mathcal{P}$ とし（$\mathcal{P} \subseteq \mathcal{E}$），その要素を $P, Q, \cdots$ と記述する☆☆． ■

ここで，RtCCS の演算子の意味を非形式的に述べておく．$\mathbf{0}$ はデッドロックまたは終了したプロセスを表す，$\alpha.E$ は事象 α を実行後，動作式 E として振る舞うことを表す，$E_1 + E_2$ は E_1 と E_2 のどちらかを実行することを表す，$E_1 | E_2$ は E_1 と E_2 が並列に実行されることを表す，$E[f]$ は E 中の事象名を関数 f で変更する，$E \backslash L$ は E に含まれる事象のうち集合 $L \cup \overline{L}$ に含まれる事象の観測を制限する，$\mathbf{rec}X : E$ は，E が E 中に含まれる変数 X に束縛されることを表す，ただし，今後しばしば，$X \stackrel{\mathrm{def}}{=} E$ と略記することがある．

3.2 操作的意味論

RtCCS はラベルつき遷移システム $\langle \mathcal{E}, Act_\surd, \{\stackrel{\mu}{\to} \mid \mu \in Act_\surd\} \rangle$ として与えられる．ここで $\stackrel{\mu}{\to}$ は，**図 2** に示される推論規則を満足する最小の関係として定義される遷移関係（$\stackrel{\mu}{\to} \subseteq \mathcal{E} \times \mathcal{E}$）である．RtCCS の操作性意味論はこの遷移関係によって与えられる．

ここで，不定回の内部遷移（τ）を含む遷移関係を定義する．

$$
\frac{}{\alpha.E \stackrel{\alpha}{\to} E} \qquad \frac{}{\ell.E \stackrel{\ell}{\to} \ell.E} \qquad \frac{}{\mathbf{0} \stackrel{\surd}{\to} \mathbf{0}}
$$

$$
\frac{E_1 \stackrel{\alpha}{\to} E_1'}{E_1 + E_2 \stackrel{\alpha}{\to} E_1'} \qquad \frac{E_2 \stackrel{\alpha}{\to} E_2'}{E_1 + E_2 \stackrel{\alpha}{\to} E_2'}
$$

$$
\frac{E_1 \stackrel{\surd}{\to} E_1',\ E_2 \stackrel{\surd}{\to} E_2'}{E_1 + E_2 \stackrel{\surd}{\to} E_1' + E_2'}
$$

$$
\frac{E_1 \stackrel{\alpha}{\to} E_1'}{E_1 | E_2 \stackrel{\alpha}{\to} E_1' | E_2} \qquad \frac{E_2 \stackrel{\alpha}{\to} E_2'}{E_1 | E_2 \stackrel{\alpha}{\to} E_1 | E_2'}
$$

$$
\frac{E_1 \stackrel{a}{\to} E_1',\ E_2 \stackrel{\bar{a}}{\to} E_2'}{E_1 | E_2 \stackrel{\tau}{\to} E_1' | E_2'}
$$

$$
\frac{E_1 \stackrel{\surd}{\to} E_1',\ E_2 \stackrel{\surd}{\to} E_2',\ \neg\exists E :\ E_1 | E_2 \stackrel{\tau}{\to} E}{E_1 | E_2 \stackrel{\surd}{\to} E_1' | E_2'}
$$

$$
\frac{E \stackrel{\alpha}{\to} E',\ \alpha \notin L \cup \overline{L}}{E \backslash L \stackrel{\alpha}{\to} E' \backslash L} \qquad \frac{E \stackrel{\surd}{\to} E'}{E \backslash L \stackrel{\surd}{\to} E' \backslash L}
$$

$$
\frac{E \stackrel{\mu}{\to} E'}{E[f] \stackrel{f(\mu)}{\to} E'[f]} \qquad \frac{E\{\mathbf{rec}X : E/X\} \stackrel{\mu}{\to} E'}{\mathbf{rec}X : E \stackrel{\mu}{\to} E'}
$$

$$
\frac{E_1 \stackrel{\alpha}{\to} E_1',\ t > 0}{\langle E_1, E_2 \rangle_t \stackrel{\alpha}{\to} E_1'} \qquad \frac{E_1 \stackrel{\surd}{\to} E_1',\ t > 0}{\langle E_1, E_2 \rangle_t \stackrel{\surd}{\to} \langle E_1', E_2 \rangle_{t-1}}
$$

$$
\frac{E_2 \stackrel{\mu}{\to} E_2'}{\langle E_1, E_2 \rangle_0 \stackrel{\mu}{\to} E_2'}
$$

図 2 RtCCS の推論規則
Fig. 2 Inference rules of RtCCS.

