拡張有限状態機械でモデル化したOSIセションプロトコルの検証

4V−2

佐野 哲央† 樋口 昌宏† 関 浩之† 嵩 忠雄‡

†大阪大学基礎工学部情報工学科 ‡奈良先端科学技術大学院大学情報科学研究科

1 まえがき

筆者らは, プロトコル機械が有限制御部と非負整数値をとる有限個のレジスタを持つ Extended Communicating Finite State Machines(ECFSM) を用いてモデル化された通信プロトコルの安全性の検証法を提案しており, その検証法に基づく検証システムを試作している[3]. 提案している検証法が実用プロトコルの安全性の検証に有効であることを示すために, OSI セションプロトコルのカーネル, 全二重, 大・小・再同期機能単位を規定するプロトコルを取り上げ, 上述の検証システムを使用してその安全性の検証を行った. 本稿では特にプロトコル全体の安全性を複数のサブプロトコルの安全性の検証に帰着した点について述べる.

2 プロトコルモデルと検証法

2.1 プロトコルモデル

プロトコル機械 PM を次のような 4 字組 (S, Σ, δ, SI) で表される拡張有限状態機械で定義する. ここで $S = \langle SF, r\rangle$ はプロトコル機械の状態集合を定義する 2 字組で, SF は有限制御部の取り得る状態の有限集合, r は非負整数値を保持するレジスタ数を表す. $\Sigma = \Sigma_- \cup \Sigma_+$ はメッセージタイプの有限集合であり, $\Sigma_-(\Sigma_+)$ は送信(受信) メッセージタイプの集合, メッセージはメッセージタイプと非負整数のパラメータ値からなる. また δ は $SF \times \mathcal{N}^r \times E_v$ から $SF \times \mathcal{N}^r$ への非決定性状態遷移関数 (但し, E_v は $\{-\langle d, n\rangle | d \in \Sigma_-, n \in \mathcal{N}\}$ と $\{+\langle d, n\rangle | d \in \Sigma_+, n \in \mathcal{N}\}$ の和集合), $SI \subseteq SF \times \mathcal{N}^r$ は初期状態の集合である.

プロトコル機械 $PM_A = (\langle SF_A, r_A\rangle, \Sigma_A, \delta_A, SI_A)$ 及び $PM_B = (\langle SF_B, r_B\rangle, \Sigma_B, \delta_B, SI_B)$ に対し, $\Sigma_{A-} = \Sigma_{B+}$ かつ $\Sigma_{B-} = \Sigma_{A+}$ (Σ_{AB}, Σ_{BA} と表記する) であるとき, $\Pi = (PM_A, PM_B)$ をプロトコルと呼ぶ. また, 4 字組 $gs = \langle s_A, s_B, ch_{BA}, ch_{AB}\rangle \in (SF_A \times \mathcal{N}^{r_A}, SF_B \times \mathcal{N}^{r_B}, \langle \Sigma_{BA}, \mathcal{N}\rangle^*, \langle \Sigma_{AB}, \mathcal{N}\rangle^*)$ を Π の**合成状態**といい, $gs_I = \langle si_A, si_B, \epsilon, \epsilon\rangle (si_A \in SI_A, si_B \in SI_B)$ を Π の**初期合成状態**と呼ぶ.

プロトコル Π に対して, 初期合成状態から有限回の状態遷移により到達可能な合成状態の集合を Π の**可達集合**という. Π の可達集合がデッドロック及び未定義受信状態を含まないとき, Π は**安全性**を満たすという[3].

2.2 検証法と検証システム

このようにモデル化されたプロトコルにおいて, 検証すべき条件を表現するために, (AF1) プロトコル機械の有限制御部の状態が指定した値をもつかどうかを表す式, (AF2) 通信路上のメッセージ系列のメッセージタイプ列が指定した系列集合に属するかどうかを表す式, (AF3) 先の AF2 の式の表すメッセージ列に含まれるパラメータ列についての述語, 例えば "step1(AB)" は PM_A から PM_B への通信路上の系列が増分 1 の単調増加列であるかどうかを表している, (AF4) 通信路内のメッセージのパラメータと, プロトコル機械のレジスタとの関係を表す線形の等式・不等式, の 4 つの型の原子式を導入する. 検証すべき条件は以上を原子式とする積和形論理式として記述する.

初期合成状態から到達可能な任意の合成状態において, 論理式 F が成り立つとき F を不変式という. 我々が提案している検証法は, 検証者が不変式と思われる論理式 F を記述し, F が実際に Π の不変式であることを送受信イベントに関する構造的帰納法によって示し, その論理式を満たす合成状態集合が未定義受信及びデッドロック状態を含まないことを示すことにより, プロトコル Π が安全性を満たすと結論するものである. 検証システムは, プロトコル機械の定義と論理式 F を入力とし, 帰納段階の証明過程を自動化したものである.

