プロセス代数仕様の分割法の提案*

4M－3

郷 健太郎　白鳥 則郎†

東北大学電気通信研究所‡

1　はじめに

情報通信システムなどの大規模・複雑なシステムの開発法は一般に (1) 分割・詳細化による方法と (2) 部品合成による方法の2つに分けられる．筆者らはすでに (1) の方法に基づき，LOTOS[1] で記述されたシステムの仕様の系統的な分割法を提案した [2]．しかしこの分割法は，(a) 同一アクション名は高々一度しか記述されていない，(b) 再帰がない，(c) 内部アクションがないという制限を持ったプロセスのみを分割の対象としていた．本稿では，(b) の制限を除いたプロセスに対するより一般的な分割法を提案し，その正当性を示す．

2　準備

本論文で使用する言語は Basic LOTOS の一部であり，(1)stop, (2) アクションプレフィックス，(3) チョイス，(4) 並列オペレータ，(5) プロセスインスタンシエーションから構成される．この言語で記述されたプロセスの意味はラベル付き遷移システム (LTS)[1] によって与えられる．プロセス Bを構成するアクション集合を $Act(B)$ で表現する．

本論文では，プロセス上の同値関係として強等価 [3] を用い，これを記号$\sim$で表現する．

定義 1 (分割問題)[2]: 以下の2つの入力が与えられたとする．

- LTS の形状のプロセス B,
- $Act(B)$ を2分割したアクション集合 A_a と A_b, すなわち $A_a \cup A_b = Act(B)$ かつ $A_a \cap A_b = \emptyset$.

このとき，以下の性質を満たす2つのプロセス LC_a と LC_b, さらに付加プロセス RCを構成するのがここでの分割問題である．

- $Act(LC_a) = A_a$, $Act(LC_b) = A_b$かつ $Act(RC) \subseteq Act(B)$,
- $(LC_a \ |||\ LC_b)\ |[G]|\ RC \sim B$, ここで $RC := RC_1 \ |||\ RC_2 \ |||\ \cdots \ |||\ RC_n$ $(n \geq 1)$ かつ $G \subseteq Act(B)$.

定義 2 $\langle S, L, T, s_0 \rangle$ における S上の同値関係$\equiv$が縮退であるとは，次のときかつそのときに限る．

1. $s, s' \in S$, $s \equiv s' \Longrightarrow s \xrightarrow{a_1} s_1 \xrightarrow{a_2} \ldots \xrightarrow{a_n} s_n = s'$ or $s' \xrightarrow{a_1} s_1 \xrightarrow{a_2} \ldots \xrightarrow{a_n} s_n = s$ for some s_i,
2. $s_0 \xrightarrow{a_1} s_1 \xrightarrow{a_2} \ldots \xrightarrow{a_n} s_n, s_0 \equiv s_n \Longrightarrow s_0 \equiv s_1 \equiv \ldots \equiv s_n$.

定義 3 $B = \langle S', L, T', [s_0] \rangle$ を LTS とし，$\equiv$を縮退同値関係とする．$\equiv$によって分割された LTS $B/\equiv$ は，LTS $\langle \{[s] \mid s \in S\}, L, T', [s_0] \rangle$ である．ここで，$[s_i] \xrightarrow{a} [s_j] \in T' \Longleftrightarrow \exists s_i' \in [s_i], s_j' \in [s_j], s_i' \xrightarrow{a} s_j'$.

本論文では，同値関係と分割を同一とみなす．ゆえに，分割πは縮退であり，πに関して Bから LTS B/πを構成することが可能である．

3　プロセス代数仕様の分割法

分割アルゴリズムを構成するために，3つの副アルゴリズムを準備する．これらは，プロセスにおいてアクション集合 A に属さないアクションによる遷移を縮退させるために用いられる．

副アルゴリズム 1

Input: (1) $B = \langle S, L, T, s_0 \rangle$, (2) $A \subseteq L$.

Output: Sにおける分割 (S上の同値関係に対応する).

$\pi := \{[s] \mid s \in S\}$;
for each transition $s \xrightarrow{a} s' \in T$
　if $a \notin A$ then
　　$\pi := (\pi - \{[s], [s']\}) \cup \{[s] \cup [s']\}$;
return π.

副アルゴリズム 2 では，各アクション集合を互いに共通要素を持たないようにするため，副アルゴリズム 3 を使用する．

副アルゴリズム 2

Input: (1) $B = \langle S, L, T, s_0 \rangle$, (2) A_a, A_b, where $A_a \cup A_b = L, A_a \cap A_b = \emptyset$.

Output: $\mathcal{A} = \{A_i \mid i = 1, 2, \ldots, n\}$.

$n := 1$;
$A_1 := Init(s_0)$;
$\mathcal{A} := \{A_1\}$;
for each transition $s_i \xrightarrow{a} s_j \in T$
　begin
　　$n := n + 1$;

---

*A Proposal on a Decomposition Method of Process Specifications

†Kentaro GO and Norio SHIRATORI

‡Research Institute of Electrical Communication, Tohoku University

$A_n := \{a\} \cup Init(s_j)$;
$\mathcal{A} := \mathcal{A} \cup \{A_n\}$;
end
for each $A_i \in \mathcal{A}$
if $A_i \subset A_a$ or $A_i \subset A_b$ **then**
$\mathcal{A} := \mathcal{A} - \{A_i\}$;
$\mathcal{A}$:= 副アルゴリズム 3 ($\mathcal{A}$);
return $\mathcal{A}$.

副アルゴリズム 3
Input: $\mathcal{A} = \{A_i \mid i = 1, 2, ..., n\}$,
Output: $\mathcal{A}$.

while $\exists A_i, A_j \in \mathcal{A}\ (i \neq j), A_i \cap A_j \neq \emptyset$ **do**
$\mathcal{A} := (\mathcal{A} - \{A_i, A_j\}) \cup \{A_i \cup A_j\}$;
return $\mathcal{A}$.