☆ $\alpha.X$ の $\alpha \in Act$（ただし，α は空文字ではない）をガードという．ガードされてないとは $\mathbf{rec}X : X$，$\mathbf{rec}X : (X + E)$ などの式をいう．

☆☆ 以下，動作式を単にプロセスとして呼ぶことがある．

定義 3.3 $E, F\in\mathcal{E}$, $\mu\in Act_g$ とする．ただし，$(\xrightarrow{\tau})^*$ は $\xrightarrow{\tau}$ の不定回実行を示す．

（i） $E\overset{\mu}{\Rightarrow}F$ とは $E(\xrightarrow{\tau})^*\xrightarrow{\mu}(\xrightarrow{\tau})^*F$

（ii） $E\overset{\varepsilon}{\Rightarrow}F$ とは $E(\xrightarrow{\tau})^*\xrightarrow{\mu}(\xrightarrow{\tau})^*F$

ただし，$\mu=\tau$ のときは $E(\xrightarrow{\tau})^*F$ ■

4. 時間的双模倣性

時間的特性をもつ並列計算の動作内容および時間的特性に関する検証技法として，時間的概念をもつプロセスの等価関係を定義する．これは双模倣性 (Bisimulation)[6),11)] を拡張することにより，動作内容だけでなく時間的特性に関しても等価判定ができるようにしたものである．

4.1 時間的強等価

まず，内部事象を含むすべての事象を対象とする双模倣性である時間的強等価を定義する．

定義 4.1（時間的強等価） 二つのプロセス $P, Q\in\mathcal{P}$ 上の関係 $\mathcal{S}$ が時間的強双模倣であるとは，$(P, Q)\in\mathcal{S}$ ならば任意の $\mu\in Act_g$ について次の二つの条件が成立することである

（i） $\forall P' : P\overset{\mu}{\Rightarrow}P'\supset\exists Q' : Q\overset{\mu}{\Rightarrow}Q'\wedge(P', Q')\in\mathcal{S}$.

（ii） $\forall Q' : Q\overset{\mu}{\Rightarrow}Q'\supset\exists P' : P\overset{\mu}{\Rightarrow}P'\wedge(P', Q')\in\mathcal{S}$.

時間的強双模倣 $\mathcal{S}$ の最大の集合を時間的強等価 $\sim_g$ とする． ■

命題 4.2

1. $\mathcal{S}_i(i\in I)$ が時間的強双模倣ならば，$\cup_{i\in I}\mathcal{S}_i$ も時間的強双模倣．
2. $\sim_g$ は同値関係．

(証明) 定義 4.1 より容易に示せる． ■

二つのプロセスの動作式 P と Q が時間的強等価 ($P\sim_g Q$) であるとは，P, Q において両者の実行可能なすべての事象が互いに一致し，さらにその事象を実行した後の状態 P', Q' において同様な関係が成立することを表している．特に，RtCCS では通信や内部動作だけでなく時間経過も事象として表されるため，両者の事象が一致するとは二つのプロセスが動作内容的にも時間的にも互いに模倣しあうことを意味する．次に時間的強等価に関する基本的な性質を示す．

命題 4.3（時間的強等価の性質 1）

（1） $P+Q\sim_g Q+P$

（2） $P+(Q+R)\sim_g(P+Q)+R$

（3） $P+P\sim_g P$

（4） $P+\mathbf{0}\sim_g P$

(証明) ここでは，(1) の場合のみ成立を示す．$P+Q$ の実行可能遷移を μ とする．定義 4.1 と命題 4.2 より，$\mathcal{S}=\{(P+Q, Q+P)\mid P, Q\in\mathcal{P}\}$ が時間的強双模倣であることを示せばよい．$\mu\in Act$ の場合は，CCS[6)] と同様．$\mu=\surd$ の場合は，$P\xrightarrow{\surd}P', Q\xrightarrow{\surd}Q'$ とすると $P+Q\xrightarrow{\surd}P'+Q'$ かつ $Q+P\xrightarrow{\surd}Q'+P'$ より，$(P'+Q', Q'+P')\in\mathcal{S}$ となるので，$\mathcal{S}$ は定義 4.1 の (i) を満足する．一方 (ii) も同様．他の場合も (1) と同様に示せる． ■

命題 4.4（時間的強等価の性質 2）

（1） $P\mid Q\sim_g Q\mid P$

（2） $P\mid(Q\mid R)\sim_g(P\mid Q)\mid R$

（3） $P\mid\mathbf{0}\sim_g P$

（4） $(P+Q)\backslash L\sim_g P\backslash L+Q\backslash L$

（5） $(P\mid Q)\backslash L\sim_g P\backslash L\mid Q\backslash L$ ここで，$\Lambda(P)\cap\overline{\Lambda(Q)}\cap(L\cup\bar{L})=\phi$

（6） $(\alpha.P)\backslash L\sim_g\begin{cases}\mathbf{0} & \alpha\in L\cup\bar{L}\ \text{のとき}\\ \alpha.P\backslash L & \text{上記以外}\end{cases}$

