入力が競合する有限状態機械群からなる通信ソフトウェアの適合性試験の一手法

深田敦史[†,☆] 鍛冶忠司[†,☆☆] 東野輝夫[†]
谷口健一[†] 森将豪[††]

本論文では，入力を奪い合いながら並行に動作するDFSM群（DFSMsの直積マシン）としてモデル化される通信プロトコルのあるサブクラスに対して，DFSM群の状態と遷移の数の和に比例する程度のコストで効率良く適合性試験を行うための1つの手法を提案する．提案する手法では，まず，Wp-法を用いて単独に各DFSMの試験を行う場合に用いる特性集合の和集合をシステム全体（DFSM群）の試験を行うための特性集合とする．入力を奪いあう場合，与えられた各特性系列に対して正しく反応を返す可能性のある状態対は複数存在する可能性がある．このため，各DFSMの1つの状態sの存在を確認するために，まず一定の条件を満たすW集合を構成する．次にその状態sを含む適切な状態組を1つ選び，その状態組に対する特性系列の反応をチェックすることにより状態sの存在を確認する．

A Conformance Testing for Communication Protocols Modeled as a Set of DFSMs

Atsushi Fukada,[†,☆] Tadashi Kaji,[†,☆☆] Teruo Higashino,[†]
Kenichi Taniguchi[†] and Masaaki Mori[††]

In this paper, we propose an effective conformance testing method for a subclass of protocols modeled as a set of DFSMs. The cost in the proposed testing method is only proportional to the *sum* of the numbers of states and transitions in a given set of DFSMs. In our method, we find a characterization set for each DFSM, which is used to test the DFSM alone in Wp-method, and the union of the characterization sets is used as a characterization set for the total system. For a set of DFSMs with common inputs, there may exist two or more tuples of states that have correct responses against a given characterization set. So, in order to identify each state s in a DFSM, we find a characterization set with some specific properties. Then, we select a suitable tuple of states containing the state s, and identify the state s by checking their response to the characterization set.

1. まえがき

通信プロトコルに対する適合性試験は通信システムの信頼性を高めるために有効である．このため，従来からTT法[7]，W-法[1]，DS法[3]，UIO法[8]など，適合性試験系列を機械的に生成する手法が数多く提案されてきた．これらの試験系列生成に関する研究は，単一の決定性有限状態機械（DFSM）でモデル化される通信プロトコルを対象としたものが多いが，非決定的な動作を行うモデルや並行に協調して動作する並行モデルに対する研究も行われてきている．

近年高速ネットワークの発展にともない，ISDNのマルチリンクプロトコル（MP）などのように複数のリンクの制御を行う通信プロトコルや，ネットワークスイッチのように複数のチャネルの制御を行う通信システムが数多く利用されている．また，下位層の提供するサービスを複数並行に利用して上位層のサービスを提供するようなシステムも開発されている．そのような通信プロトコルでは，1つのリンクやチャネルの制御部を1つのDFSMで記述し，通信プロトコル全体は，並行に動作し共通する入力を奪い合いながら非決定的に動作するDFSM群（DFSMの直積マシン）

† 大阪大学基礎工学部情報科学科
Department of Information and Computer Sciences, Faculty of Engineering Science, Osaka University
☆ 現在，ヤマハ株式会社
Presently with Yamaha Co. Ltd.
☆☆ 現在，株式会社日立製作所
Presently with Hitachi Ltd.
†† 滋賀大学経済学部情報管理学科
Faculty of Economics, Shiga University

からなる並行モデルとしてモデル化することが自然な場合も多い．このような並行モデルとして与えられた通信プロトコルに対する適合性試験の1つの方法は，与えられたDFSM群全体の動作を表す単一の非決定性有限状態機械（NFSM）を構成し，そのNFSMに対してGWp-法[6]などの一般のNFSMに対する試験手法を適用することである．しかし，一般に，このNFSMの状態数はもとのDFSM群の状態数の積に比例するため，生成される試験系列の数がDFSM群の状態数の積に比例するという問題がある．この問題を解決する1つの方法として，DFSM群の各DFSMに対して，個別に試験を行う手法が考えられる．文献5)は，FSM間の通信やFSMの内部動作をともなう通信プロトコルを対象に，この方針に基づいた試験を行っている．

本研究では，共通する入力を奪い合うことで全体として非決定的な動作を行うDFSM群で表される通信プロトコルのあるサブクラスに対し，実装も同じ個数のDFSM群（各DFSMを部分実装と呼ぶ）で構成され，各部分実装の状態数は対応する仕様のDFSMの状態数を超えないと仮定する．この仮定の下で，DFSM群の状態と遷移の数の和のオーダーで仕様の各DFSMのすべての状態と遷移の確認を行う適合性試験手法を提案する．

提案する手法では，まずWp-法を用いて単独に各DFSMの試験を行う場合に用いる特性集合の和集合をシステム全体（DFSM群）の試験を行うための特性集合とする．特性集合としては，共通する入力の奪い合いがあっても，各DFSMの状態を識別できるような系列の集合を選べるものと仮定する．また，各DFSMには特性集合に含まれるすべての共通入力に反応しない状態が少なくとも1個存在するものと仮定し，そのようなクラスを試験対象とする試験は，システム内の1つのDFSM A_i のある状態 s の存在を調べる際に，ある方針に基づきその状態 s を含む適切な状態組 $\langle ..., s, ... \rangle$ を1つ選ぶ．次に，選んだ状態組に対してその状態 s の確認のための特性系列を与え仕様と同じ反応を返すかどうか試験する．すべての状態に対して同様の方法で試験し，仕様と同じ反応を返せば，状態の確認が行えたと判断する．

