確率時間オートマトンを用いた汎用結合可能性を有する暗号プロトコルの設計手法の構築

山　根　智　　金沢大学大学院自然科学研究科電子情報科学専攻

1 はじめに

本研究は、堅牢な暗号プロトコルの設計検証手法の構築を行うものである。具体的な手法は、計算論的アプローチと R.Canetti の“汎用結合可能性(Universal Composability)の概念” [1]とを統合化して、理想機能と現実機能をそれぞれ確率時間オートマトンで仕様記述して、現実機能が理想機能を確率時間模倣するかどうかを自動検証する。これにより、現実機能が正しく理想機能を実現していることを保証する手法を開発した。研究内容の概要を以下に説明する。

【研究内容】

以上を実現するために、我々は、本研究で以下の成果を上げた。

（1） モデル化

暗号プロトコルの理想機能と現実機能の個々の構成要素を確率時間オートマトンでモデル化する。理想機能と現実機能はそれぞれ確率時間オートマトンの並列合成としてモデル化する。

（2）仕様記述言語

暗号プロトコルの理想機能と現実機能を仕様記述するために、以下の言語を定義した。

① 時間オートマトンを離散確率分布で拡張して、確率時間模倣検証を形式化するために、ロケーションに離散確率を割り付けた。

② 入力アクションと出力アクションで同期するように言語を拡張して、理想機能などを記述できるようにして、並列合成演算を定義した。

（3）確率時間模倣検証

汎用結合可能性における区別されないという概念を確率時間模倣関係で表現して、自動検証を実現した。現実機能が理想機能を確率時間模倣すれば、現実機能は正しく理想機能を実現しており、正しい暗号プロトコルと見なす。確率時間模倣関係は確率模倣と時間模倣を融合した新しい関係である。

（4）検証器の実装

C++言語で約 9000 行で確率時間模倣検証器を実装して、簡単な暗号プロトコルで実証実験をした。確率時間模倣検証器は、確率時間オートマトン解析器と検証器などから構成される。

また、本研究は類似研究に対して、以下のように新規性がある。

【類似研究との比較】

（1）MIT の R.Canetti や N.Lynch などは確率 I/O オートマトンを用いて演繹的に理想機能と現実機能を仕様記述して演繹的に検証する手法を提案している。[2]我々の手法は、確率時間オートマトンを用いてリアルタイム性を考慮しており、自動検証を実現している点で新規性がある。しかし、自動検証を実現するために、確率時間オートマトンのロケーションを有限に制限しているために、乱数発生に制限が加えられる。

（2）オックスフォード大学の M. Kwiatkowska らが確率時間オートマトンのモデル検査器[3]及び記号モデル検査器[4]を開発している。モデル検査器と我々の確率時間模倣検証器とでは、おおいに異なる。

また、以下のように、今後の課題があり、現在、継続して研究している。

【今後の課題と現在進行中の継続研究】

（1）現実機能が理想機能と区別されないという概念は難しい概念であり、正確には、“計算論的に、ある確率以下で区別されない”というものである。本研究で提案した確率時間模倣は少し強すぎるので、現在、“計算論的に、ある確率以下で区別されない”を検討している。

（2）自動検証するためには、乱数の概念を弱める必要がある。乱数の概念を弱めないで、自動検証するためには、無限のロケーションを持つシステムとして記述する必要があり、演繹的に検証せざるをえない。そこで、現在、抽象化して有限化する自動演繹的検証手法を検討している。これは、述語抽象化精錬検証手法である。

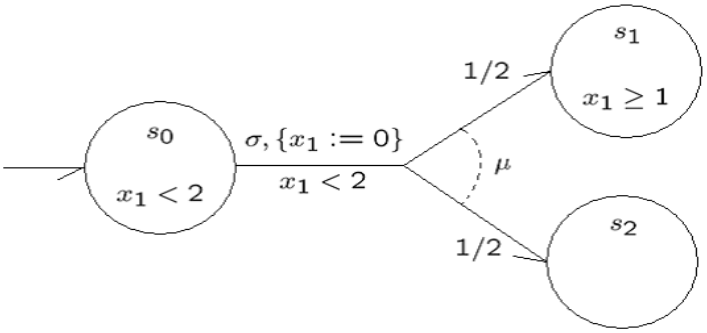
（3）理想機能と現実機能との間には、大きなギャップがあり、理想機能からゲーム変換を行いながら、模倣検証により信頼性を保証する必要がある。このために、我々は確率時間ゲーム理論を研究している。