（7） $(P+Q)[f]\sim_g P[f]+Q[f]$

（8） $(P\mid Q)[f]\sim_g P[f]\mid Q[f]$

（9） $(\alpha.P)[f]\sim_g f(\alpha).P[f]$

ここで，$\Lambda(P)$ はプロセス P に現れる通信事象名全体の集合を表す．

(証明) Act 内の事象の遷移の場合は CCS[6)] と，$\surd$ 事象の場合は命題 4.3 と同様に証明することができる． ■

命題 4.5（時間的強等価の性質 3）

（1） $\langle\alpha.P, \alpha.P\rangle_t\sim_g\alpha.P$

（2） $\langle\alpha.P, \langle\alpha.P, Q\rangle_t\rangle_s\sim_g\langle\alpha.P, Q\rangle_{s+t}$

（3） $\langle\tau.P+Q, R\rangle_t\sim_g\tau.P+Q\quad(t>0)$

（4） $\langle P, Q\rangle_t\backslash L\sim_g\langle P\backslash L, Q\backslash L\rangle_t$

（5） $\langle P, Q\rangle_t[f]\sim_g\langle P[f], Q[f]\rangle_t$

（6） $\langle P, Q+R\rangle_t\sim_g\langle P, Q\rangle_t+\langle P, R\rangle_t$

（7） $\langle P+Q, R\rangle_t\sim_g\langle P, R\rangle_t+\langle Q, R\rangle_t$

（8） $\langle\langle P, Q\rangle_s, R\rangle_t\sim_g\begin{cases}\langle P, R\rangle_t & (s>t)\\ \langle P, \langle Q, R\rangle_{t-s}\rangle_s & (s<t)\end{cases}$

(証明) 最も証明が困難な (8) の $(s<t)$ の証明を示す．他の場合も同様である．まず，左辺を Q_1, 右辺を Q_2 とする．Q_1 の実行可能事象を $\mu\in Act_g$ とする．$\mu\in Act$, $P\xrightarrow{\mu}P'$ のとき．$Q_1\xrightarrow{\mu}P'$, $Q_2\xrightarrow{\mu}P'$ より明らか．次に $\mu=\surd$ のときを考える．まず $s=0$ の場合は明らか．定義 4.1，命題 4.2 より，$s>0$ の場合は次の $\mathcal{S}$ が時間的強双模倣であることを示せば十分．$\mathcal{S}=\{(\langle\langle P_1, P_2\rangle_s, P_3\rangle_t, \langle P_1, \langle P_2, P_3\rangle_{t-s}\rangle_s)\mid 0\leq s<t, P_1, P_2, P_3\in\mathcal{P}\}$．$P_1\xrightarrow{\surd}P_1'$ とすると，$Q_1\xrightarrow{\surd}\langle\langle P_1', P_2\rangle_{s-1}, P_3\rangle_{t-1}(\equiv Q_1')$．$Q_2\xrightarrow{\surd}\langle P_1', \langle P_2, P_3\rangle_{t-s}\rangle_{s-1}(\equiv Q_2')$ となり，$(Q_1', Q_2')\in\mathcal{S}$ ■

$Q_2')\in\mathcal{S}$. よって定義 4.1 の(i)を満足する. (ii)も同様. ∎

命題 4.6 (展開規則)

(1) $P\equiv\sum_{i\in I}\alpha_i.P_i$ かつ $Q\equiv\sum_{j\in J}\beta_j.Q_j$ のとき
$P|Q\sim_g\sum_{i\in I}\alpha_i.(P_i|Q)+\sum_{j\in J}\beta_j.(P|Q_j)+\sum_{\alpha i=\overline{\beta}j}\tau.(P_i|Q_j)$

(2) $P\equiv\langle\sum_{i\in I}\alpha_i.P_i,P'\rangle_1$ かつ $Q\equiv\sum_{j\in J}\beta_j.Q_j$ のとき
$P|Q\sim_g\langle\sum_{i\in I}\alpha_i.(P_i|Q)+\sum_{j\in J}\beta_j.(P|Q_j)+\sum_{\alpha i=\overline{\beta}j}\tau.(P_i|Q_j),P'|Q\rangle_1$

(3) $P\equiv\langle\sum_{i\in I}\alpha_i.P_i,P'\rangle_1$ かつ $Q\equiv\langle\sum_{j\in J}\beta_j.Q_j,Q'\rangle_1$ のとき
$P|Q\sim_g\langle\sum_{i\in I}\alpha_i.(P_i|Q)+\sum_{j\in J}\beta_j.(P|Q_j)+\sum_{\alpha i=\overline{\beta}j}\tau.(P_i|Q_j),P'|Q'\rangle_1$

(証明) 命題 4.3〜4.5 と同様 ∎

この展開規則により並列プロセスを等価な逐次プロセスに変換することができる. この展開規則では時限演算子の制限時間は1の場合を示す. 2以上の場合は時限演算子の代数的な意味が大きく変化し, 洗練された形での結果を得られていない.

命題 4.7 (合同性) 時間的強等価は合同関係である.