3 プロトコルの合成手法

実用プロトコルである OSI セションプロトコルは, 様々なサービスが規定されている. そのため規模も非常に大きく, 直接その検証を行うのは適切ではない. よってプロトコルを複数のサブプロトコルに分解して検証を行った. サブプロトコルへの分解は, それらの合成の逆操作と考えられる. 以下では利用した 2 つの合成手法について述べる.

3.1 フェーズ合成

有限状態機械でモデル化された通信プロトコルに対して提案された手法であり, フェーズと呼ばれる性質をもつプロトコルを縦列に接続して, 各フェーズ間を順次推移するプロトコルを合成するものである[1]. 以下これについて簡単に説明する.

定義 3.1

1. プロトコル機械 PM において状態 s からいかなる遷移も定義されていないとき, s は**最終状態**であるという.
2. 以下の条件を満たすとき, プロトコル Π が**適切終了**するという. Π の到達可能合成状態 $\langle s_A, s_B, x, y\rangle$ に対して, s_A(または s_B) がプロトコル機械 PM_A(または PM_B) の最終状態であるならば $x = \epsilon$(または $y = \epsilon$) であり, プロトコル機械 PM_B(または PM_A) は状態 s_B から y(または s_A から x) の受信動作のみを行って最終状態 s'_B(または s'_A) へ到達する.
3. プロトコル Π の到達可能合成状態 $\langle s_A, s_B, \epsilon, \epsilon\rangle$ は状態 s_A 及び s_B が最終状態であるとき, **適切終了状態**と呼ぶ.
4. プロトコル Π が適切終了し, かつ未定義受信フリーかつ適切終了状態を除く状態においてデッドロックフリーであるとき, Π は **p-安全**であるという.
5. Π を p-安全なプロトコル, 状態 s_A, s_B をそれぞれ PM_A, PM_B の最終状態とする. 2 字組 (s_A, s_B) は, 合成状態 $\langle s_A, s_B, \epsilon, \epsilon\rangle$ が到達可能なとき, **終了状態**と呼ぶ.
6. p-安全なプロトコル Π は, プロトコル機械 PM_A または PM_B におけるすべての最終状態が終了状態集合にちょうど一回だけ現われるときフェーズと呼ぶ.

[フェーズ合成によるプロトコルの構成]

プロトコル $\Pi_\alpha = (PM_{A_\alpha}, PM_{B_\alpha})$, $\Pi_\beta = (PM_{A_\beta}, PM_{B_\beta})$ をそれぞれ終了状態集合が E_α, E_β であるようなフェーズとし, C を E_α の部分集合とする. C によって Π_α と Π_β を合成したプロトコル $\Pi = (PM_A, PM_B)$ を以下のように定義し, $< \Pi_\alpha, C, \Pi_\beta >$ で表す.

A Verification for OSI Session Protocol Modeled as Extended Communicating Finite State Machines

Tetsuo SANO†, Masahiro HIGUCHI†, Hiroyuki SEKI†, Tadao KASAMI‡

†Osaka University, ‡Advanced Institute of Science and Technology, Nara

PM_Aは, Cに属する組 (s_A, s_B) の第1成分 s_Aを, PM_{A_β}の初期状態と一致させることによって, PM_{A_α}と PM_{A_β}から構成されるプロトコル機械であり, 初期状態は PM_{A_α}の初期状態とする. PM_Bについても同様に定義する.

定理 3.1 プロトコル$\Pi_\alpha = (PM_{A_\alpha}, PM_{B_\alpha})$ 及び$\Pi_\beta = (PM_{A_\beta}, PM_{B_\beta})$ をそれぞれ終了状態集合が E_α, E_βであるような二つのフェーズとし, Cを E_αの部分集合とする. このとき合成プロトコル $\Pi = (PM_A, PM_B)$ は, 終了状態集合が $(E_\alpha \cup E_\beta) - C$であるようなフェーズである.

[ECFSM への拡張]

以下では, 上述のフェーズ合成を ECFSM モデルに拡張した結果について説明する.

PM_Aは $f : SF_{A_\alpha} \rightarrow SF_{A_\beta}$で $\{f(s_A) | \exists_{s_B}\{(s_A, s_B) \in C\}\} \supseteq SI_{A_\beta}$を満たす写像 fが存在するとき, $f(s_A) \in SI_{A_\beta}$なるすべての s_Aを $f(s_A)$ と一致させることによって PM_{A_α}と PM_{A_β}から構成されるプロトコル機械であり, 初期状態は PM_{A_α}の初期状態とする. PM_Bについても同様に定義する. この合成によって ECFSM においても定理 3.1と同様の結果が得られる.