2 堅牢な暗号プロトコルの設計検証手法の構築

2-1 モデル化と仕様記述言語

暗号プロトコルの理想機能と現実機能の個々の構成要素を確率時間オートマトンでモデル化する。理想機能と現実機能はそれぞれ確率時間オートマトンの並列合成としてモデル化する。

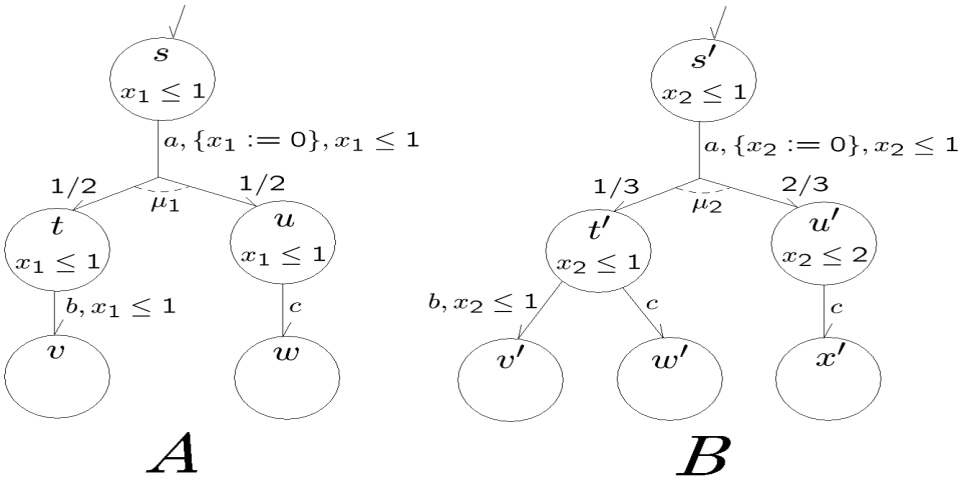
まず、確率時間オートマトンを図示する。紙面の都合上から、形式的な定義は文献に譲る。



ここで、確率時間オートマトンは、ロケーション s0, s1, s2、クロック変数 x1、離散確率分布 μ で定義される。システムはロケーション s0 に x1<2 を満たす間は留まり、その後、枝に付いている遷移条件を満たすならば、確率分布に従って離散遷移する。この例では、s1 の不変条件が x1＞＝1 なので遷移できない。これを x1＞＝0 と変更すると、遷移できる。

2-2 確率時間模倣検証

暗号プロトコルの理想機能と現実機能のすべての構成要素を確率時間オートマトンで仕様記述する。以下では、確率時間模倣を事例で説明する。



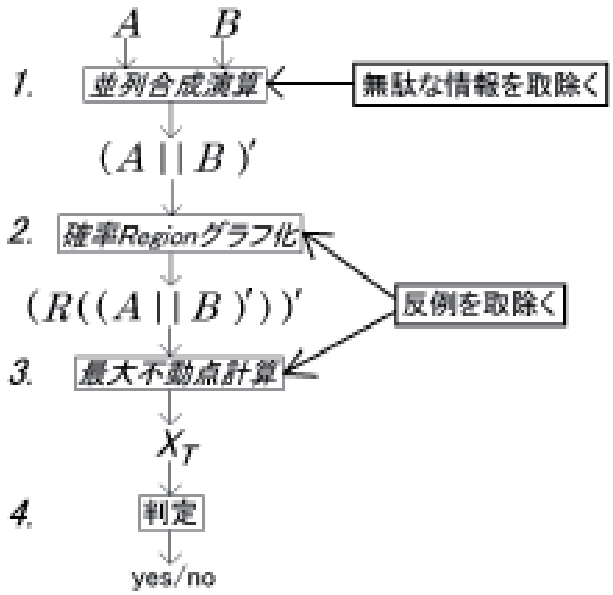
確率時間オートマトンＡはＢを確率時間模倣する。Ａは確率 1/2 で ab を受理する。一方、Ｂもロケーション ｔ´ から出ている２つの遷移の重みを調整することによって、確率 1/2 で ab を受理するようにできる。この重みの調整はアルゴリズムで計算できるので、確率時間模倣検証アルゴリズムは存在する[5]。