(証明) 時限演算子に関する場合として, $P_1,P_2,Q\in\mathcal{P}$ かつ $P_1\sim_gP_2$ のとき $\langle P_1,Q\rangle_t\sim_g\langle P_2,Q\rangle_t$ の証明を示す. $\langle P_1,Q\rangle\overset{\mu}{\rightarrow}R_1$ とする. $\mu\in Act$ のときは $P_1\sim_gP_2$ より明らか. 次に $\mu=\surd$ のときを考える. $t=0$ の場合は明らか. 一方, $t>0, P_1\overset{\mu}{\rightarrow}P_1'$ の場合は, 定義 4.1, 命題 4.2 より, 次の $\mathcal{S}$ が時間的強双模倣であることを示せば十分, $\mathcal{S}=\{(\langle P_1,Q\rangle_t,\langle P_2,Q\rangle_t):P_1\sim_gP_2,t>0\}$. $P_1\sim_gP_2$ より $P_2\overset{\mu}{\rightarrow}P_2'$ かつ $P_1'\sim_gP_2'$ となる P_2' が存在して, $\langle P_1,Q\rangle_t\overset{\mu}{\rightarrow}\langle P_1',Q\rangle_{t-1},\langle P_2,Q\rangle_t\overset{\mu}{\rightarrow}\langle P_2',Q\rangle_{t-1}$ より $(\langle P_1',Q\rangle_{t-1},\langle P_2',Q\rangle_{t-1})\in\mathcal{S}$ となる. これにより定義 4.1 の(i)に示せた. (ii)についても同様である. さらに, 時限演算子に関するもう一つの場合, $\langle Q,P_1\rangle_t\sim_g\langle Q,P_2\rangle_t$ も同様.

また, 時限演算子以外の演算子に関しては Act 事象の遷移は CCS[6] と同様, $\surd$ 遷移は上記と同様である. ∎

時間的強等価は合同関係であることより, 時間的強等価である二つのプロセスは互いに入れ換えても, そのプロセスを構成要素とするシステム全体の動作と時間性に影響を与えないことになる. このため, 時間的強等価はプロセスの再利用性や置換可能性の判定の基礎となる.

4.2 時間的弱双模倣性

プロセス間の相互作用を考える場合, プロセスの内部的な動作より外部環境との通信やその実行タイミングが重要となる. そこで, 観測可能な事象とその実行タイミングだけを比較対象とする等価関係を CCS の観測等価[6]をもとに定義する.

定義 4.8 (時間的観測等価) 二つのプロセス $P,Q\subset\mathcal{P}$ 上の関係 $\mathcal{S}$ が時間的弱双模倣であるとは, $(P,Q)\in\mathcal{S}$ ならば任意の $\mu\in Act_g$ について次の二つの条件が成立することである.

(i) $\forall P':P\overset{\hat{\mu}}{\Rightarrow}P'\supset\exists Q':Q\overset{\hat{\mu}}{\Rightarrow}Q'\wedge(P',Q')\in\mathcal{S}$.

(ii) $\forall Q':Q\overset{\hat{\mu}}{\Rightarrow}Q'\supset\exists P':P\overset{\hat{\mu}}{\Rightarrow}P'\wedge(P',Q')\in\mathcal{S}$.

時間的弱双模倣 $\mathcal{S}$ の最大の集合を時間的観測等価 $\approx_g$ とする. ∎

命題 4.9

1. $\mathcal{S}_i\ (i\in I)$ が時間的弱双模倣ならば, $\cup_{i\in I}\mathcal{S}_i$ も時間的弱双模倣.
2. $\approx_g$ は同値関係.

(証明) 定義 4.8 より容易に示せる. ∎

この時間的観測等価 $\approx_g$ は, 二つのプロセスが互いに通信事象と時間経過を模倣し合い, 時計をもつ外部観測者から両者が区別できないことを表す.

ところで, CCS の観測等価[6]と時間的観測等価の最大の違いは, 時間的な等価判定性の有無であるが, さらに両者は内部遷移による無限実行 (Divergence)[13] の取扱いに違いがある. CCS の観測等価では τ 遷移の捨象により, 内部遷移による無限実行が外部的に観測不可能で, 内部的無限ループを含むプロセスと含まないプロセスが等価になる可能性があるという問題があった. しかし, RtCCS では実行可能な内部事象は必ず時刻印事象より先に実行されるという制限により, 時刻印事象の実行は内部事象の無限実行は存在しないことを意味する. この結果, 時間的観測等価な二つのプロセスにおいてどちらか一方だけに内部事象の無限実行が存在することがなく, 時間的観測等価は内部的無限ループを明示的に扱う等価関係であるといえる.