以下，2章で本論文で扱うモデルCoupled DFSMsとそれに対するフォールトモデルを定義する．3章では，従来提案されている非決定性有限状態機械に対する適合性試験手法であるGWp-法の概要について述べる．4章で提案する試験手法について説明する．5章で提案する試験手法の正しさを説明する．最後にまとめを6章で述べる．

2. 入力が競合する有限状態機械群

2.1 通信プロトコルの仕様と実装

定義2.1（有限状態機械） 有限状態機械（FSM）は，

$$A = (S, X, Y, \delta, \lambda, s_0)$$

と定義される．ここで，S，X，Y はそれぞれ，A の状態集合，入力記号の集合，出力記号の集合である．δ は遷移関数 $(S \times X \to S)$ で，λ は出力関数 $(S \times X \to Y)$ である．また，s_0 は A の初期状態である． □

2つの状態 s, t が**等価**であるとは，任意の入力系列 σ^i に対して，$\lambda(s, \sigma^i) = \lambda(t, \sigma^i)$ が成り立つことである．また，2つのFSMが**等価**であるとは，その初期状態どうしが等価であることである．FSMが**最小**とは，その機械の相異なる2つの状態が等価でないことである．FSMが**完全**とは，任意の状態と入力の組に対して，遷移関数と出力関数がともに定義されていることである．本論文では，与えられたFSMが完全でない場合，遷移関数，出力関数が定義されていない状態 s，入力 x の組に対して，何も出力しないことを表す特殊な出力記号 (ε) を出力し，s 自身に遷移するような遷移を付け加えることにより，完全なFSMと見なす．以下，そのような遷移が存在する場合，状態 s は入力 x を**無視する**という．FSMが**初期状態から連結**とは，その機械の任意の状態に対して，初期状態からその状態への遷移系列が存在することである．FSMが決定的な動作を行うとは，ある状態と入力に対して，一意に出力と遷移先の状態が定まることである．決定的な動作のみを行うFSMを決定性FSM（DFSM）といい，DFSMでないものを非決定性FSM（NFSM）という．また，1つの状態で，a/b，a/c のように同一の入力 a に対して異なる出力を行うような非決定性遷移を観測可能な非決定性遷移といい，a/b，a/b のように同一の入出力を行う非決定性遷移を観測不能な非決定性遷移という．非決定性遷移が観測可能な遷移のみであるようなNFSMを観測可能なNFSM（Observable NFSM: ONFSM）という．

定義2.2（Coupled DFSMs）
Coupled DFSMsは k 個のDFSMの組，

$$A = (A_1, A_2, \ldots A_k)$$

からなるDFSM群である．$A_1, A_2, \ldots, A_k$ はそれぞれDFSMであり，各 $A_i (1 \le i \le k)$

$$A_i = (S_i, X_i, Y_i, \delta_i, \lambda_i, s_{i0})$$

は，完全，初期状態から連結，最小のDFSMである．また，Coupled DFSMs全体を同時に初期状態に戻す

リセット操作の存在を仮定する. □

以下, $x \in X_i \cap X_j (\neq \emptyset)$ なる入力 x を**共通する入力**と呼ぶ. 共通する入力 x が環境から与えられた場合, A_i と A_j の一方が非決定的に選ばれ, 状態遷移を行う. ただし, A_i が入力 x を無視せず, A_j が入力 x を無視する場合, 必ず A_i が入力 x を獲得し, 対応する状態遷移を行うものとする.

定義 2.3 (通信プロトコルの仕様)
本論文で扱う通信プロトコルの仕様は, k 個の DFSM からなる Coupled DFSMs,

$$A = (A_1, A_2, \ldots A_k)$$

として与えられ, 全体として観測不能な非決定性遷移を含まないものとする. □

以下, $A_i (1 \leq i \leq k)$ を A の部分仕様という. 通信プロトコルの仕様 A が全体として観測不能な非決定性遷移を含まないと仮定しているので, A_i に a/b なる遷移が存在する場合, 別の A_j には a/c なる遷移は存在してもよいが, 同じ a/b なる遷移は存在してはならない.

定義 2.4 (フォールトモデル)
本論文で扱う実装 I は仕様 A と同じ個数 (k 個) の DFSM からなる Coupled DFSMs,

$$I = (I_1, I_2, \ldots, I_k)$$

として構成されるものとする. また, I_j は次のような性質を満たすものとする.

- I_j は A_j と同じ入力記号の集合 X_j を持つ.
- I_j の出力記号の集合は, 通信プロトコル全体の出力記号の集合 $Y = Y_1 \cup Y_2 \cup \ldots \cup Y_k$ である (他の DFSM で使われる出力記号を出力するような誤りがあってもよい).
- I_j はたかだか A_j の状態数 n_j 個以下の状態で実現されている.

また, I 中のすべての DFSM を同時に初期状態に戻すリセット操作には誤りがないと仮定する. □

以下, $I_j (1 \leq j \leq k)$ を部分実装という. 上述のようにプロトコルの実装 I は仕様と同じ個数の DFSM 群で実現されていると仮定するが, 内部の状態は観測できない, すなわち, 実装はブラックボックスとして試験を行う. また, 仕様 A は全体として ONFSM でなければならないが, 実装 I は観測不能な非決定性遷移を持っても構わない.

定義 2.5 (通信プロトコルの適合性) 上述のようなモデルで記述された通信プロトコルの仕様 $A = (A_1, A_2, \ldots, A_k)$ と実装 $I = (I_1, I_2, \ldots, I_k)$ に対して, I のすべての部分実装 I_j が対応する部分仕様 A_j と等価であるとき, I は A の正しい実装であると定義する. □

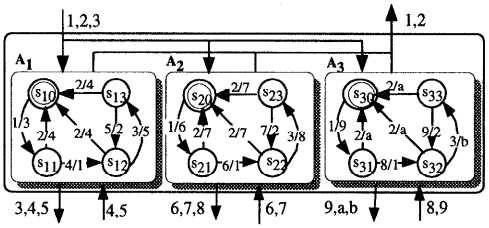


図 1 通信プロトコルの例
Fig. 1 Example of communication protocol.

