

# ECFSM モデルの通信プロトコルに 故障耐性機能を付加する一手法

片倉 健一 樋口 昌宏 藤井 護

大阪大学基礎工学部

e-mail: katakura@ics.es.osaka-u.ac.jp

通信システムの高信頼化のためには、フォールトを考慮した設計が重要である。本稿では、フォールトが発生しないという前提で安全性が検証されている 2-ECFSM モデルの通信プロトコルについて、self-stabilizing となるように拡張する一手法を提案する。提案手法ではデッドロック状態と未定義受信状態から回復するためのアクションを付加する。回復先の状態を現在の状態に最も近い安全な状態とし、そのためにプロトコルの状態間の距離を定義し、距離について成立するいくつかの性質について議論した。

## A Method for Constructing Self-Stabilizing Communication Protocols modeled as 2-ECFSMs

Kenichi Katakura Masahiro Higuchi Mamoru Fujii

Faculty of Engineering Science,  
Osaka University

e-mail: {katakura, higuchi, fujii}@ics.es.osaka-u.ac.jp

For implementing reliable communication systems, it is important to design communication protocols considering faults. In this paper, we propose a method in which self-stabilizing protocols are constructed based on communication protocols which are shown to be safe on the assumption that no faults occur in 2-ECFSM model. In proposing method, protocols are extended to be able to recover from every deadlock state or unspecified reception state to a safe state by adding some action. As such a safe state, a state is chosen so that it is the closest to current state among all safe states. For such a purpose, we define a distance between two states in a protocol, and discuss some properties that hold on the notion.

### 1 まえがき

通信システムの高信頼化のためにはフォールトを考慮したシステム設計が重要である。例えば、フォールトを考えないならばデッドロックに陥ることがないことが保証されている通信プロトコルにおいても、フォールトを原因として本来到達可能でない状態に遷移し、デッドロックに陥ることがある。そのような場合に対処するため、フォールトを原因として通信システムがシステム本来の機能が実行できない状態になったときに、システム自身によりシステム本来の機能が実行できる状態になることが望ましい。このような機能のことを

self-stabilizing 機能と呼ぶ。

self-stabilizing 機能を持つ通信プロトコルに関する数学的性質が Gouda らにより提案されている [1] [2] [3]。しかし、大規模な通信システムの構築の際に最初から self-stabilizing 機能を持つ通信プロトコルを設計することは困難である。そこで、既存の通信プロトコルを self-stabilizing 機能を含むように拡張する手法を確立することは有効な方法であると思われる。

拡張有限状態機械 (ECFSM) は有限状態機械 (CFSM) に複数のレジスタを付け加えたもので、多くの通信プロトコ

ルをモデル化することができる。本稿では、フォールトが発生しない時には安全性が保証されている ECFSM モデルの通信プロトコルを、self-stabilizing 機能を含むようなプロトコルに拡張する一手法を提案する。通信プロトコルの安全性とは、通信システムがデッドロック状態のようなシステム本来の機能が実行できない状態にならないことであり、ECFSM モデルの通信プロトコルの安全性を不変式を用いて検証する手法が極口らにより提案されている<sup>[4]</sup>。ここで、不変式とはシステムの初期状態から到達可能な状態で成立する論理式のことである。

本稿で提案する拡張方法は 極口らの検証法<sup>[4]</sup>により安全性が示されているプロトコルを対象とし、デッドロックに陥った状態からの回復の契機にはタイムアウトを用いる。

以降、2 では self-stabilizing 機能を付加する手法を提案するための準備としてプロトコルモデル、検証に用いられる不変式の形式について述べ、3 では提案手法を述べる。

## 2 準備

### 2.1 プロトコルモデル

通常、通信プロトコルは2つのプロトコル機械とそれらを接続する双方向の通信路でモデル化される。本報告ではプロトコル機械を拡張有限状態機械 (ECFSM) の部分クラスであるカウンタ付有限状態機械 (FSM-C) で、また、通信路を双方向無限 FIFO でモデル化する。