命題 4.10 (時間的観測等価の性質)

(1) $\tau.P\approx_gP$
(2) $\alpha.\tau.P\approx_g\alpha.P$
(3) $\tau.P+P\approx_g\tau.P$
(4) $\alpha.(\tau.P+Q)\approx_g\alpha.(\tau.P+Q)+\alpha.Q$
(5) $\langle\tau.P,Q\rangle_t\approx_gP\quad(t>0)$
(6) $\langle P,\tau.Q\rangle_t\approx_g\langle P,Q\rangle_t$

(7) $\langle \alpha.P, \tau.\langle \alpha.P, Q\rangle_s\rangle_t \approx_g \langle \alpha.P, Q\rangle_{s+t}$

(証明) 命題 4.3, 4.5 と同様 ∎

RtCCS の時間的観測等価は選択演算子と時限演算子で保存されない．しかし，時間的観測等価をもとに合同関係を定義できる．

定義 4.11 (時間的観測合同) 二つのプロセス $P, Q \in \mathcal{P}$ 上の関係 $\mathcal{S}$ が時間的観測合同であるとは，$(P,Q)\in\mathcal{S}$ ならば任意の $\mu \in Act_g$ について次の二つの条件が成立することである．

(i) $\forall P' : P \Rightarrow P' \supset \exists Q' : Q \Rightarrow Q' \wedge P' \approx_g Q'$.

(ii) $\forall Q' : Q \Rightarrow Q' \supset \exists P' : P \Rightarrow P' \wedge P' \approx_g Q'$.

プロセス P と Q が時間的観測合同であるとき，$P =_g Q$ と書く． ∎

時間的観測等価と時間的観測合同の違いは，右辺の遷移関係が $\Rightarrow$ から $\Rightarrow$ に変わることにある．これは時間的観測等価では τ 遷移を捨象することができたが，時間的観測合同では τ 遷移は最初の一回だけ捨象できないことを意味する．

命題 4.12 (時間的観測合同の性質)

(1) $\alpha.\tau.P =_g \alpha.P$

(2) $\tau.P + P =_g \tau.P$

(3) $\alpha.(\tau.P+Q) =_g \alpha.(\tau.P+Q)+\alpha.Q$

(4) $\langle P, \tau.Q\rangle_t =_g \langle P, Q\rangle_t \quad (t>0)$

(5) $\langle \alpha.P, \tau.\langle \alpha.P, Q\rangle_s\rangle_t =_g \langle \alpha.P, Q\rangle_{s+t} \quad (t>0)$

(証明) 命題 4.3, 4.5 と同様 ∎

命題 4.13 (合同性) 時間的観測合同 $=_g$ は合同関係である．

(証明) 命題 4.6 と同様 ∎

最後に，三つの等価関係 $\sim_g$, $\approx_g$, $=_g$ の間の関係を示す．

命題 4.14 プロセス $P, Q \in \mathcal{P}$ において以下の性質が成立する．

(1) $P \sim_g Q$ ならば $P =_g Q$

(2) $P =_g Q$ ならば $P \approx_g Q$

(証明) 各等価関係の定義より明らかである． ∎

これより，時間的強等価で成立する等価関係，命題 4.3〜4.6 は時間的観測等価，時間的観測合同においても成立することになる．

5. 時間的強等価に基づく公理系

機械的な証明を可能にする方法として，符号 (=) に関する等価式から構成される公理系 $\mathcal{I}$ を**図 3** に示す．以下では，この公理系が時間的強等価 $\sim_g$ に対して健全かつ完全になっていることを示す．ただし，本節では議論の対象を有限プロセス*に限定する．

定理 5.1 (健全性) 有限プロセス $P, Q \in \mathcal{P}$ で $P=Q$ ならば $P \sim_g Q$ である．

(証明) 命題 4.3〜4.6 より等価プロセスは時間的強等価 ∎

これより公理系 $\mathcal{I}$ が健全であることがわかる．次に公理系 $\mathcal{I}$ が完全であることを示すため，RtCCS プロセスに関する標準形を定義する．

定義 5.2 (標準形) プロセス標準形 ($\mathcal{NF}$) は次のように帰納的に定義される式である．ただし，$\mathcal{I}$ は 0 を含む正整数の有限集合とする．

(1) $\mathcal{NF}_\Sigma$ または $\mathcal{NF}_\diamond$ は $\mathcal{NF}$ である．

(2) P_i が $\mathcal{NF}$ であれば，$\sum_{i\in\mathcal{I}} \alpha_i.P_i$ は $\mathcal{NF}_\Sigma$ である．

(3) P が $\mathcal{NF}_\Sigma$ で，かつ Q が $\mathcal{NF}$ であれば，$\langle P, Q\rangle_1$ は $\mathcal{NF}_\diamond$ である． ∎