表 1 図 1 の入力, 出力, 状態
Table 1 Inputs, outputs and states of Fig. 1.

Inputs	
1	Link Increase request
2	Link Decrease request
3	Data Transfer request
4, 6, 8	Connect Confirm
5, 7, 9	Data Transfer ack
Outputs	
1	Connect Confirm
2	Data Transfer ack
3, 6, 9	Link Increase request
4, 7, a	Link Decrease request
5, 8, b	Data Transfer request
States	
s_{i0}	disconnected (initial state)
s_{i1}	wait for Connect Confirm
s_{i2}	connected
s_{i3}	wait for Data Transfer

2.2 通信プロトコルの仕様記述例

図 1 の通信プロトコルを考える. 図 1 は, 動的に下位層とのリンクの数を変化させることができるようなプロトコルの仕様である. この仕様では上位層からの指示によって下位層と最大 3 個までのリンクを構成することができ, 単一のリンクの制御部が 1 つの DFSM としてモデル化されている. **表 1** で入力, 出力, 状態の内容を概説する.

上位層からこのプロトコルに対して "リンク増加要求" が与えられると, まだ下位層とリンクを持っていない DFSM の中の 1 つがリンクを構成する. 逆に, 上位層から "リンク削減要求" が与えられると, 下位層とのリンクを持っている DFSM の中の 1 つがリンクを切断する. このようなリンクの数を増減させる要求は, 動作可能な DFSM が要求を奪い合うと考えることができる. すなわち, 入力 1 ("リンク増加要求") が与えられた場合, 1 を無視しない状態にある DFSM の 1 つが非決定的に選ばれ動作する. このため, たとえば, A_2 が 1 を無視する状態 (状態 1 など) にある場合は, A_2 が選ばれることはない. すべての DFSM

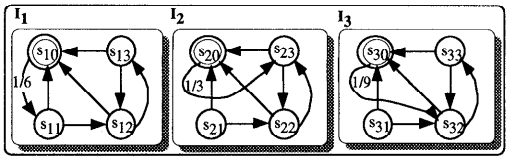


図2 図1の誤った実装の例
Fig. 2 Example of faulty IUT for Fig. 1.

が1を無視する状態にある場合，そのうちの1つのDFSMが入力1を受け取り ε を出力すると考える．なお，この仕様では，リンクの数を増減させる要求に対してどのDFSMが動作したかは出力によって知ることができる．

一方，**図2**は，図1の通信プロトコルに対する誤った実装の例である．図2の部分実装 I_1 や I_2 を単独に試験を行う場合には，状態 s_{20} で入力1を与えることによって仕様と異なる出力が得られるため誤りを検出できる．しかし，Coupled DFSMsの状態組 (s_{10}, s_{20}, s_{30}) においては，入力1を与えても仕様と同じ出力（3,6,9のいずれか）が出力されるため，単純に誤りがあることは検出できない．

3. GWp-法の概要

GWp-法は，単一のDFSMに対する適合性試験手法の1つであるWp-法をONFSMを扱えるように拡張したものである．試験は，仕様の状態が実装にも存在するかどうかの確認（状態の確認）と仕様の遷移が実装にも正しく実装されているかどうかの確認（遷移の確認）の2つを行う．GWp-法で用いられる試験系列集合は以下のような2種類の系列集合，特性集合（W 集合）と先行系列の集合（V 集合），を構成することによって得られる．

定義3.1（特性集合） FSMの特性集合は，特性系列と呼ばれる入力系列の集合 W である．FSMの任意の状態に対して，W のすべての入力系列を加えることによって得られる出力系列の集合から，入力系列を与えたときのFSMの状態がどの状態であったのかを特定できるとき，W を特性集合という． □

定義3.2（先行系列） FSMの状態 s への先行系列は，FSMを初期状態から状態 s へ遷移させるような入力系列である．与えられたFSMのすべての状態への先行系列の集合が V 集合である． □

定義3.3（試験系列） 状態確認のために実装に与える試験系列の集合は V と W の連結 $V.W$ である．また，遷移確認は，V と X（入力記号の集合）と W を用いて，

$$V.X \oplus W = \{\sigma.w | \sigma \in V.X, s_0 \xrightarrow{\sigma} s_i, w \in W\}$$

で与えられる． □

GWp-法では，まず実装にリセット操作を施してから試験系列を与え，得られた出力系列が仕様の出力する系列と一致するかどうかを調べる．しかし，ONFSMは非決定的な動作を行うため，一般には，1つの入力系列に対する出力系列は複数存在する．そこで，同じ系列を複数回与え，得られた出力系列の集合が仕様の出力系列の集合と一致するかどうかを調べる．一致しない場合には実装に誤りがあると判断する．すべての試験系列に対して一致した場合に，与えられた実装は正しいと判断する．また，ある状態 s への先行系列を与えたときの出力系列が予想した出力系列でない場合には，そのFSMが非決定的な動作により目的の状態以外の状態に遷移したと考え，試験をやり直す．

4. 提案する試験手法

提案する手法では，Coupled DFSMsとしてモデル化された通信プロトコルの仕様 $A = (A_1, A_2, \ldots, A_k)$ と実装 $I = (I_1, I_2, \ldots, I_k)$ に対して，試験に用いる特性集合に若干の条件を課し，その条件を満たす特性集合を用いてDFSMの状態数の和に比例するコストで(1)状態の確認と，(2)遷移の確認の2つの試験を行う．

4.1 特性集合の与え方

提案する手法で状態確認を行うにあたって，試験に用いる各部分仕様（DFSM）A_i の特性集合 W_i の生成に関して，下記のような2つの条件（条件1，2）を満足するような W_i を生成できるものと仮定する．