**定義 1** プロトコル機械  $PM$  は、以下のような四宇組  $(S, \Sigma, T, SI)$  で定義される。

- $S = \langle SF, r \rangle$  は状態の集合を定義する。  $SF$  は有限制御部の状態の有限集合、  $r$  は整数値レジスタの数であり、整数値レジスタは代入、整数数の加減算および大小比較の機能を持つ。また、  $s \in SF \times \mathcal{N}^r$  を  $PM$  の状態と呼ぶ。ここで、  $\mathcal{N}$  は非負整数集合である。
- $\Sigma = \Sigma_1 \cup \Sigma_2$  はメッセージ型を定義する。  $\Sigma_1, \Sigma_2$  はそれぞれ  $PM$  が送ることのできるメッセージ型の集合、  $PM$  が受けとることのできるメッセージ型の集合であり、互いに素である。メッセージとはメッセージ型  $d \in \Sigma$  と  $N_d$  個のパラメータ  $n_1, \dots, n_{N_d} \in \mathcal{N}$  の組であり、  $N_d$  の値は  $d$  により一意に定まる。
- $T$  はアクションの集合である。アクションは五宇組  $(sf, d, sf', C, R)$  で表され、  $sf, sf' \in SF, d \in \Sigma$  である。  $x, y$  はレジスタあるいはメッセージのパラメータであり、  $c$  は整数数であるとする。  $C$  は連立差分不等式であり、差分不等式は  $x - y \leq c$  の形である。また、  $R$  はレジスタ代入式の集合であり、レジスタに代入できる値は  $x + c$  の形で与えられる。  $PM$  の状態が  $s$  であり、  $t = (sf_t, d_t, sf'_t, C_t, R_t)$  がアクションである時、  $s$  の有限制御部の状態が  $sf'_t$  であり、  $s$  の各レジスタの値が  $C_t$  のすべての差分不等式を満たし、  $d_t \in \Sigma_1$  ならば、  $C_t$  のすべての差分不等式を満たすようなパラメータの値についてメッセージ  $\langle d_t, n_1, \dots, n_{N_{d_t}} \rangle$  を送り、有限制御部の状態を  $sf'_t$  とし、  $R_t$  に従ってレジスタの値を更新す

る。また、  $s$  の有限制御部の状態が  $sf_t$  であり、  $PM$  へのメッセージが  $\langle d_t, n_1, \dots, n_{N_{d_t}} \rangle$  (ただし、  $d_t \in \Sigma_2$ ) であり、  $s$  のレジスタの値とメッセージのパラメータの値が  $C_t$  のすべての差分不等式を満たすならば、メッセージ  $\langle d_t, n_1, \dots, n_{N_{d_t}} \rangle$  を受けとり、有限制御部の状態を  $sf'_t$  とし、  $R_t$  に従ってレジスタの値を更新する。

- $SI$  は  $SF \times \mathcal{N}^r$  の部分集合で、初期状態集合を表す。 □

**定義 2** 2つのプロトコル機械  $P, Q$  があり、  $\Sigma_P = \Sigma_Q$  かつ  $\Sigma_P = \Sigma_Q$  のとき、  $\Pi = (P, Q)$  をプロトコルと呼ぶ。また、  $P$  の状態  $p$ 、  $Q$  の状態  $q$ 、  $P$  から  $Q$  への通信路の状態 (メッセージ系列)  $u$ 、  $Q$  から  $P$  への通信路の状態  $v$  の四宇組  $(p, q, u, v)$  を  $\Pi$  の系の状態と呼ぶ。また、ある系の状態  $gs$  において、一方のプロトコル機械がメッセージを送る、またはメッセージを受けとるような一回のアクションを実行した後、系の状態が  $gs'$  となると、  $gs$  から  $gs'$  への遷移を系の状態遷移といい、  $gs \vdash gs'$  と表す。関係  $\vdash$  の反射推移閉包を  $\vdash^*$  で表し、  $gs \vdash^* gs'$  であるとき  $gs$  から  $gs'$  へ到達可能であるという。また、初期状態から到達可能な状態を単に到達可能という。 □

ECFSM モデルの例プロトコルを、有限制御部の状態をノード、アクションを枝とした有向グラフで表したものが図 1 である。ここで、初期状態における有限制御部の状態がそれぞれ  $sp_1, sq_1$  であり、レジスタの値は  $VH(P) = VH(Q) = \text{Max}$ 、  $VL(P) = VL(Q) = \text{Min}$  である。状態  $sp_1$  から  $sp_2$  へのアクションは遷移条件が  $Sn - VH(P) \leq 0, VL(P) - Sn \leq 0$  なので、  $P$  の有限制御部の状態が  $sp_1$  のとき  $VL(P) \leq Sn \leq VH(P)$  であるようなメッセージ  $\langle \text{Sup}, Sn \rangle$  を  $Q$  に送り、レジスタ  $VH(P)$  にパラメータ  $Sn$  の値を代入し、状態を  $sp_2$  にすることを表す。有限制御部の状態  $sq_1$  から  $sq_2$  へのアクションは遷移条件が  $\text{true}$  なので、  $Q$  の有限制御部の状態が  $sq_1$  であり、かつ  $P$  から  $Q$  への通信路の先頭のメッセージが  $\langle \text{Sup}, Sn \rangle$  ならば、そのメッセージを受けとり、レジスタ  $VH(Q)$  にパラメータ  $Sn$  の値を代入し、状態を  $sq_2$  にすることを表す。

