

非同期通信仕様を表すプロトコル状態マシンからの 試験ケース生成方式の提案と評価

川崎 将人*1 大塚 亮*2 後沢 忍*3

三菱電機株式会社 情報技術総合研究所*1 *2 *3

1. 背景と従来研究

開発物の設計を行う際に、有限状態マシン(FSM: Finite State Machine)を利用するケースが多い。有限状態マシンは、状態の集合および状態遷移の集合で定義され、開発物の外部からの入力イベント系列によって変化する状態の系列を規定している。FSMにはさまざまな拡張が考案されており、UML(Unified Modeling Language, [1])1.0では遷移後の処理を表すアクションや遷移条件を状態遷移に付与できる。また開発物を外部から利用する手順仕様を表現するのにも利用され、UML 2.0では「プロトコル状態マシン」(PSM: Protocol State Machine)として規定している。PSMでは事前条件や事後条件を状態遷移に付与することができる。プロトコル状態マシンの記述形式として他にも文献[2]などがある。

一般的に、複雑なFSMからすべての遷移パスを網羅する試験ケース群を手作業で作成するのは、多くの工数を要するとともに、試験ケースの作成漏れの恐れがある。そこで、FSMから試験ケースを網羅的に自動生成する研究が旧来より行われている。その例として文献[3]がある。文献[3]では、次のようにして試験ケースを得ている。(1)試験ケース生成問題を定義、(2)試験ケース生成問題を充足可能性問題(SAT:Satisfiability problem)へ変換するアルゴリズムを規定、(3)FSMや試験対象となる状態遷移の集合などを(2)のアルゴリズムに入力し、和積形論理式fに変換、(4)和積形論理式fの充足可能性を判定、(5)充足可能であればf中の各変数に対する真理値割り当てを算出、(6)試験ケース生成問題に逆変換して試験ケースを得る。

ここで、文献[3]には次のような課題がある。

- ・ イベントが持つパラメータに対し、試験実行時の状況に適切して適切なパラメータ値を設定することができない。例：システム日時。結果、不適切な試験ケースが生成ならびに実行されることになり、非効率である。
- ・ 時間制約はタイムアウトイベントを用いて表

現。そのため試験ケース生成後に、人手による送受信時刻の入力作業が発生し非効率的。そこで本研究では、これらの課題を解決する試験ケース生成方式を提案する。

2. 試験ケース生成方式

提案する試験ケース生成方式の概要を図1に示す。

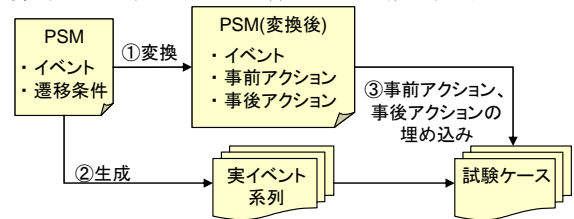


図1 試験ケース生成方式の概要

- ①入力となるPSMを変換。遷移条件を事前アクションおよび事後アクションに置換。
- ②入力となるPSMを元に、具体的なパラメータ値や送受信時刻を持つ実イベント系列を生成。尚、送受信時刻は試験開始時刻または先行する送受信イベントからの相対時刻で保持する。
- ③生成した実イベント系列に、事前アクションおよび事後アクションを埋め込み。

2.1. 入力：プロトコル状態マシン

入力とするPSMの遷移パスの例を図2に示す。

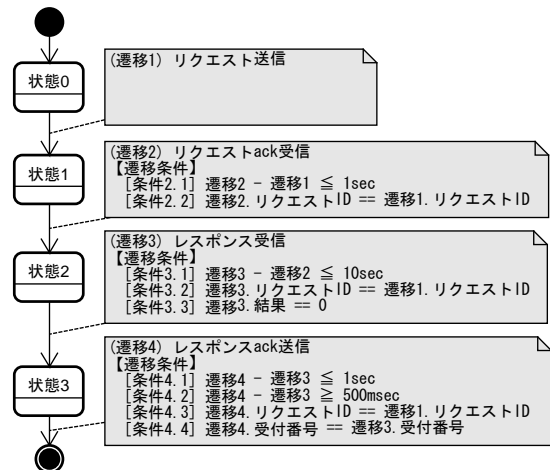


図2 プロトコル状態マシンの例

PSMのデータ構造は、UML 1.0のFSMのように状態遷移に遷移条件を付与する。遷移条件には制限を加え、パラメータ値制約(例:条件2.2)や時間制約(例:条件2.1)を記述するものとする。

Test Case Generation Method for Protocol State Machine.