定義4.1（共通入力記号に関する制約（条件1）） 部分仕様 A_i に対する特性集合 W_i は，その入力系列に他のDFSMに奪われる可能性のあるような共通入力記号が含まれる場合，W_i からその入力記号を除いた入力系列でも A_i の各状態を区別できるような W_i である． □

次に条件1を満足する W_i に対して，$W = W_1 \cup W_2 \cup \cdots \cup W_k$ となる W を考える．また，W の各入力系列 σ に対して，部分実装 I_i の反応する可能性のある記号のみを残し，残りの記号を取り除いた系列を $\overline{\sigma_i}$ で表し，W のすべての入力系列 σ に対する $\overline{\sigma_i}$ の集合を W_i' とする．

例として，図1のプロトコルを考えてみる．求める W は，たとえば，$W_1 = \{35, 4, 5\}$，$W_2 = \{16, 7, 6\}$，$W_3 = \{18, 9, 8\}$ の場合，$W = \{35, 4, 5, 16, 7, 6, 18, 9, 8\}$ となる．また，$W_1' =$

表2 図1に対するW集合の例
Table 2 Example of W sets for Fig.1.

W_1 for A_1			
State	35	4	5
s_{10}	$\varepsilon\varepsilon$	ε	ε
s_{11}	$\varepsilon\varepsilon$	1	ε
s_{12}	$\underline{5}2$	ε	ε
s_{13}	$\varepsilon 2$	ε	2
W_2 for A_2			
State	16	7	6
s_{20}	$\underline{6}1$	ε	ε
s_{21}	$\varepsilon 1$	ε	1
s_{22}	$\varepsilon\varepsilon$	ε	ε
s_{23}	$\varepsilon\varepsilon$	2	ε
W_3 for A_3			
State	18	9	8
s_{30}	$\underline{9}1$	ε	ε
s_{31}	$\varepsilon 1$	ε	1
s_{32}	$\varepsilon\varepsilon$	ε	ε
s_{33}	$\varepsilon\varepsilon$	2	ε

$\{35,4,5,1\}$, $W_2' = \{16,7,6,3\}$, $W_3' = \{18,9,8,3\}$ である.

以上の例に対して条件1がどのような意味を持つのかを説明する. たとえばW_1の生成に関して上で述べた条件を考慮する必要がある. **表2**より$W_1'' = \{35,4\}$としてもA_1の4状態を区別できる. しかし, W_1''の系列に含まれる入力記号3は他のA_2かA_3によって奪われ, 表中の下線の付いている出力5がεになる可能性がある. このとき状態s_{12}と状態s_{13}は区別できない. そのため, そのような場合でも各状態を判別できるような特性集合を構成するために, 表2のW_1の要素にあるように, 系列5を加えることで, たとえ共通する入力が他のA_2かA_3に奪われた場合でも状態s_{12}と状態s_{13}を区別できるようにしている. なお, W_2の系列6とW_3の系列8も同様の理由により加えている.

定義4.2(**他の部分仕様に関する制約(条件2)**) 部分仕様A_iに対する特性集合W_i'の系列の中で他のFSMに奪われてしまう可能性のある共通入力記号の集合をC_iとする. このときW_i'は, C_iに含まれる共通入力記号すべてを, 各部分仕様$A_j(j \neq i)$の少なくとも1つの状態が無視するようなW_i'である. □

状態確認に関しては, 以上の2条件を仮定する. 一般には, 各部分仕様A_iに対する特性集合が存在しても上述のような2条件を満足するW_iが必ず存在するとは限らない. しかし, 共通する入力が頻繁に現れない場合, 特性集合に共通入力記号が含まれる可能性はそれほど高くないので, このようなW_iを与えることができる場合が多い.

一方, 遷移の確認に関しては次のような問題点がある. たとえば, ある部分仕様A_iに共通入力xを無視するような状態sがあり, 別の部分仕様A_jがその共通入力xに対してどの状態でも同じ出力yを出力するような場合, A_iの状態sが共通入力xを無視することを確認する方法がない. これは, どのような状態組で試験を行ってもA_jがyを出力するため, A_iの状態sで共通入力xに対してyを出力しても, xを無視しても, 外部からは区別できないためである. そこで, 遷移の確認に関しては各DFSMに次のような仮定をおく.

定義4.3(**DFSMに対する仮定**) すべての共通入力記号それぞれに対して, 各部分仕様A_iにその入力を無視するような状態が少なくとも1つあるものと仮定する. □

4.2 状態の確認

各部分実装$A_i(1 \leq i \leq k)$のすべての状態に対して, 次のような方法で状態の確認を行う.

まず, 状態組の選択を行う. 各$A_j(j \neq i)$に対して, C_iに含まれる共通入力記号をすべて無視するような状態をs_j^iとする. A_iに含まれるN_i個$(N_i = |S_i|)$の各状態s_pに対して, 状態組$ss_p^i = \left(s_1^i, s_2^i, \cdots, s_k^i\right)\left(s_i^i = s_p\right)$を$A_i$の状態$s_p$を確認するための状態組とする. 状態組$ss_p^i$に$C_i$に含まれる共通入力記号を与えると, 仕様上は$A_i$の状態$s_p$のみが反応可能である.

次に, 試験系列を与える. 上述の方法で求めたN_i個の状態組に対して, 実装に試験系列$V.W_i'$を繰り返し与える. このときのVの各要素v_p^iは, 状態組ss_p^iの各状態$s_1^i, s_2^i, \cdots, s_k^i$への先行系列$v_1^i, v_2^i, \cdots, v_k^i$を連結した系列であり, 毎回同じものを用いる. 先行系列v_p^iを与えたときの出力系列が予想した出力系列でない場合は, DFSM群が非決定的な動作によりss_p^i以外の他の状態組に遷移したと考え, 試験をやり直す. 予想した出力系列が得られた場合, W_i'に対する状態組ss_p^iの反応を調べ, 仕様から考えられる反応と異なる場合, この実装には誤りがあると判断する.