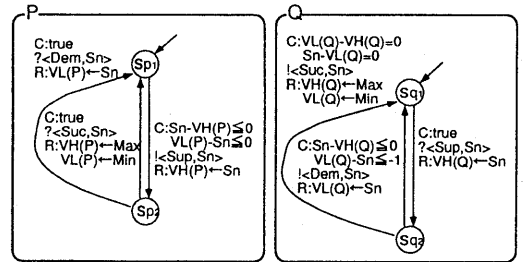


図 1: プロトコルモデルの例

**定義 3** プロトコル  $\Pi$  の到達可能な状態に以下のような状態が含まれていないとき、  $\Pi$  は安全であるという。

デッドロック状態 2つのプロトコル機械を接続する通信路が両方も空であり、2つのプロトコル機械の状態がともに実行可能な送信アクションがない状態

未定義受信状態 一方のプロトコル機械のある状態が、そのプロトコル機械への通信路の先頭のメッセージを受けとるアクションが定義されていない状態 □

## 2.2 不変式

プロトコルIIの系の状態に関する論理式  $F$ が、初期状態からの到達可能なすべての系の状態で成立するとき、論理式  $F$ をIIの不変式と呼ぶ。

樋口らの検証法<sup>[9]</sup>においてプロトコルIIの安全性を検証するために用いられる不変式は以下の原子式を用いて積和形の論理式として記述される。

AF1 2つのプロトコル機械  $P, Q$ の有限制御部の状態の対を指定する。

AF2 2つの通信路のメッセージ系列のメッセージ型の系列を正規表現を用いて表す。例えば、 $\langle \epsilon, MIA^* \rangle$ は  $P$ から  $Q$ への通信路が空であり、 $Q$ から  $P$ への通信路がメッセージ型  $MIA$ のメッセージが0個以上ある系列であることを表す。

AF3 2つの通信路のメッセージ系列のパラメータの状態を定義述語で表す。例えば、 $P$ から  $Q$ への通信路のメッセージ系列で、メッセージ型が  $MIA$ であり、かつそれのみを順に抽出した系列のパラメータは  $step1$ という定義述語を満たすことを  $step1(subseq(PQ, \{MIA\}))$ と表す。

AF4 プロトコルのレジスタと通信路上のメッセージのパラメータの関係を表す式。ただし、関係式は差分不等式で表されるとする。

本報告では、2つの通信路が空であるような系の状態を空チャンネル状態として、一般性を失うことなく不変式  $F$ が次の式で表されるとする。

$$F = R_1 \vee \dots \vee R_{N_c} \vee R_{N_c+1} \vee \dots \vee R_{N_F}$$

ここで、 $1 \leq i \leq N_c$ について、積項  $R_i$ が空チャンネル状態でのみ成立し、 $N_c + 1 \leq i \leq N_F$ について、積項  $R_i$ は空チャンネル状態では成立しないとす。空チャンネル状態で成立する積項  $R_i$ においては、原子式 AF2, AF3 の記述はなく、原子式 AF4 はレジスタのみの関係式となる。

提案されている検証法<sup>[4]</sup>では、 $F$ が不変式であること、 $F$ がいかなる安全でない状態でも成立しないことを検証システムを用いて示すことにより与えられたプロトコルの安全性を示す。

## 3 Self-Stabilizing 機能の付加

通信システムにフォールトが発生すると、そのことを原因としてシステム本来の機能を実行できない状態に遷移することがある。このような状態からシステム本来の機能を実行で

きる状態にシステム自身で回復する機能を self-stabilizing 機能と呼ぶ。

本稿では、フォールトを考慮しない場合、樋口らの手法<sup>[4]</sup>により安全性が保証されている FSM-C モデルの通信プロトコルに self-stabilizing 機能を付加する一手法を議論する。

ここではプロトコルで定義されたアクションは全て本来の機能を実行するものと考え、そのようなアクションを実行できない状態であるデッドロック状態、未定義受信状態から安全な状態への回復を考える。