3　確率時間模倣検証の実験

3-1 検証器

我々は C++言語で約 9000 行で検証器を実装して，コンパイラは gcc version 3.2.1 を用いた。また、検証器は Sun Blade 1000(CPU UltraSPARC-Ⅲ 900MHz，主記憶 1024MB)上の Solaris 8 オペレーティングシステム上に実装した。

検証器は確率時間模倣を満たさない反例を検出しながら、確率時間模倣関係を最大不動点として計算するものである。確率時間オートマトンＡがＢを確率時間模倣するかどうかを以下のように検証する。



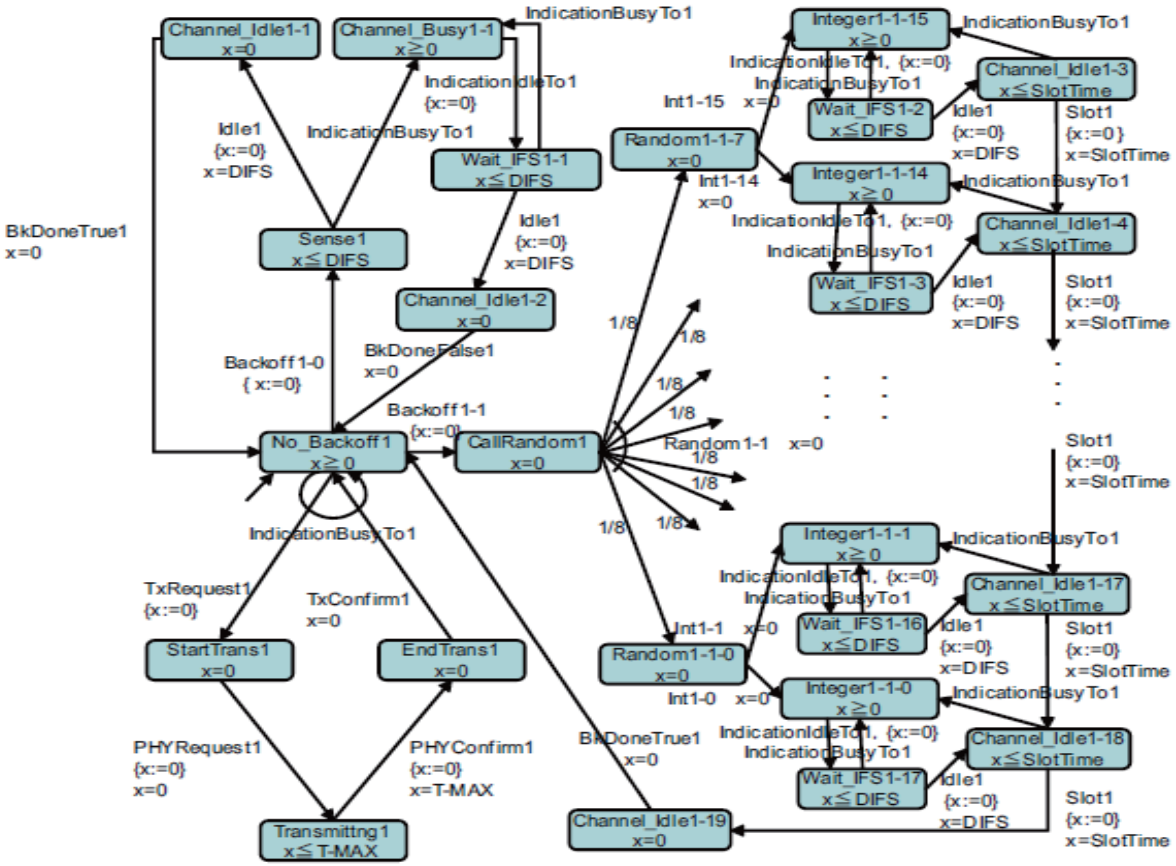
3-2 モデル化と仕様記述、検証

ワイヤレス LAN に関するプロトコルを事例として仕様記述と検証の実験を行った。

理想機能と現実機能はそれぞれ 7 個の確率時間オートマトンの並列合成としてモデル化した。検証実験の所要メモリと所要時間を UNIX の PS コマンドで計測し，得られた計測値を表に示す。選択幅とは、確率分布であり、例えば、選択幅 4 とは 1/4 が 4 本あることを意味する。所要時間は検証時間であり、所要メモリは検証メモリであり、状態数と遷移数は検証器の内部で生成されたオートマトンの大きさである。選択幅に大きさに比例して、指数的に検証コストが増大した。大規模なシステムを検証するためには、抽象化テクニック等が必要である。

選択幅	所要時間	所要メモリ	状態数	遷移数
16	12 時間 52 分	874.30MB	29,231	25,030
8	3 分 20 秒	80.62MB	4,897	5,020
4	7 秒	11.35MB	1,068	1,308
2	1 秒	3.18MB	336	468

また、仕様記述例を示す。



4　まとめ

本研究は、堅牢な暗号プロトコルの設計検証手法の構築するために、計算論的アプローチと R.Canetti の“汎用結合可能性(Universal Composability)の概念”とを統合化して、理想機能と現実機能をそれぞれ確率時間オートマトンで仕様記述して、現実機能が理想機能を確率時間模倣するかどうかを自動検証する手法を提案した。まず、暗号プロトコルの理想機能と現実機能の個々の構成要素を確率時間オートマトンでモデル化する。理想機能と現実機能はそれぞれ確率時間オートマトンの並列合成としてモデル化した。次に、暗号プロトコルの理想機能と現実機能を仕様記述するために、時間オートマトンを離散確率分布で拡張して、確率時間模倣検証を形式化するために、ロケーションと実数空間に離散確率を割り付けた。次に、汎用結合可能性における区別されないという概念を確率時間模倣関係で表現して、自動検証を実現した。現実機能が理想機能を確率時間模倣すれば、現実機能は正しく理想機能を実現しており、正しい暗号プロトコルと見なす。最後に、C++言語で約 9000 行で確率時間模倣検証器を実装して、簡単な暗号プロトコルで実証実験をして、その有効性を示した。