以上を各部分実装のすべての状態に対して調べ, 誤りが検出できなければ, すべての状態の確認が行えたと判断する.

例としてここでは, 図1の部分仕様A_1の状態の確認を行うことにする.

まず, 状態組の選択を行う. W_1'には共通する入力$C_1 = \{1,3\}$が含まれている. たとえば, 部分仕様A_2の状態s_{21}と, A_3の状態s_{31}はともにこれらの共通する入力1, 3を無視しているので, 状態s_{10}につい

ては状態組 (s_{10}, s_{21}, s_{31}) を選ぶ．同様に，状態 s_{11} については状態組 (s_{10}, s_{21}, s_{31}) を，状態 s_{12} については状態組 (s_{12}, s_{21}, s_{31}) を，状態 s_{13} については状態組 (s_{13}, s_{21}, s_{31}) を選ぶ．

次に，試験系列を与える．実装に試験系列 $V.W_1'$ を繰り返し与え反応を調べる．このときの V は，先ほど選んだ4つの状態組への先行系列の集合，たとえば，$V = \{11, 111, 1411, 14511\}$ とする．また，もし状態組 (s_{12}, s_{21}, s_{31}) の試験を行う際に用いる先行系列として 1411 を与えたときの出力系列が 3169 でない場合は，非決定的な動作により状態組 (s_{12}, s_{21}, s_{31}) に遷移できなかったと判断し，試験をやり直す．

4.3 遷移の確認を行う

各部分仕様 A_i 上の各遷移 $s_p \xrightarrow{x/y} s_q (x \in X, y \in Y \cup \{\varepsilon\})$ が対応する部分実装 I_i にも正しく実装されているかどうかを次の2つの場合に分けて調べる．

入力 x を他の DFSM に奪われる可能性がない遷移の場合は，試験系列 $v.x.W_i'$ を与える．このときの v は状態 s_p の状態確認のときに用いた遷移系列を用いる．

入力 x を他の DFSM に奪われる可能性がある遷移の場合は，試験系列 $v.W \cup v.x.W_i'$ を与える．このときの v は次のように定める．まず入力 x を無視するような各 $A_j (j \neq i)$ の状態 s_j^i を見つけ，状態組 $ss_p^i = \left(s_1^i, s_2^i, \cdots, s_k^i\right) \left(s_i^i = s_p\right)$ を考える（前節の条件 1, 2 よりこのような状態組は必ず存在する）．このときの先行系列 v は状態組 ss_p^i に遷移させるための任意の遷移系列とする．試験系列 $v.W$ により実際にその状態組に遷移しているかどうかを調べ，$v.x.W_i'$ によりその遷移の正しさと遷移後の状態を調べる．

以上の試験をすべての遷移に対して行い，それらの実装に誤りがない場合，与えられた仕様に対して各遷移の実装は正しいと判断する．

なお，提案する手法では，状態確認，遷移の確認とも，試験系列の数は DFSM 群の状態と遷移の数の和でそれぞれおさえられる．

5. 試験手法の正しさについて

5.1 状態確認の手法の正しさ

仕様が Coupled DFSMs で与えられる場合，以下に示すような2つの点を考慮する必要がある．

1つは，共通する入力の存在である．ある状態組において，ある部分実装 I_i の状態 s_j の存在を調べる際に，共通入力 x を与え仕様どおりの出力 z が得られたとする．このとき，実際には，I_i の該当する状態が z を出力するようには実装されず，別の I_j が共通入力 x に対し z を出力するように誤って実装されている可能性がある．このように，実装が仕様どおりの反応を返すからといって，該当する状態の存在が確認されるわけではない．しかし，提案する試験法では，I_i 以外のすべての部分実装の状態を共通入力すべてを無視するような状態に遷移させるような先行系列を与えてから試験を行っている．たとえば，図1の A_1 の状態確認については，A_2，A_3 をそれぞれ状態 s_{21}，s_{31} に遷移させ，状態組 (s_{10}, s_{21}, s_{31})，(s_{11}, s_{21}, s_{31})，(s_{12}, s_{21}, s_{31})，(s_{13}, s_{21}, s_{31}) に対して試験を行っている．このため，そのような状態組では，仕様上は I_i のみが共通入力 x に反応するように記述されている．よって，仕様上はそのような状態組で共通入力 x に対して z 以外の出力が出力されることはない．このため試験に成功した場合，I_i の該当する状態は共通入力 x を無視するか，仕様どおりの正しい出力 z を返すかのいずれかの可能性しかない（z 以外の誤った出力を返す場合，試験の際にその誤った出力が観測されるので誤りを発見できる）．

もう1点は，実装 I に観測不能な非決定性遷移があってもよいと仮定している点である．この場合，実装 I にある1つの先行系列 α を与え，予想した出力系列が観測されたとしても，非決定性遷移により，毎回同じ状態組に遷移するとは限らない．このように，部分実装 I_i が先行系列 α の実行後複数の状態に遷移する場合でも，提案する試験法では，それらの各状態に複数回同じ試験系列を与え予想した出力系列が得られるかどうかを試験しているので，部分実装 I_i の各状態の試験系列に対する反応としては，仕様どおりの正しい出力を返す状態か，上述のように共通入力を無視する状態，かのいずれかの可能性しかない．

たとえば，A_1 の状態 s_{12} の確認に関して，W_1' に対する状態 s_{12} の反応は，仕様上は

$35/52, 4/\varepsilon, 5/\varepsilon, 1/\varepsilon$

である．しかし，上述のように，部分実装 I_1 が共通する入力 3 を無視するように誤って実装され，他の I_2 か I_3 のいずれかの1つの状態が誤って 5 を出力している可能性を考慮すると，I_1 の状態としては，上記のような反応を返す状態のほかに

$35/\varepsilon 2, 4/\varepsilon, 5/\varepsilon, 1/\varepsilon$

のような反応を返す状態が存在するかもしれない．このように，I_1 にはこれら2つの反応を返す状態のうち少なくとも1つが存在する（両方の状態が存在する可能性もある）．

以下，上述の議論をもう少し形式的に行う．まず，試験系列 W_1' に対して，A_1 の各状態に相当する反応

表3 論理変数
Table 3 Logical variables.