補題 5.3 任意のプロセス P において，$P=P'$ となる標準形 ($\mathcal{NF}$) のプロセス P' が存在する．

(証明) 動作式は再帰的に定義されることにより，ある動作式が標準系に変形可能であると仮定したとき，その動作式に各演算子を適用して構成された動作式も標準系に変形可能であることを帰納的に示せばよい ∎

$$
\begin{array}{lcl@{\qquad}lcl}
P+Q & = & Q+P & P+(Q+R) & = & (P+Q)+R \\
P+P & = & P & P+0 & = & P \\
(P+Q)\backslash L & = & P\backslash L + Q\backslash L & (\alpha.P)\backslash L & = & \begin{cases} 0 & \alpha \in L\cup\bar{L} \text{ のとき} \\ \alpha.P\backslash L & \text{上記以外} \end{cases} \\
(P+Q)[f] & = & P[f]+Q[f] & (\alpha.P)[f] & = & f(\alpha).P[f] \\
\langle a.P, a.P\rangle_t & = & a.P & \langle \tau.P+Q, R\rangle_t & = & \tau.P+Q \\
\langle\langle P,Q\rangle_s, R\rangle_t & = & \langle P, R\rangle_t \quad (s\geq t) & \langle\langle P,Q\rangle_s, R\rangle_t & = & \langle P, \langle Q,R\rangle_{t-s}\rangle_s \quad (s<t) \\
\langle P, Q+R\rangle_t & = & \langle P,Q\rangle_t + \langle P,R\rangle_t & \langle P+Q, R\rangle_t & = & \langle P,R\rangle_t + \langle Q,R\rangle_t \\
P|Q & = & Q|P & P|(Q|R) & = & (P|Q)|R \\
P|0 & = & P & & & \\
P|Q & = & \sum_{\alpha_i=\beta_j} \tau.(P_i|Q_j) + \sum_{i\in I} \alpha_i.(P_i|Q) + \sum_{j\in J} \beta_j.(P|Q_j) & & & \\
 & & \text{ただし } P \stackrel{\text{def}}{=} \sum_{i\in I} \alpha_i.P_i \text{ かつ } Q \stackrel{\text{def}}{=} \sum_{j\in J} \beta_j.Q_j & & & \\
P|Q & = & \langle \sum_{\alpha_i=\beta_j} \tau.(P_i|Q_j) + \sum_{i\in I} \alpha_i.(P_i|Q) + \sum_{j\in J} \beta_j.(P|Q_j), P'|Q\rangle_1 & & & \\
 & & \text{ただし } P \stackrel{\text{def}}{=} \langle \sum_{i\in I} \alpha_i.P_i, P'\rangle_1 \text{ かつ } Q \stackrel{\text{def}}{=} \sum_{j\in J} \beta_j.Q_j & & & \\
P|Q & = & \langle \sum_{\alpha_i=\beta_j} \tau.(P_i|Q_j) + \sum_{i\in I} \alpha_i.(P_i|Q) + \sum_{j\in J} \beta_j.(P|Q_j), P'|Q'\rangle_1 & & & \\
 & & \text{ただし } P \stackrel{\text{def}}{=} \langle \sum_{i\in I} \alpha_i.P_i, P'\rangle_1 \text{ かつ } Q \stackrel{\text{def}}{=} \langle \sum_{j\in J} \beta_j.Q_j, Q'\rangle_1 & & &
\end{array}
$$

図 3 時間的強等価 $\sim_g$ に基づく公理系 $\mathcal{I}$
Fig. 3 An equational proof system based on $\sim_g$.

* 再帰式を含まず，かつ導出木が有限となるプロセス．

補題 5.4 二つのプロセス P と Q において以下の性質が成立する．

1. $P\sim_g Q$ かつ P が $\mathcal{IF}_\Sigma$ で Q が $\mathcal{IF}$ であるならば，Q は $\mathcal{IF}_\Sigma$ に変形可能
2. $P\sim_g Q$ かつ P が $\mathcal{IF}_\diamond$ で Q が $\mathcal{IF}$ であるならば，Q は $\mathcal{IF}_\diamond$ に変形可能

(証明) $\mathcal{IF}_\Sigma$ と $\mathcal{IF}_\diamond$ における $\vee$ 事象による遷移を比較すればよい．P_s を $\mathcal{IF}_\Sigma$ のプロセス，P_t を $\mathcal{IF}_\diamond$ のプロセスとする．ただし，$P_t\equiv\langle P_1, P_2\rangle_1$ かつ $P_1\not\sim_g P_2$ とする ($P_1\sim_g P_2$ の場合は P_t は $\mathcal{IF}_\Sigma$ に容易に変形可能)．P_s は $P_s\rightarrow P_s'\equiv P_s$，$P_t$ は $P_t\rightarrow P_t'\equiv P_2$ より，$\sim_g$ を満足するプロセスは共に $\mathcal{IF}_\Sigma$，または共に $\mathcal{IF}_\diamond$ となる． ∎

以上の結果から以下の定理が導き出される．

定理 5.5 (完全性) プロセス $P, Q\in\mathcal{P}$ で $P\sim_g Q$ ならば，$P=Q$ である．

(証明) 補題 5.3 と補題 5.4 より $P\sim_g Q$ となるプロセス P と Q は同一の標準形に変形可能であることより明らか． ∎

これより公理系 $\mathcal{IF}$ が完全であることがわかる．

6. 例　　題

RtCCS の記述と検証技法を示すため，線形結合された N 個のマルチプロセッサ間の通信の記述と解析を行う．

ここでは，まず隣接プロセッサ間の通信プロトコルを RtCCS の動作式で記述し，これをもとに N プロセッサ間の通信を表す動作式を求める．ただし，隣接プロセッサ間の結合回路のデータ転送時間を d とし，各プロセッサにおける処理時間を p (ただし $p<d$) とする．