### 3.1 デッドロックからの回復機能の付加

デッドロック状態は、空チャンネル状態であり実行可能な送信アクションが存在しないので、この状態で実行可能となる特別なアクションが必要となる。プロトコル機械に一定時間状態遷移がないときに状態遷移を行なう契機を与える事象をタイムアウトと呼ぶ。本報告で提案する手法では、デッドロック状態からタイムアウトにより状態を遷移し、さらにシステム本来の動作をする状態として安全性を保証する不変式  $F$ の成立する状態に遷移することにより回復をはかる。

提案手法では、一方のプロトコル機械のみがタイムアウトを行なうことにし、図2のように、デッドロック状態からの回復のための新しい状態とアクションを付加する。ここで、 $s_p, s_q$ がデッドロック状態におけるそれぞれのプロトコル機械の有限制御部の状態、 $s_{p'}, s_{q'}$ が回復後の有限制御部の状態であり、それ以外の状態と、アクションが新しく付加されるものである。

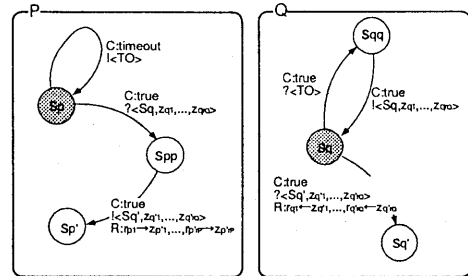


図2: デッドロックからの回復のための状態遷移

$P$ がタイマを持つプロトコル機械であり、遷移条件が  $timeout$  となっているものがタイムアウトによるアクションを表す。図2におけるデッドロックからの回復は次のように行なわれる。

1.  $P$ でタイムアウトが発生し  $Q$ へタイムアウトを知らせるメッセージ  $\langle TO \rangle$ を送る。
2.  $Q$ でメッセージ  $\langle TO \rangle$ を受けると、 $P \rightarrow Q$ の状態を表すメッセージ  $\langle S_q, z_{q1}, \dots, z_{qn} \rangle$ を送る。ここで、 $z_{q1}, \dots, z_{qn}$ は  $Q$ の各レジスタの値である。

3.  $P$ でメッセージ $\langle S_{q_1, z_{q_1}}, \dots, z_{q_{r_Q}} \rangle$ を受けると、回復先の系の状態 $\langle p', q', \epsilon, \epsilon \rangle$ を以下で述べる方法により決定し、 $PM_P$ は自分の状態を $p'$ に遷移させて、遷移すべき $Q$ の状態 $q'$ を指定したメッセージ $\langle S_{q_1, z_{q_1}}, \dots, z_{q_{r_Q}} \rangle$ を $Q$ へ送る。
4.  $Q$ でメッセージ $\langle S_{q_1, z_{q_1}}, \dots, z_{q_{r_Q}} \rangle$ を受けると、 $\langle S_{q_1, z_{q_1}}, \dots, z_{q_{r_Q}} \rangle$ で指定される $Q$ の状態 $q'$ に遷移する。

デッドロック状態においては2つの通信路は空であるので、回復先の系の状態として不変式 $F$ の中の空チャネルの系の状態で成立する積項 $R_i$ を満たす状態を考える。ここで、どの積項 $R_i$  ( $1 \leq i \leq N_c$ )を満たす状態にするかを決定するために、系の状態 $s$ と積項 $R_i$ との距離 $D(s, R_i)$ を次のように定義し、積項 $R_i$  ( $1 \leq i \leq N_c$ )の中で、デッドロック状態 $s$ との距離 $D(s, R_i)$ の値が最小となるような積項 $R_{min}(s)$ を満たす状態をデッドロック状態 $s$ からの回復先とする。

**定義 4** ある空チャネル状態 (デッドロック状態)  $s$  と不変式  $F$  中の空チャネルの系の状態で成立する積項  $R_i$  との距離  $D(s, R_i)$  を次に定義する距離  $D_s(s, R_i)$ ,  $Dr(s, R_i)$  の和とする。