【参考文献】

[1]Ran Canetti: Universally Composable Security: A New Paradigm for Cryptographic Protocols. FOCS 2001: 136-145(2001).

[2] Ran Canetti, Ling Cheung, Dilsun Kirli Kaynar, Moses Liskov, Nancy A. Lynch, Olivier Pereira, Roberto Segala: Analyzing Security Protocols Using Time-Bounded Task-PIOAs. Discrete Event Dynamic Systems 18(1): 111-159 (2008)
[3]Marta Z. Kwiatkowska, Gethin Norman, Roberto Segala, Jeremy Sproston: Automatic verification of real-time systems with discrete probability distributions. Theor. Comput. Sci. 282(1): 101-150 (2002)
[4]Marta Z. Kwiatkowska, Gethin Norman, Jeremy Sproston, Fuzhi Wang: Symbolic model checking for probabilistic timed automata. Inf. Comput. 205(7): 1027-1077 (2007)
[5] Satoshi Yamane: Probabilistic Timed Simulation Verification and Its Application to Stepwise Refinement of Real-Time Systems. ASIAN 2003: 276-290(2003)

〈発　表　資　料〉

題　　名	掲載誌・学会名等	発表年月
確率時間オートマトンの確率強模倣検証アルゴリズムの実現	電子情報通信学会技術報告書	2007 年 4 月
確率時間オートマトンの確率時間強模倣検証器の開発	コンピュータソフトウェア	2008 年（印刷中）
確率時間ゲーム理論による組込みシステムの形式的検証	日本ソフトウェア科学会大会	2007 年 9 月
述語抽象化とその洗練による確率時間オートマトンの到達可能性解析手法	電子情報通信学会技術報告書	2008 年 6 月（印刷中）