隣接プロセッサ間の通信プロトコル

隣接した i と $i+1$ 番目のプロセッサの間の通信を**図 4** に示す．これは，通信路 (M) と，データ通信路に送る送信側 (S_i)，通信路からデータを受け取る受信側 (R_{i+1}) から構成され，次のようなプロトコルをもつ．

- 送信側 (S_i) はデータを受け取ると (s_i)，通信路 (M) にデータを送信し ($\overline{put}$)，受信通知 (ack) を待ち，時間 t (ただし $t>d$) が経過しても受信通知を受け取らないときは再送信する☆．

☆ 通信路に失敗がないことと $t>d$ より実際には再送信することはない．

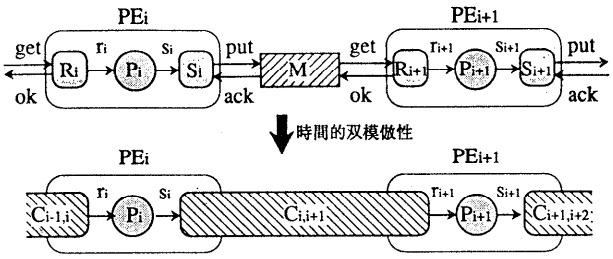


図 4 隣接プロセッサ間の通信
Fig. 4 A communication for two processors and its abstraction.

- 通信路 (M) はデータ転送に d 単位時間を要する．ただし，受信通知の転送時間は十分に小さいとする．
- 受信側 (R_{i+1}) は通信路 (M) からデータを受け取ると (get)，受信確認 ($\overline{ok}$) を送信して初期状態の戻り，同時にデータ ($\overline{r_{i+1}}$) を非同期送信する．

$$S_i\stackrel{def}{=}s_i.S_i'$$
$$S_i'\stackrel{def}{=}\overline{put}.\langle ack.S_i, S_i'\rangle_t$$
$$M\stackrel{def}{=}put.\langle \mathbf{0}, \overline{get}.M\rangle_d+ok.\overline{ack}.M$$
$$R_{i+1}\stackrel{def}{=}get.\overline{ok}.(R_{i+1}|\overline{r_{i+1}}.\mathbf{0})$$

R_{i+1} では，$\overline{ok}$ を実行したあと $\overline{r_{i+1}}$ の実行を待つことなく再び R_{i+1} となることにより，$\overline{r_{i+1}}$ の非同期送信を実現していることに注意されたい．

i と $i+1$ 番目間の隣接プロセッサ間の通信経路のプロトコル全体は，プロセス S_i, M, R_{i+1} の並列合成として表現される．これは，命題 4.3〜4.6 より，構造が簡単で動作の内容とタイミングが一致する動作式に変換できる．さらに，時間的観測等価により，内部遷移が捨象して外部的動作とその時間性だけを抽出したプロセス $C_{i,i+1}$ に変換することができる．

$$(S_i|M|R_{i+1})\backslash\{get, put, ack, ok\}\approx_g C_{i,i+1}$$
$$\text{ただし } C_{i,i+1}\stackrel{def}{=}s_i.\langle \mathbf{0}, C_{i,i+1}|\overline{r_{i+1}}.\mathbf{0}\rangle_d$$

ここで，$C_{i,i+1}$ をこのプロトコルの外部的な動作内容や時間性を表す仕様と考えれば，この等価関係はプロトコルの実現を表す $(S_i|M|R_{i+1})\backslash\{get, put, ack, ok\}$ と仕様 $C_{i,i+1}$ が一致し，実現は仕様を満足していることの検証に相当する．

線形接続された N プロセッサ間の通信

各プロセッサの処理時間が p であることから，i 番目のプロセッサは前述の受信側 (R_i) からデータを受け取り (r_i)，p 単位時間の実行後に送信側 (S_i) にデータを渡す (s_i) ことより，i 番目のプロセッサはつぎの P_i と記述される．

$P_i \stackrel{def}{=} r_i.\langle \mathbf{0}, \overline{s_i}\rangle_p$

そして，N個のプロセッサを線形接続したときの通信は，1からN番目のプロセッサを表す$P_1, \cdots P_N$を隣接通信経路を記述した$C_{1,2}, \cdots C_{N-1,N}$を介して並列合成したプロセス$P$として表現される．

$$P \stackrel{def}{=} (P_1 | C_{1,2} | P_2 | C_{2,3} \cdots P_{N-1} | C_{N-1,N} | P_N) \setminus \bigcup_{1 \le i < N} \{s_i\} \setminus \bigcup_{1 < i \le N} \{r_i\}$$