**距離  $D_s(s, R_i)$ :**  $s$  における2つのプロトコル機械の有限制御部の状態と、積項  $R_i$  の AF1 型原子式の状態対のそれぞれの状態がいくつ異なるかを表す。

**距離  $Dr(s, R_i)$ :**  $s$  において積項  $R_i$  のすべての AF4 型原子式を満たすため値を変更すべきプロトコル機械のレジスタの数の最小値を表す。 □

距離  $D_s(s, R_i)$  は定義より簡単に値を決定できる。そこで、以下で距離  $Dr(s, R_i)$  の値を決定する方法を議論する。

積項  $R_i$  の AF4 型原子式の集合を  $C$  とし、集合  $C$  に  $x_i - x_j \leq c_1$ ,  $x_i - x_j \leq c'$  ( $c \neq c'$ ) が同時に存在することはなく、また集合  $C$  に  $y_1 - x \leq c_1$ ,  $x - y_2 \leq c_2$  が存在し  $y_1 - y_2 \leq c'$  ( $c' \leq c_1 + c_2$ ) が存在しなければ集合  $C$  に  $y_1 - y_2 \leq c_1 + c_2$  を加えるとする。 $s = (p, q, \epsilon, \epsilon)$  とし、集合  $C$  の中で状態  $s$  で成立しない差分不等式の集合を  $C_u$  とするとき、 $Dr(s, R_i)$  について以下の性質が成立する。

**定理 1**  $C_u$  が空集合の時は、 $Dr(s, R_i) = 0$ 。 □

**定理 2** あるレジスタ  $x$  について次の2つの条件が成立するとき、かつそのときのみ、 $Dr(s, R_i) = 1$ 。

- 集合  $C_u$  のすべての差分不等式に  $x$  が現れる。
- 集合  $C_u$  の差分不等式の  $x$  の係数がすべて同じである。(すなわち、すべて1か、すべて-1である。) □

**定理 3** あるレジスタ  $x, y$  について、次の条件が成立するときは、 $Dr(s, R_i) = 2$ 。

- 集合  $C_u$  のすべての差分不等式に  $x$  または、 $y$  が現れる。

- 集合  $C_u$  の  $y$  が現れないすべての差分不等式の  $x$  の係数がすべて同じであり、 $x$  が現れないすべての差分不等式の  $y$  の係数がすべて同じであり、 $x, y$  の両方が現れる差分不等式があれば、 $x$  の係数が  $y$  の現れない差分不等式の  $x$  の係数と同じであるか、 $y$  の係数が  $x$  の現れない差分不等式の  $y$  の係数と同じである。 □

定理 1 は自明。定理 2, 3 の証明は 3.2 で与える。

ここで、以上3つの性質を満たさないときは  $Dr(s, R_i)$  は2以上ということになるので、本稿ではこのとき便宜上  $Dr(s, R_i) = \infty$  であるとし、このような積項  $R_i$  が  $R_{min}(s)$  とならないようにする。このため、 $R_{min}(s)$  を求めることができなかつた場合、初期状態に回復することにする。

以上で議論したデッドロック状態からの回復機能をプロトコルに付加するためには、あらかじめすべてのデッドロック状態を列挙し、それらの状態からの回復のための有限制御部の状態とアクションを付加することが必要となる。

以下では、 $(s_p, s_q)$  を次の条件を満たす有限制御部の状態とする。

- 2つのプロトコル機械の有限制御部の状態がそれぞれ  $s_p, s_q$  である状態について、あるレジスタの値において実行可能な送信アクションがない。すなわち、 $s_p$  と  $s_q$  において、それぞれの有限制御部の状態が始状態となるアクションがすべて受信アクションであるか、送信アクションが存在したとしてもそれらのアクションの遷移条件の論理和が恒真とならない。

次に、プロトコルの2つのプロトコル機械のレジスタの数をそれぞれ  $r_P, r_Q$  とするとき、それぞれのレジスタにおけるすべてのとりうる値の範囲  $\mathcal{N}^{(r_P+r_Q)}$  を次の条件を満たすように分割することを考える。すなわち、それぞれの分割は互いに素であり、すべての分割の和をとると  $\mathcal{N}^{(r_P+r_Q)}$  に一致する。以下、そのような分割数を  $N_j$  とし、 $E_j$  を一つの分割とする。

- ある分割  $E_j$  に含まれる任意のレジスタ値について、空チャネル状態で成立する各積項  $R_i$  ( $1 \leq i \leq N_c$ ) の AF4 型原子式のそれぞれの差分不等式が成立するかしないかが一意に定まる。

このとき、有限制御部の状態  $s_p, s_q$  について図 2 のように新しい有限制御部の状態とアクションを付加する。ここで、有限制御部の状態が  $s_p, s_q$  であり、レジスタの値が分割  $E_j$  を満たす系の状態  $s$  に対して積項  $R_{min}(s)$  を満たすような状態に遷移し、遷移条件が  $E_j$  であるようなアクションを、 $s_{pp}$  から  $s_{p'}$  へのアクションの代わりに付加する。