アルゴリズム 1
Input: (1) B, (2) A_a, A_b, where $A_a \cup A_b = L, A_a \cap A_b = \emptyset$.
Output: (1) LC_a, LC_b, where $Act(LC_a) = A_a, Act(LC_b) = A_b$. (2) $RC_i, i = 1, 2, ..., n$, where each $Act(RC_i)$ is pairwise disjoint.
Step 1 : 副アルゴリズム 2 に B, A_a, A_bを入力し, $A_1, A_2, ..., A_n$を作る.
Step 2 (LC_a, LC_b, RCの構成):
各$\alpha = a, b$ に対して, 副アルゴリズム 1 に B, A_αを入力し, π_αを得る. その後 $LC_\alpha = \langle S/\pi_\alpha, A_\alpha, T_\alpha, [s_0]_\alpha \rangle$ を構成する.
各 $i = 1, 2, ..., n$ に対して, 副アルゴリズム 1 に B, A_iを入力し, π_iを得る. その後 $RC_i = \langle S/\pi_i, A_i, T_i, [s_0]_i \rangle$ を構成する.

アルゴリズム 1 により構成されたプロセスは以下の性質を持つ.

補題 1 RCを構成する各アクション集合 $Act(RC_1), Act(RC_2), ..., Act(RC_n)$ は互いに素である.

以下では, $B' = (LC_a \ |||\ LC_b)\ |[G]|\ (RC_1 \ |||\ ... \ |||\ RC_n)$, $G = \bigcup_i Act(RC_i)$ とし, B'における任意の状態を $n+2$ 項組 $\langle [s_a]_a, [s_b]_b, [s_1]_1, ..., [s_n]_n \rangle$ で表現する. ここで (1) $s_a, s_b, s_1, ..., s_n$ は Bの状態; (2) $[s_a]_a, [s_b]_b, [s_i]_i$ はそれぞれπ_a, π_b, π_iに関する同値類; (3) π_a, π_b, π_iはそれぞれ Bから LC_a, LC_b, RC_iを構成するために使われる分割である.

補題 2 s を Bの状態とし, B'においてある$\tilde{s}$が存在し, $\langle [s]_a, [s]_b, [s]_1, ..., [s]_n \rangle \xrightarrow{a} \tilde{s}$を仮定する. このとき, Bにある s'が存在して, $\tilde{s} = \langle [s']_a, [s']_b, [s']_1, ..., [s']_n \rangle$ すなわち $s \xrightarrow{a} s'$.

補題 3 s と s'を Bの状態とする.
Bにおいて $s \xrightarrow{a} s' \iff B'$において
$\langle [s]_a, [s]_b, [s]_1, ..., [s]_n \rangle \xrightarrow{a} \langle [s']_a, [s']_b, [s']_1, ..., [s']_n \rangle$.

定理 1 Bを (1) 内部アクション i がない; (2) 各アクションは高々1 度しか表われないという制限を持つプロセスとする. A_aと A_bを $Act(B)$ を 2 分割した集合とする. すなわち, $A_a \cup A_b = Act(B)$ かつ $A_a \cap A_b = \emptyset$. LC_a, LC_b, RC_i $(i = 1, 2, ..., n)$ をアルゴリズム 1 に B, A_a, A_bを入力することによって構成されたプロセスとする. このとき, $B \sim (LC_a \ |||\ LC_b)\ |[G]|\ (RC_1 \ |||\ ... \ |||\ RC_n)$ が成立する.

例 1 $B = a;(b;B \ []\ c;(d;f;B \ []\ e;g;C))$, $C = h;j;(k;m;C \ []\ l;n;B)$ を仮定し, さらに $Act(B) = \{a, b, c, d, e, f, g, h, j, k, l, m, n\}$ を $A_a = \{a, b, f, g, h, m, n\}$ と $A_b = \{c, d, e, j, k, l\}$ に分割すると仮定する. アルゴリズム 1 に B, A_a, A_bを入力すると, 以下のプロセスが得られる. $LC_a = a;(b;LC_a \ []\ f;LC_a \ []\ g;LC'_a), LC'_a = h;(m;LC'_a \ []\ n;LC_a)$, $LC_b = c;(d;LC_b \ []\ e;LC'_b), LC'_b = j;(k;LC'_b \ []\ l;LC_b)$, $RC_1 = a;(b;RC_1 \ []\ c;RC_1)$, $RC_2 = d;f;RC_2$, $RC_3 = e;g;RC_3$, $RC_4 = h;j;RC_4$, $RC_5 = k;m;RC_5$, $RC_6 = l;n;RC_6$. 定理 1 より, $B \sim (LC_a \ |||\ LC_b)\ ||\ (RC_1 \ |||\ ... \ |||\ RC_6)$ が成立する.

4 おわりに

本論文では, 再帰を含むプロセス代数仕様の分割法について述べた. 本分割法の制限は, (1) 同一アクション名は高々一度しか記述されていない, (2) 内部アクションがない仕様を分割できないことである. 今後の課題として, (a) 上記の制限をすべて取り除くこと, (b) 構造を持った仕様の分割法の構成が挙げられる.

参考文献

[1] ISO, "Information Processing Systems — Open Systems Interconnection — LOTOS — A formal description technique based on the temporal ordering of observational behaviour," ISO 8807, 1989.

[2] K. Go and N. Shiratori, "Modularization of a Specification in LOTOS," in *Proc. the International Conference on Network Protocols*, pp. 55-62, 1993.

[3] R. Milner, *Communication and Concurrency*, Prentice-Hall, 1989.