Logical Variables for A_1	
Variable	Response for W_1'
$\varphi 10$	$35/\varepsilon\varepsilon, 4/\varepsilon, 5/\varepsilon, 1/3$
$\varphi_\varepsilon 10$	$35/\varepsilon\varepsilon, 4/\varepsilon, 5/\varepsilon, 1/\varepsilon$
$\varphi 11$	$35/\varepsilon\varepsilon, 4/1, 5/\varepsilon, 1/\varepsilon$
$\varphi 12$	$35/52, 4/\varepsilon, 5/\varepsilon, 1/\varepsilon$
$\varphi_\varepsilon 12$	$35/\varepsilon 2, 4/\varepsilon, 5/\varepsilon, 1/\varepsilon$
$\varphi 13$	$35/\varepsilon 2, 4/\varepsilon, 5/2, 1/\varepsilon$
Logical Variables for A_2	
Variable	Response for W_2'
$\varphi 20$	$16/61, 7/\varepsilon, 6/\varepsilon, 3/\varepsilon$
$\varphi_\varepsilon 20$	$16/\varepsilon 1, 7/\varepsilon, 6/\varepsilon, 3/\varepsilon$
$\varphi 21$	$16/\varepsilon 1, 7/\varepsilon, 6/1, 3/\varepsilon$
$\varphi 22$	$16/\varepsilon\varepsilon, 7/\varepsilon, 6/\varepsilon, 3/8$
$\varphi_\varepsilon 22$	$16/\varepsilon\varepsilon, 7/\varepsilon, 6/\varepsilon, 3/\varepsilon$
$\varphi 23$	$16/\varepsilon\varepsilon, 7/2, 6/\varepsilon, 3/\varepsilon$
Logical Variables for A_3	
Variable	Response for W_3'
$\varphi 30$	$18/91, 9/\varepsilon, 8/\varepsilon, 3/\varepsilon$
$\varphi_\varepsilon 30$	$18/\varepsilon 1, 9/\varepsilon, 8/\varepsilon, 3/\varepsilon$
$\varphi 31$	$18/\varepsilon 1, 9/\varepsilon, 8/1, 3/\varepsilon$
$\varphi 32$	$18/\varepsilon\varepsilon, 9/\varepsilon, 8/\varepsilon, 3/b$
$\varphi_\varepsilon 32$	$18/\varepsilon\varepsilon, 9/\varepsilon, 8/\varepsilon, 3/\varepsilon$
$\varphi 33$	$18/\varepsilon\varepsilon, 9/2, 8/\varepsilon, 3/\varepsilon$

（出力系列）を返す状態とそれらの反応のうち共通入力を無視するような反応を返す状態をすべて求め，各反応を返す状態が I_1 の状態として存在するとき真，存在しないとき偽になるような論理変数を導入する．**表3** の A_1 についての変数は，A_1 に対するこのような論理変数の集合である．表3の A_1 に対する論理変数には，仕様どおりの反応を返す状態を表す論理変数 $\varphi 1k$（状態 s_{12} では $\varphi 12$）を必ず1つ含み，それ以外は，その状態変数の表す反応（出力）のうち共通入力に対する反応を ε に置き換えた反応を返す状態に相当する論理変数 $\varphi_\varepsilon 1k$（状態 s_{12} については $\varphi_\varepsilon 12$）が並ぶ．ただし，状態 s_{11}，s_{13} のように共通入力に対する反応がもともと ε の場合，$\varphi_\varepsilon 1k$ と $\varphi 1k$ は同じになるので，そのような場合 $\varphi_\varepsilon 1k$ は生成しない．

定義4.1で述べた条件1より，各特性集合 W_i' は，その入力系列に他のDFSMに奪われる可能性のあるような共通入力記号が含まれる場合，W_i' の共通入力に対する反応が ε になっても A_i の各状態を区別できる（W_i' の共通入力記号を除いた入力系列でも A_i の各状態を区別できる）ような特性集合である．

上述の性質より，たとえば，表3中の A_1 に対する論理変数が表す反応はすべて異なるものである．また，この例では部分実装 I_1 に対して，4つの状態組 (s_{10}, s_{21}, s_{31})，(s_{11}, s_{21}, s_{31})，(s_{12}, s_{21}, s_{31})，(s_{13}, s_{21}, s_{31}) を選び試験を行っているので，それらの状態組に対して次のような論理式が成り立つ．

- 状態組 (s_{10}, s_{21}, s_{31}) に対して，$\varphi 10 \vee \varphi_\varepsilon 10$
- 状態組 (s_{11}, s_{21}, s_{31}) に対して，$\varphi 11$
- 状態組 (s_{12}, s_{21}, s_{31}) に対して，$\varphi 12 \vee \varphi_\varepsilon 12$
- 状態組 (s_{13}, s_{21}, s_{31}) に対して，$\varphi 13$