以下では、分割  $E_j$  ( $1 \leq j \leq N_j$ ) を求める手順を考える。

1. 空チャネル状態で成立するすべての積項  $R_i$  ( $1 \leq i \leq N_c$ ) の AF4 型原子式において、レジスタ  $x_k, x_l$  ( $1 \leq k < l \leq r$ ) の関係について表している差分不等式 ( $x_k - x_l \leq c$  または、 $x_l - x_k \leq c$ ) の集合を  $C_{kl}$  とする。
2. 集合  $C_{kl}$  中の差分不等式  $r_1, \dots, r_{N_{kl}}$  を以下のように定義される整数  $d$  の昇順  $r_{u_1}, r_{u_2}, \dots, r_{u_{N_{kl}}}$  で並べる。

- $r_m : x_k - x_l \leq c_m$ ならば,  $d_m = c_m$ とする.
- $r_m : x_l - x_k \leq c_m$ ならば,  $d_m = -c_m - 1$ とする.

ここで,  $m \neq n$  に対して  $d_m = d_n$  である場合, 一方を除く.

3.  $x_k - x_l$  のとりうる値を次のような区間に分割し, その集合を  $I_{kl}$  とする.
  - (a)  $r_{u_1}$  について, 区間  $[-\infty, d_{u_1}]$  を集合  $I_{kl}$  に加える.
  - (b)  $u_1 \leq u_m \leq u_{N_{kl}-1}$  で,  $r_{u_m}, r_{u_{m+1}}$  について, 区間  $[d_{u_m} + 1, d_{u_{m+1}}]$  を集合  $I_{kl}$  に加える.
  - (c)  $r_{u_{N_{kl}}}$  について, 区間  $[d_{u_{N_{kl}}} + 1, \infty]$  を集合  $I_{kl}$  に加える.
4. 以上 1., 2., 3. の操作をすべてのレジスタ  $x_k, x_l$  ( $1 \leq k < l \leq r$ ) に関しておこない, すべての集合  $I_{kl}$  から区間を 1 つずつ選び,  $x_k$  と  $x_l$  の差分が区間内にあることを表す差分不等式を求め, その積を  $E_j$  とする.
5.  $E_j$  を満たすレジスタの値が存在するかどうかは Bellman-Ford のアルゴリズム [5] を用いて  $O(rN_I)$  (ただし  $N_I$  は  $E_j$  を表す連立不等式の不等式の数) で確かめることができ, 存在するならば  $E_j$  を求める分割とする.

### 3.2 定理の証明

以下で定理 2, 3 を証明する. また, 集合  $C$  と集合  $C_u$  の差集合を  $C_s$  とする.

#### 定理 2 の証明

あるレジスタ  $x$  が集合  $C_u$  のすべての差分不等式に現れ, 集合  $C_u$  の差分不等式のレジスタ  $x$  の係数がすべて -1 であるときを考える.

集合  $C_u$  の差分不等式を

$$\begin{aligned} y_1 - x &\leq c_1 \\ &\dots \\ y_{N_u} - x &\leq c_{N_u}. \end{aligned} \quad (1)$$

とする. 式 (1) を成立させるためには,  $x$  の値を以下の条件を満たす  $h_1$  を用いて  $x + h_1$  で置き換えてやればよい.

$$y_1 - x = c_1 + h_1 \quad (2)$$

ここで, 式 (1) が成立しないことより  $h_1 > 0$  である.

集合  $C_u$  の他の差分不等式についても同様にして考えると, 式  $h = \max[h_1, \dots, h_{N_u}]$  で得られる  $h$  の値について, レジスタ  $x$  の値を  $x + h$  に変更することにより,  $C_u$  のすべての差分不等式が成立するようになることがわかる.

一方そのような変更を行っても集合  $C_s$  の差分不等式が成立しなくなることがないことを一般性を失うことなく,  $\max[h_1, \dots, h_{N_u}] = h_1$  と仮定して示す.

#### 1. 集合 $C_s$ 中の差分不等式 $y_i - x \leq c_i$ :

$h_1 > 0$  なので, レジスタ  $x$  の値を  $x + h$  と変更しても差分不等式  $y_i - x \leq c_i$  は成立する.

#### 2. 集合 $C_s$ 中の差分不等式 $x - y_j \leq c_j$ :

レジスタ  $x$  の値を  $x + h_1$  と変更したときこの差分不等式が成立しなくなる, すなわち, 式 (3) が成立するとする.

$$(x + h_1) - y_j > c_j \quad (3)$$

$y_1 - x \leq c_1$  と  $x - y_j \leq c_j$  が集合  $C$  に含まれていることより,  $y_1 - y_j \leq c'$  (ただし  $c' \leq c_1 + c_j$ ) も集合  $C$  に存在する. ところが, 式 (2) (3) より  $(c_j + y_j) < (x + h_1) = (y_1 - c_1)$  が得られ,  $(y_1 - y_j) > (c_1 + c_j)$  となるので,  $y_1 - y_j \leq c'$  が成立しないことがわかる. これは  $C$  中の  $x$  を含まない不等式が全て成立するという仮定に反する.