3.2 優先サービスと通常サービスの合成

文献 [2] では, プロトコルを優先サービス, 通常サービスそれぞれを定義した2つのサブプロトコルの合成によって得られるものと考え, その合成方法を定義し, 次に定義した合成方法に従ってプロトコルを合成したときに, 合成前の各サブプロトコルで成立していた安全性 (未定義受信及びデッドロックフリー性) が合成後のプロトコルにおいても保存されるための十分条件を示している.

4 検証対象とするプロトコル

検証の対象として OSI セションプロトコル[4]の主要部であるカーネル, 全二重, 大・小・再同期機能単位のコネクション確立, データ転送, コネクション解放フェーズを取り上げた.

以下は, 大・小・再同期機能単位の簡単な説明である. ここで同期点とは転送されるデータ間に付与される一連番号であり, 送受信の確認及び再送の際の目印として利用される. またその際に用いられるレジスタを表1に示す.

小同期機能単位 一方の利用者が, 転送する一連のデータ間に同期点設定を要求できる. 確認の応答を返すか否かは通知を受けた側に任せられる.

大同期機能単位 一方の利用者が, 転送する一連のデータ間に同期点設定を要求できる. 通知を受けた側は確認の応答を返す必要がある.

再同期機能単位 同期点番号を再度設定し直すサービス. 再設定のオプションとして, 放棄, 再始動, 設定の3つがある.

[OSI セションプロトコルのサブプロトコル化]

OSI セションプロトコルを機能毎にサブプロトコルとして定義し (図1), それらを3で述べた合成手法により合成したプロトコル $\Pi_{SESSION}$が, カーネル, 全二重, 大・小同期, 再同期の機能単位を規定するプロトコルに一致することを確認した.

表 1: 大・小・再同期機能単位で利用するレジスタ (一部)

レジスタ	役割
Vm	次に用いる同期点番号を示している
Va	同期確認が期待される同期点番号を示している
Vr	再同期可能な同期点番号を示している

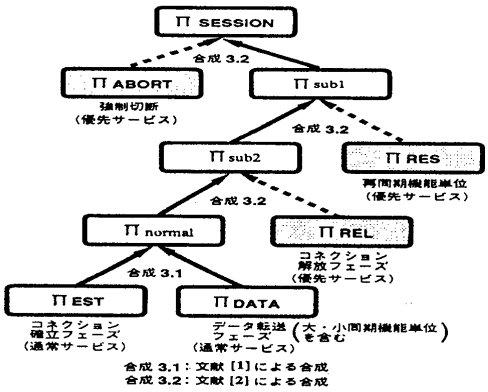

図 1: $\Pi_{SESSION}$の合成過程

表 2: 検証結果

	不変式の積項数	CPU 時間 (秒)	メモリ使用量 (MB)
Π_{EST}	7	0.08	0.056
Π_{DATA}	153	98.27	10
Π_{REL}	8	0.17	0.152
Π_{RES}	72	6.78	0.832
Π_{ABORT}	7	0.08	0.040
$\Pi_{SESSION}$		合計 105.38	最大 10

5 検証結果とまとめ

表2に検証システムを用いた各サブプロトコルの検証結果を示す.

本検証で最も多くの手間を要したΠ_{DATA}の検証の際に検証者が記述した論理式が最終的に不変式となるまでに要した変更回数は 43 回, 不変式の積項数は 153 項, 作業時間は延べ 27.3 時間であった. 不変式記述において, 検証システムの出力の利用により機械的作業は軽減されるが, 本質的にはプロトコルの内容に関する理解が不可欠であった.

また本検証に要した総 CPU 時間は約 105 秒, 最大メモリ使用量は 10MBytes と実際的な値で検証が可能であり, 我々の提案する検証法とプロトコルの合成手法を併用することにより, OSI セションプロトコルのような実用プロトコルの検証が可能であることが確認された. 今回検証対象としたプロトコルは, Major-Token 及び Minor-Token の2つのトークンを利用しているが, これらのトークンを一つにまとめたプロトコルの検証[3]と比較して, CPU 時間で9倍, メモリ使用量で 10 倍, 不変式の総積項数は5倍となっている. さらに折衝解放機能単位が利用可能となる場合には, Release-Token を加えた3つのトークンを利用することになり, 検証が飛躍的に煩雑になることが予想される. 現在これらトークンを利用するプロトコルの検証の煩雑化を軽減するための手法について検討中である.

参考文献

[1] C. Chow, et al.: "A Discipline for Constructing Multiphase Communication Protocols", ACM Trans. CS, vol.3, pp.315-343, 1985-11.

[2] M. Higuchi, et al.: "A Method of Composing Communication Protocols with Priority Service", to appear in IEICE Trans. Commun., 1992-10.

[3] M. Higuchi, et al.: "A Verification Procedure via Invariant for Extended Communicating Finite-State Machines", Proc. of 4th Intern. Workshop on Computer Aided Verification, pp.359-370, 1992-07.

[4] ISO: "Basic Connection Oriented Session Protocol Specification", ISO 8327.