このとき，各部分実装 I_i の状態数は対応する部分仕様 A_i の状態数 N_i を超えないと仮定しているので，各部分実装ごと，真となる論理変数はたかだか N_i 個である．また，上述のように部分仕様 A_i の状態ごとに1つの論理式が生成されるので，全部で N_i 個の論理式が得られる．このため，各論理式に2つの論理変数がある場合，いずれか一方のみが真になる．さらに，共通入力記号を無視するような論理変数 $\varphi_\varepsilon ih$ が真であるとすると，その特性集合を与えたときに得られた出力 z は，状態確認を行っている部分実装 I_i 以外のいずれかの部分実装 I_j のある状態で出力していることになる．しかし，I_j についても I_i と同様の試験を行うため，I_j に対する試験においても，仕様どおりの反応を返す状態か，そのうちの共通入力に対する反応が ε に置き換わったような反応を返す状態しか存在しないことが保証できる．たとえば，部分実装 A_2 については，状態組 (s_{11}, s_{20}, s_{31})，(s_{11}, s_{21}, s_{31})，(s_{11}, s_{22}, s_{31})，(s_{11}, s_{23}, s_{31}) に遷移させる先行系列を与えると，表3のような A_2 に対する6通りの論理変数が生成され，次のような4つの論理式が成り立つことが分かる．

- 状態組 (s_{11}, s_{20}, s_{31}) に対して，$\varphi 20 \vee \varphi_\varepsilon 20$
- 状態組 (s_{11}, s_{21}, s_{31}) に対して，$\varphi 21$
- 状態組 (s_{11}, s_{22}, s_{31}) に対して，$\varphi 22 \vee \varphi_\varepsilon 22$
- 状態組 (s_{11}, s_{23}, s_{31}) に対して，$\varphi 23$

表3のような A_2 に対するどの論理変数を見ても，共通入力3に対して A_1 が出力する5を反応として返す状態はない．同様に，A_3 についても表3のような論理変数しか生成されず，A_1 が出力する5を反応として返す状態はない．このように，ある部分実装 I_i に共通入力に対する出力 z を行わずその共通入力記号を無視するような状態があった場合でも，代わりにその出力 z を反応として返すような状態は I_i 以外のどの部分実装にも存在しない．よって，すべての状態組に対し試験が成功した場合，共通入力記号を無視するような状態 $\varphi_\varepsilon is$ が存在しないことが保証できる．

以上のことから，それぞれの論理式において，仕様どおりの反応を返す状態を表す論理変数 (φih) のみが真であることが保証でき，提案する方法で状態の確認が行える．

5.2 遷移確認の手法の正しさ

提案する手法（試験系列 $v.x.W_i'$ または $v.W \cup v.x.W_i'$ を与える方法）で，部分仕様 A_i 上の各遷移 $s_p \stackrel{x/y}{\rightarrow} s_q (x \in X, y \in Y \cup \{\varepsilon\})$ が対応する部分実装 I_i にも正しく実装されていることを確認できる理由を以下で説明する．

（Case 1）

まず，入力 x を他の DFSM に奪われる可能性がない遷移の場合，実装に試験系列 $v.x.W_i'$ （ただし v は状態 s_p の状態確認のときに用いた遷移系列）を与えると，v は s_p の状態確認の際に用いられているので，実装で x/y を実行する開始状態は s_p に相当する状態であることが分かる．また，$x.W_i'$ に対する実装の反応を調べることにより，x/y の実行後の状態も確認できる．

（Case 2）

入力 x を他の DFSM に奪われる可能性がある遷移の場合は，実装に試験系列 $v.W \cup v.x.W_i'$ を与える．一般に状態確認が済んでいない場合は，各 W_i' に対して期待どおりの反応があっても，仕様どおりの反応を返さない状態が存在する可能性もあるが，状態確認が終わっている場合，仕様どおりの反応が返ってくれば，それは部分仕様 A_i の状態に対応する状態である（他の可能性がないため）．このため，v を与えたときに遷移する状態組（$ss_p^i = \left(s_1^i, s_2^i, \cdots, s_k^i\right) \left(s_i^i = s_p\right)$ に相当する状態組）で W に対して期待どおりの反応が返された場合は，仕様どおりの状態組であることが保証できる．

次に $v.x.W_i'$ を用いて遷移の正しさを確認できる理由を次のような 3 つの場合に分けて説明する．

（Case 2.1）

まず，遷移 $s_p \stackrel{x/y}{\rightarrow} s_q (x \in X, y \in Y \cup \{\varepsilon\})$ の実行によって状態が変化する場合 $(s_p \neq s_q)$ を考える．I_i 以外の部分実装の各状態 s_j^i は入力 x を無視するように指定されているので，v を与えた状態では入力 x に対して y のみが出力されるはずである．このとき，入力 x が他の部分実装に奪われたかどうかは W_i' の反応で確認できる．もし，他の部分実装が y を出力し部分実装 I_i の状態 s_p が入力 x を無視するように実装されている場合，W_i' を与えたときに反応が s_q に対するものでなく，s_p に対する反応となるので，その誤りを検出できる．

（Case 2.2）

次に，遷移 $s_p \stackrel{x/y}{\rightarrow} s_q (x \in X, y \in Y \cup \{\varepsilon\})$ の実行によって状態が変化せず $(s_p = s_q)$, $(y = \varepsilon)$ の場合を考える．この場合，v を与えた状態組では共通する入力 x はすべての部分仕様で無視するようになっている．このため，v を与えた状態組で x を与えたときに，出力が ε でなければ，そのことが検出できる．また，他の状態に遷移する場合，W_i' を繰り返し与えたときの反応が状態 s_p の確認の際に出力される反応と異なってしまい，その誤りを検出できる．