よって, レジスタ  $x$  の値を  $x + h_1$  に変更しても差分不等式  $x - y_j \leq c_j$  は成立する.

以上より, 集合  $C_u$  の差分不等式のレジスタ  $x$  の係数がすべて -1 であるときは, レジスタ  $x$  の値のみを変更することにより, 集合  $C$  のすべての差分不等式を成立させることができることがわかる.

同様に, レジスタ  $x$  の係数がすべて -1 であるときも, 一つのレジスタの値のみを変更することにより, 集合  $C$  のすべての差分不等式を成立させることができることを示せる.

次に, 集合  $C_u$  の差分不等式のレジスタ  $x$  の係数が同じでないときを考える.

このとき, 集合  $C_u$  中に以下のような 2 つの不等式が存在する.

$$x - y \leq c_y \quad (4)$$

$$z - x \leq c_z \quad (5)$$

式 (4) (5) がともに成立していない場合, レジスタ  $x$  の値をどのように変更しても式 (4) (5) をともに成立させることができないことは明らか. よって, 集合  $C_u$  の差分不等式のレジスタ  $x$  の係数が同じでないときは, レジスタ  $x$  の値を変更しても集合  $C$  のすべての差分不等式を成立させることができないことがわかる.  $\square$

#### 定理 3 の証明

集合  $C_u$  の差分不等式について次の 2 つの集合を定義する.

- レジスタ  $x$  が現れレジスタ  $y$  が現れない差分不等式の集合  $C_u(x)$
- レジスタ  $y$  が現れレジスタ  $x$  が現れない差分不等式の集合  $C_u(y)$

また, レジスタ  $x, y$  の両方が現れる差分不等式があり,  $x$  の係数が集合  $C_u(x)$  の差分不等式の  $x$  の係数と同じであれば, この差分不等式を集合  $C_u(x)$  に加え,  $y$  の係数が集合  $C_u(y)$  の差分不等式の  $y$  の係数と同じであれば, この差分不等式を集合  $C_u(y)$  に加える. このようにすることにより集合  $C_u$  を集合  $C_u(x)$  と  $C_u(y)$  に分割できる.

$C_u(x)$  のすべての差分不等式の係数が同じであるので, 定理 2 と同様にレジスタ  $x$  の値を変更すれば,  $C_s$  中の差分不

等式が成立したままで  $C_u(x)$  のすべての差分不等式を成立させることができる。  $C_u(y)$  についても同様である。

よって、以上より、レジスタ  $x, y$  について定理 3 の条件が成立する時、レジスタ  $x, y$  の値を変更することにより集合  $C$  のすべての差分不等式を成立させることができることがわかる。

### 3.3 未定義受信からの回復機能の付加

未定義受信状態は、一方のプロトコル機械への通信路の先頭のメッセージ  $\langle d, n_1, \dots, n_{N_d} \rangle$  を受けとるアクションが実行できない系の状態である。そのようなメッセージを受けとった場合、現在の状態にもっとも近いそのメッセージを受けとることができる状態でのそのメッセージを受けとった場合の遷移先へ遷移するようなアクションを付加する。

各アクション  $t = (sf, d, sf', C, R) \in T$  について、アクション  $t$  が実行可能な状態で成立する論理式  $s_a(t)$  を以下のように定める。

定義 5 アクション  $t = (sf, d, sf', C, R) \in T$  について、論理式  $s_a(t)$  は以下のような原子式の積で表される。

AF1': 有限制御部の状態  $sf$ .

AF2': 連立差分不等式  $C$ .