Pは時間的観測等価関係によりプロセスP'に等価変形することができる．

$P \approx_g P'$ ただし

$$P' \stackrel{def}{=} r_1.(\langle \mathbf{0}, P'\rangle_{d+p} | \langle \mathbf{0}, \overline{s_N}.\mathbf{0}\rangle_{(N-1)\times d + N \times p})$$

これより，Nプロセッサ間の通信の動作内容とその時間性をより簡単な構造をもつ等価なプロセスP'を用いて解析することができる．例えば，N経路間のデータ転送に要する時間が$(N-1)\times d + N \times p$であること，最前のプロセッサが次のデータを受理可能になるまでの時間が$d+p$であることなどがP'より容易にわかる．

7. おわりに

本論文では，プロセス計算の体系CCSに時刻印事象と時限演算子を導入した形式系であるRtCCSを構築した．これはCCSの有意義な特性を保持しながら，数量的時間と時間経過に依存した動作が表現可能で，並列計算に含まれるプロセス間通信とその時間性に関する解析の基礎となるものである．さらに，この形式系に基づく証明技法として，時間性を考慮したプロセスの等価関係である時間的強等価，時間的観測等価，時間的観測合同を与え，また，機械的な証明を可能にする方法として時間的強等価に基づく健全かつ完全な公理系を与えた．これにより時間的特性を考慮した並列計算の検証が可能になる．

今後の課題として，時間的観測合同に基づく公理系の構築や，RtCCSに基づく様相論理・時相論理体系の提案などがあげられる．

謝辞 有意義なコメントを頂いた査読者の方々，ならびに高汐一紀氏，小杉尚子氏，伊藤純一郎氏をはじめとする慶應大学所研究室の方々と，執筆にあたり励ましをして頂いた慶應大学の天野英晴講師に心から感謝いたします．

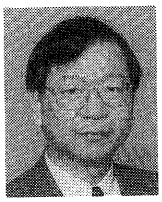
参考文献

1) Beaten, J. C. M. and Bergstra, J. A.: *Process Algebra*, Cambridge University Press (1990).
2) Hansson, H. and Jonsson, B.: A Calculus of Communicating Systems with Time and Probabilities, *Proc. IEEE 11th Real-Time Systems Symposium*, pp. 278-287 (1990).
3) Hennessy, M. and Regan, T.: A Temporal Process Algebra, Technical Report 2/90, University of Sussex (1990).
4) Hoare, C. A. R.: *Communicating Sequential Processes*, Prentice Hall (1985).
5) Jeffrey, A.: A Linear Time Process Algebra, *Proc. CAV '91*, LNCS, Vol. 575, pp. 432-441 (1991).
6) Milner, R.: *Communication and Concurrency*, Prentice Hall (1989).
7) Moller, F. and Tofts, C.: A Temporal Calculus of Communicating Systems, *Proc. CONCUR '90*, LNCS 458, pp. 401-415 (1990).
8) Moller, F. and Tofts, C.: Relative Processes with Respect of Speed, *Proc. CONCUR '91*, LNCS 527, pp. 424-438 (1991).
9) Nicollin, X. and Sifakis, J.: The Algebra of Timed Process ATP: Theory and Applications, IMAG Technical Report, RT-C 26 (Dec. 1990).
10) Nicollin, X. and Sifakis, J.: An Overview and Synthesis on Timed Process Algebras, *Proc. CAV '91*, LNCS, Vol. 575 (July 1991).
11) Park, D.: Concurrency and Automata on Infinite Sequences, *Proc. 5th GI*, LNCS 104 (1981).
12) 富樫 敦，二木厚吉：プロセス代数とその応用，*bit*, Vol. 24, No. 8, 共立出版，pp. 901-911 (1992).
13) Walker, D.: Bisimulation and Divergence, *Information and Computation 85*, pp. 202-241 (1990).
14) Yi, W.: CCS+Time=an Interleaving Model for Real Time Systems, *Proc. ICALP '91*, LNCS 510, pp. 217-228 (July 1991).

(平成4年9月10日受付)
(平成4年12月10日採録)

佐藤　一郎

1991 年慶應義塾大学理工学部電気工学科卒業．1993 年同大学大学院理工学研究科計算機科学専攻修士課程修了．現在，同大学大学院博士課程在学中．並行・分散計算モデル，オブジェクト指向計算，プログラミング言語処理系，プログラミング環境に興味を持っている．日本ソフトウェア科学会，ACM 各会員．



所　真理雄（正会員）

1970 年慶應義塾大学工学部電気工学科卒業．1975 年同大学博士課程修了．工学博士．同大学電気工学科助手，専任講師，助教授を経て現在教授．その間，1979 年ウォータールー大学訪問助教授，1980 年カーネギーメロン大学訪問助教授．1988 年よりソニーコンピュータサイエンス研究所副所長を兼務．計算モデル，プログラミング言語，分散・開放型システム，人工知能などに興味を持っている．主要著書に「計算システム入門」(岩波書店)，「Object Oriented Concurrent Programming」(MIT Press，編書) などがある．日本ソフトウェア科学会，電子情報通信学会，ACM，IEEE ほか各会員．