*1 Masato Kawasaki

*2 Ryo Otsuka

*3 Shinobu Ushirozawa

Information Technology R&D Center,
Mitsubishi Electric Corporation

2.2. ①プロトコル状態マシンの変換

PSMの変換概要を表1に、変換後のPSMを図3に示す。尚、now() はシステム日時を表す。

表1 PSMの変換概要

	送信イベント	受信イベント
パラメータ値制約	[他遷移の事後アクション] パラメータ値の保存 [自遷移の事前アクション] パラメータ値の設定	[他遷移の事後アクション] パラメータ値の保存 [自遷移の事後アクション] パラメータ値の検査
最早時間制約	[他遷移の事後アクション] 時刻の保存 [自遷移の事前アクション] 所定時刻までの待機	[他遷移の事後アクション] 時刻の保存 [自遷移の事前アクション] 受信時刻の検査
最遅時間制約	[他遷移の事後アクション] 時刻の保存 [自遷移の事後アクション] 送信時刻の検査	[他遷移の事後アクション] 時刻の保存 [自遷移の事後アクション] 受信時刻の検査

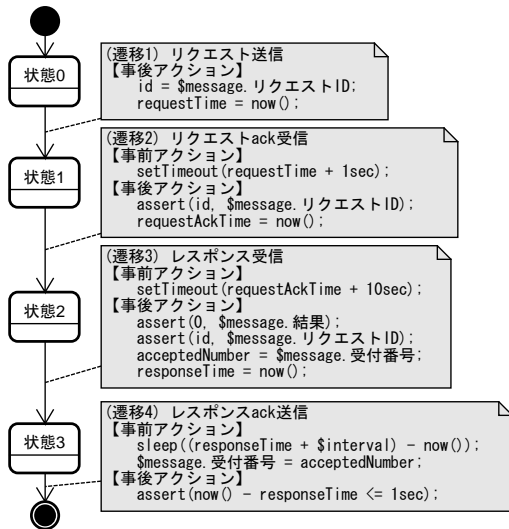


図3 変換後のプロトコル状態マシン

2.3. ②実イベント系列の生成

生成した実イベント系列を図4に示す。

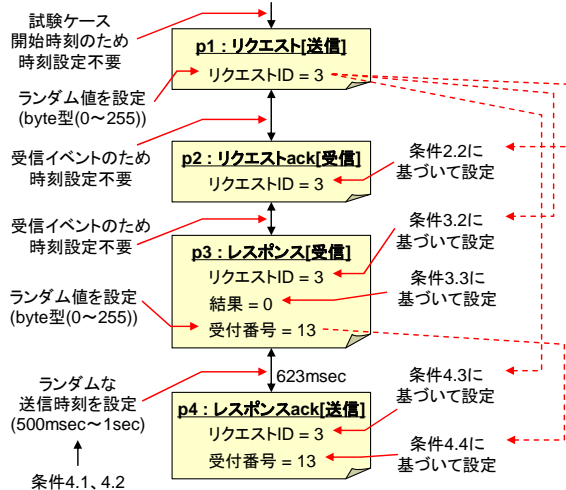


図4 生成した実イベント系列の例

図4では、パラメータ値や送信時刻に乱数値を設定する場合の試験ケースを示しており、p4のp3からの相対的な送信時刻が明らかにされている。

2.4. ③事前/事後アクション埋め込み

図4のイベント系列に、事前アクションと事後アクションを埋め込んだ試験ケースを図5に示す。

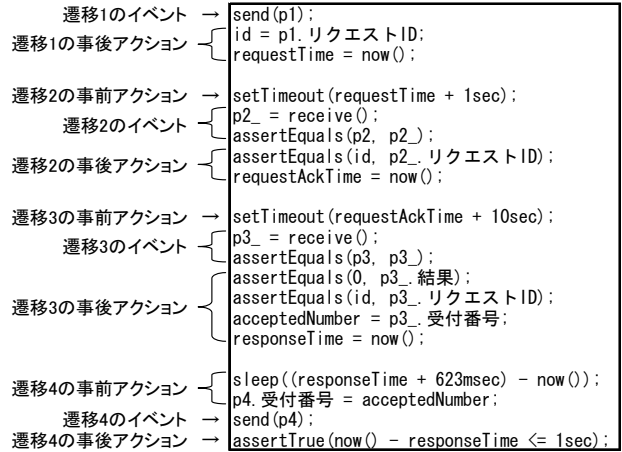


図5 試験ケースの例

図5はJUnit([4])に類似の擬似コードを用いて記述している。また送受信パケットp1~p4は図4のものと同じものを指している。図5では、事前アクションや事後アクションの内容を実イベントに変換し、試験ケースを生成している。

図5を見ると、受信イベントp3_の受付番号を、送信イベントp4に設定する動作が含まれる。

このように生成した試験ケースでは、試験実行時の状況に適したパラメータ値を設定可能であり、具体的な送受信時刻の入力が不要となっており、試験作業の効率化が期待できる。

3. まとめと今後の課題

本研究では、プロトコル状態マシンの遷移条件にパラメータ値制約や時間制約の制限を加え、そのプロトコル状態マシンから生成した実イベント系列と、遷移条件に対応した事前アクションおよび事後アクションを組み合わせることで、試験ケースを生成する。

生成した試験ケースは、イベントのパラメータ値が実行時の状況によって決まる場合にも適応でき、また具体的なイベント送受信時刻の入力が不要となっており、試験作業の効率化が期待できる。

今後は、生成された試験ケースにより実システムの試験を行い、その効果を評価する。

4. 参考文献

[1] Object Management Group -UML-, <http://www.uml.org/>
 [2] Luca de Alfaro et al., "Interface automata", ACM SIGSOFT Software Engineering Notes, 2001
 [3] 森 亮憲ら, "通信プロトコルの試験系列生成問題に対する SAT アルゴリズム適用法の提案", 電子情報通信学会論文誌 Vol. J85-D-I No. 11 pp. 1038-1046, 2002
 [4] JUnit.org, <http://www.junit.org/>