メッセージ型  $d$  のメッセージを受けとるアクション  $t_d$  が  $N_{t_d}$  個ある場合、どのアクション  $t_{d_i}$  ( $1 \leq i \leq N_{t_d}$ ) についての論理式  $s_a(t_{d_i})$  を満たす状態がそのメッセージを受けとる状態とするかを決定するために、系の状態  $s$  と論理式  $s_a(t_{d_i})$  との距離  $D(s, s_a(t_{d_i}))$  を定義する。そこで、論理式  $s_a(t_{d_i})$  ( $1 \leq i \leq N_{t_d}$ ) の中で  $D(s, s_a(t_{d_i}))$  の値が最小となるような論理式  $s_a(t_{d_{\min}})(s)$  を満たす状態をそのメッセージを受けとる状態とし、未定義受信状態  $s$  からの回復のためのアクションを付加する。

ここで、距離  $D(s, s_a(t_{d_i}))$  は 3.1 の距離  $D(s, R_i)$  と同様

に定義でき、値を決定できる。  
以上で議論した未定義受信状態からの回復機能をプロトコルに付加するためにはあらかじめすべての未定義受信状態を列挙し、それらの状態からの回復のためのアクションを付加することが必要となる。ここでは、一方のプロトコル機械  $P$  にメッセージ型が  $d$  であるメッセージに対する未定義受信状態からの回復機能の付加を考える。

以下では  $s_p$  を次の条件を満たす  $P$  の有限制御部の状態とする。

- 有限制御部の状態が  $s_p$  である  $P$  の状態について、あるレジスタ、メッセージのパラメータの値において実行可能なメッセージ  $\langle d, n_1, \dots, n_{N_d} \rangle$  を受けとるアクションがない。すなわち、 $s_p$  が始状態となるメッセージ  $\langle d, n_1, \dots, n_{N_d} \rangle$  を受けとるアクションがないか、そのようなアクションが存在してもそれらのアクションの遷移条件の論理和が恒真とならない。

次に、 $P$  のレジスタの数を  $r_p$  とするとき、値の範囲  $\mathcal{N}^{(r_p+N_d)}$  を次の条件を満たすように分割し、そのような分割数を  $N'_j$  とし、 $E'_j$  を一つの分割とする。

- ある分割  $E'_j$  に含まれる任意のレジスタ、メッセージのパラメータの値について、各論理式  $s_a(t_{d_i})$  ( $1 \leq i \leq N_{t_d}$ ) の AF2' 型原子式のそれぞれの差分不等式が成立するかしないかが一意に定まる。

このような  $E'_j$  を 3.1 と同様求めることができる。

それぞれの  $E'_j$  ( $1 \leq j \leq N'_j$ ) について、有限制御部の状態が  $s_p$  で、レジスタの値が  $E'_j$  を満たすような系の状態  $s$  に対して論理式  $s_a(t_{d_{\min}})(s)$  を満たすような状態でアクション  $t_{d_{\min}}$  を実行して遷移した場合の遷移先を  $p'$  とするとき、 $s_p$  を始状態、 $E'_j$  を遷移条件とし、メッセージ  $\langle d, n_1, \dots, n_{N_d} \rangle$  を受けとり、 $p' \rightarrow$  遷移するようなアクションを付加することにより、 $P$  にメッセージ  $\langle d, n_1, \dots, n_{N_d} \rangle$  に対する未定義受信状態からの回復機能を付加する。

## 4 まとめ

本稿では、ECFSM のサブクラスである FSM-C でモデル化された通信プロトコルにおいて、フォールトが発生しない時には安全性が保証されているプロトコルに self-stabilizing 機能を付加する手法を提案した。提案手法ではできるだけ初期状態以外の安全な状態の中で最も近い状態に回復するようにした。現在、図 1 で表された例プロトコルに提案手法により self-stabilizing 機能を付加したものについて、回復先の状態を初期状態とするようなアクションを実行する確立や、拡張後のプロトコルの規模を調べることにより、提案手法の評価を行なっている。今後、いくつかの例プロトコルについても、同様に提案手法の評価を行なう予定である。

## 参考文献

- [1] M.Gouda, and N.Multari. "Stabilizing communication protocols." *IEEE Trans. Comput.*, vol.40, no.4, pp.448-458, 1991.
- [2] A.Arora, and M.Gouda. "Closure and convergence: A formulation of fault-tolerant computing." *Proc. 22nd Int. Symp. Fault-Tolerant Computing*, 1992, pp. 396-403.
- [3] M.Gouda, and A.Arora. "Closure and Convergence : A Foundation of Fault-Tolerant Computing ." *IEEE Trans. on SOFTWARE ENG.*, VOL.19, NO.11, NOVEMBER 1993.
- [4] M.Higuchi, et al. "A Verification Method via Invariant for Communication Protocols Modeled as Extended Communicating Finite-State Machines." *IEICE TRANS. COMMUN.*, VOL.E76-B, NO.11 NOVEMBER 1993.
- [5] Cormen,T.H., et al. "Introduction to Algorithms." *The MIT Press.*, pp.539-543, 1990.