2K－2

# 動的再構成システム仕様の挙動検証

加納　稔久　高橋　薫　加藤　靖
仙台電波工業高等専門学校

## 1　まえがき

分散並行システムにおいては，リンクを経由してシステム構成要素間で通信が行われ，所望の分散アプリケーションが達成される．リンクが静的ではなく動的に構成・確立されるのであれば，システムは動的再構成システムとして複雑さを呈することになる．本稿では，システム構成要素を有限状態機械の概念でモデル化し，それらが動的にリンクを確立・解消しながら並行実行するような動的再構成システムの形式的仕様化と仕様の挙動検証を可能とする方法について述べる．

## 2　システムモデル

動的再構成システムを，互いにリンクを形成し通信しあういくつかの有限状態機械の集まりとしてモデル化する．

**定義 1** 動的再構成システム $Sys$ を

$$Sys = \langle CFSM_1, \cdots, CFSM_k, \cdots, CFSM_n \rangle$$

で定義する．ここで，$CFSM_k(1 \leq k \leq n)$ は通信型有限状態機械であり次のように定義する：

$CFSM_k = \langle Q_k, E_k, \delta_k, q_{k0} \rangle$
$Q_k$ : $CFSM_k$の状態の有限集合
$E_k$ : $CFSM_k$のイベントの集合
$E_k = IP_k \times (\Sigma_k \cup \Delta_k \cup C_k)$
$IP_k$ 作用点の集合　$IP_k = \{1, \cdots, n\} - \{k\}$
$\Sigma_k$ 入力イベントの集合
$\Delta_k$ 出力イベントの集合
$C_k$ 制御イベントの集合
$\delta_k : E_k \times Q_k \rightarrow Q_k$ : $CFSM_k$の状態遷移関数
$q_{k0} \in Q_k$ : $CFSM_k$の初期状態

制御イベントには *connect* (通信リンクの確立) と *disconnect* (通信リンクの切断) がある．入力イベント (メッセージ，作用の受信) または出力イベント (メッセージ，作用の送信) は一対の $CFSM$ 間にリンクが確立して始めて実行可能になる．

## 3　システムの挙動と検証

### 3.1　システムの挙動

システムの挙動の特性化は，文献[1]による．すなわち，(a) 個々の $CFSM$ の状態，(b) $CFSM$ 対間のリンク端に仮定されたイベントキューの内容の組で表現されるシステム状態を定義し，システム状態の遷移系列として挙動を定義している．

システム状態遷移系列は一つのグラフ構造を形成し，そのグラフ (システム状態グラフ) がシステムの挙動全体を表現する．

**定義 2** 動的再構成システム $Sys$ のシステム状態 $s$ を以下のように定義する：

$$s = \langle (q_1, \cdots, q_k, \cdots, q_n), (c_{1,2}, \cdots, c_{i,j}, \cdots, c_{n,n-1}) \rangle$$

ここで，$q_k(1 \leq k \leq n)$ は $CFSM_k$ の状態である．$c_{i,j}(1 \leq i,j \leq n, i \neq j)$ は $CFSM_i$ と $CFSM_j$ 間のリンクの $CFSM_j$ 側のイベントキューの内容である．

**定義 3** システム状態グラフ $G$ は次の4項組である：

$$G = \langle S, E, \delta, s_0 \rangle$$

ここで，$S$はシステム状態の集合，$E$はイベントの集合，$\delta$ $(S \times E \rightarrow S)$ はシステム状態の遷移関数，$s_0$は初期システム状態である．

### 3.2　挙動の検証

#### 3.2.1　検証項目

システム $Sys$ に対応するシステム状態グラフ $G$ を用いることにより，$G$ の生成過程で以下の状態を検証できる：

---

Validation of the Behavior of a Dynamically Reconfigurable System
Toshihisa Kano, Kaoru Takahashi and Yasushi Kato
Sendai National College of Technology
Kitahara 1, Kamiayashi, Aoba-ku, Sendai, 989-31, Japan

- デッドロック状態
  遷移先がなくキュー内容が空である状態
- チャネルオーバフロー状態
  イベント送信によりチャネルの容量を越えてしまった状態
- 未指定受信状態
  遷移先がなくイベント受信において対応するキューの先頭にそのイベントが存在しない状態

システム状態グラフ $G$ が上記の3種類を含まないならば，システム $Sys$ は安全であるという．

さらにリンク関連では，以下に示すエラーをも検出できる:

- 未リンク送信状態
  $CFSM_i$ と $CFSM_j$ の間にリンクが確立されていないにもかかわらず，$CFSM_i$ が $CFSM_j$ にイベントを送信可能である状態
- 未リンク $CFSM$ 対
  リンクが決して確立されることのない $CFSM$ 対
- 重複確立状態
  リンクが確立されているにもかかわらず，再び確立をしようとした状態
- 重複切断状態
  リンクが切断されているにもかかわらず，再び切断をしようとした状態

### 3.2.2 検証アルゴリズム

ここでは，システム状態グラフの生成を基本とした具体的な検証法の概要を与える．

[検証アルゴリズム]
入力 1. システム仕様 $Sys$
2. チャネル容量 $C(C \geq 1)$

**begin**
*initialize* (* 初期化 *)
$S_p := S_p \cup \{s_0\}$ (* $s_0$は初期システム状態 *)
**while** $(S_p \neq \phi)$ **do**
**begin**
$S_p$の任意の要素 $s$ を選び以下の手続きの実行
$normal(s)$ (* システム状態グラフの生成 *)
$unsafe(s)$ (* 安全性の検証 *)
$link(s)$ (* リンク関連エラーの検証 *)
$S_p := S_p - \{s\}$
$S := S \cup \{s\}$
**end**
*decision*1 (* 安全性に対するエラーの判定 *)
*decision*2 (*リンクに対するエラーの判定 *)
**end**

$S_p$はこれから検証を行うシステム状態の集合であり，$S$は検証を終了したシステム状態の集合である．$S_p$にはエラーが検出されない限り次システム状態が加えられる．エラーが検出された場合はシステム状態 $s$ 以降の検証を行わない．

この検証アルゴリズムは

1. システム状態グラフの生成 (*normal*)
2. 安全性の検証 (*unsafe*)
   デッドロック状態，チャネルオーバフロー状態，未指定受信状態の検証
3. リンク関連エラーの検証 (*link*)
   未リンク送信状態，未リンク $CFSM$ 対，重複確立 (切断) 状態の検証

を $S_p = \phi$(全ての $s$ の検証が終了) になるまで繰り返し行う．

チャネル容量を制限しているため，システム状態グラフの生成は無限には行われず，このアルゴリズムは有限ステップで停止する．

## 4 おわりに

本稿では，動的再構成システム仕様の挙動に関する検証アルゴリズムを提示した．今後は，他の種類のエラーにも対処できるようにアルゴリズムを拡張し，システムの実装を行う．

## 参考文献

[1] 佐藤，関，渡部，高橋：“動的再構成システムの仕様化とそのシミュレータ，”信学技報 SSE96-72 (1996).