（Case 2.3）

最後に遷移 $s_p \stackrel{x/y}{\rightarrow} s_q (x \in X, y \in Y \cup \{\varepsilon\})$ の実行によって状態が変化せず $(s_p = s_q)$，かつ，$(y \neq \varepsilon)$ の場合を考える．この場合も，基本的に $(y = \varepsilon)$ の場合と同様に出力の確認や出力後の状態 s_p の確認を行える．また，v を与えた状態組では A_i 以外の部分仕様は共通する入力 x を無視するようになっている．しかし，$(y \neq \varepsilon)$ の場合，I_i は状態 s_p に相当する状態で入力 x を無視する (x/ε) ように実装され，I_i 以外のある部分実装 I_j が入力 x に対して状態変化を起こさず y を出力するように実装されていた，という可能性がある．$(s_p = s_q)$ かつ $(y \neq \varepsilon)$ の場合，このような可能性は I_i の遷移の確認のみでは否定できない．しかし，その誤りの可能性は状態確認のときと同様，出力 y が ε に変わる可能性だけである．このような ε に変わる可能性がないことも，部分実装 I_j の遷移確認を行うことにより保証できる．すなわち，部分実装 I_j のすべての状態で同様に遷移確認を行うことにより，部分実装 I_j では，入力 x に対して A_j のみで使われる出力 z か，あるいは，ε のみしか出力されないことが保証できる．このため，I_j が A_i のみで使われる出力 y を出力することがないことも保証でき，結果として I_i が y を出力していたことが分かる．

以上の結果から，提案する手法で遷移の実装の正しさも保証できる．

6. あとがき

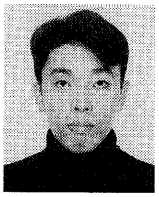
本研究では，並行に動作し，入力を奪い合う DFSM 群からなる通信プロトコルに対して，各 DFSM ごとに状態と遷移の確認を行うような適合性試験手法を提案した．また，我々は本手法を文献 4) のアブダカダブラプロトコルの Estelle 仕様を一部修正し簡単化した仕様に適用した．本来アブダカダブラプロトコルでは送信側から受信側への通信路は 1 つであるが，修正した仕様では 3 つの通信路を考え，送信プロセスを 3 個並行に動作させている．この仕様に対して 4 章の条件を満たす試験系列の生成が行えることを確認した（詳細は文献 2) 参照）．提案する手法では，一定の性質を持つ特性集合の存在を仮定しているが，アブダカダブラプロトコルなど，多くのプロトコルの仕様がこ

れらの条件を満足すると考えられる．FSM 間の通信や FSM の内部動作をともなう通信プロトコルを扱えるよう提案する手法を拡張することなどが今後の課題である．

参考文献

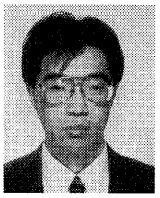
1) Chow, T.S.: Testing software design modeled by finite-state machines, *IEEE Trans. Soft. Eng.*, Vol.4, No.3, pp.178–186 (1978).
2) Fukada, A., Kaji, T., Higashino, T., Taniguchi, K. and Mori, M.: A Conformance Testing for Communication Protocols Modeled as A Set of DFSMs with Common Inputs, *Proc. 10th IFIP Int. Workshop on Testing of Communicating Systems* (*IWTCS'97*), pp.239–254 (1997).
3) Gonenc, G.: A method for the design of fault-detection experiments, *IEEE Trans. Comput.*, Vol.C-19, No.6, pp.551–558 (1970).
4) ISO: Information Technology – Open Systems Interconnection – Guidelines for the application of Estelle, LOTOS and SDL, ISO/IEC/TR 10167 (1991).
5) Lee, D., Sabnani, K.K., Kristol, D.M. and Paul, S.: Conformance testing of protocols specified as communicating FSMs, *Proc. IEEE INFOCOM'93*, pp.115–127 (1993).
6) Luo, G., Bochmann, G.v. and Petrenko, A.: Test selection based on communicating nondeterministic finite state machines using a generalized Wp-method, *IEEE Trans. Softw. Eng.*, Vol.20, No.2, pp.149–162 (1994).
7) Naito, S. and Tsunoyama, M.: Fault detection for sequential machines by transition tours, *Proc. 11th IEEE Int. Symp. on Fault Tolerant Computing* (*FTCS-11*), pp.238–243 (1981).
8) Sabnani, K.K. and Dahbura, A.T.: A protocol testing procedure, *Computer Networks and ISDN Systems*, Vol.15, No.4, pp.285–297 (1988).

(平成 9 年 8 月 11 日受付)
(平成 10 年 6 月 5 日採録)



深田 敦史

平成 10 年大阪大学大学院基礎工学研究科情報工学分野博士前期課程修了．工修．同年ヤマハ（株）勤務．在学中，通信プロトコルの適合性試験に関する研究に従事．



鍛 忠司

平成 8 年大阪大学大学院基礎工学研究科情報工学分野博士前期課程修了．工修．同年日立製作所勤務．在学中，通信プロトコルの適合性試験に関する研究に従事．

東野 輝夫（正会員）

昭和 54 年大阪大学基礎工学部情報工学科卒業．昭和 59 年同大学大学院博士後期課程修了．工博．現在，同大基礎工学部情報科学科助教授．平成 2 年，6 年モントリオール大学客員研究員．分散システム，通信プロトコル等の研究に従事．

谷口 健一（正会員）

昭和 40 年大阪大学工学部電子学科卒業．昭和 45 年同大学大学院博士課程修了．現在，同大基礎工学部情報科学科教授．工博．この間，計算理論，ソフトウェアやハードウェアの仕様記述・実現・検証の代数的手法および支援システム，関数型言語の処理系，分散システムや通信プロトコルの設計・検証法などに関する研究に従事．

森 将豪（正会員）

昭和 46 年名古屋工業大学工学部電子工学科卒業．昭和 48 年大阪大学大学院修士課程修了．現在，滋賀大学経済学部情報管理学科教授．工博．並行処理系の検証および代数的手法によるソフトウェアの設計に関する研究に従事．