

Title	時間オートマトンから時間モジュールへの変換に関する考察
Author(s)	館, 宜伸
Citation	
Issue Date	2001-03
Type	Thesis or Dissertation
Text version	author
URL	http://hdl.handle.net/10119/1485
Rights	
Description	Supervisor:落水 浩一郎, 情報科学研究科, 修士

時間オートマトンから時間モジュールへの変換に関する考察

館 宜伸

北陸先端科学技術大学院大学 情報科学研究科

2001 年 2 月 15 日

キーワード: 時間オートマトン、時間モジュール、実時間システム、ソフトウェアプロセス.

1 本論文の背景

通信システム、論理回路、オペレーティングシステムといった実時間システムの大規模化や多様化にともない、これらのシステムの設計段階における信頼性の検証が重要になってきている。これらの大規模なシステムでは、論理的動作の信頼性だけでなく、動作のタイミングの信頼性も要求されており、シミュレーションによるテストは困難である。そこで、設計されたシステムを数学的手法により信頼性を検証できることが望ましい。このような背景のもとで、実時間システムの仕様記述および検証のために時間オートマトン [1] および時間モジュール [2] が提案されている。前者は、仕様記述の分野でよく知られているオートマトンについて、その枝に時間制約や時間のリセットを付加し拡張したものである。後者は、ブール変数の値の遷移を式で表現するリアクティブモジュールに、時間に関する遅延や時間制約等の概念を付加し拡張したものである。この拡張により、真偽値に関することと時間に関することを、変数値の遷移という同一の枠組で扱うことが可能となった。ところで、Alur らは、文献 [2] において、時間モジュールが時間オートマトンの一般化であるということを示唆してはいるが、後者から前者への具体的な変換手続きは与えていない。

2 時間オートマトンから時間モジュールへの変換

そこで、本論文では、時間オートマトンから時間モジュールへの変換について考察した。この変換は以下の2つの部分からなる。

- 時間オートマトンから時間モジュールへの等価変換
- 等価変換によって得られた時間モジュールの最適化

これらの2つの手法をそれぞれ提案し、さらに、その正当性の証明を行った。なお、本論文では、変換対象を時間I/O オートマトンとする。これは、時間オートマトンのイベントに入力と出力の区別を付加し拡張したものである。対象を時間I/O オートマトンとする理由は、本論文で記述の対象とするシステムやプロセスでは、イベントにおける入力と出力の区別が必要だからである。

この変換を、本論文では、以下の2つの例について適用した。

- 実時間システムの例である、列車制御システムの仕様
- 実時間システム以外の例である、ソフトウェアプロセス記述

ソフトウェアプロセス記述の例題としては、Kellner らによる「ソフトウェアプロセスモデリングのための例題」[3] が有名であり、本論文もこれを例にとる。ソフトウェアプロセスを時間I/O オートマトンで記述する試みは、本論文の特徴である。

そして、列車制御システムとソフトウェアプロセスに対して、時間I/O オートマトン記述と変換後(最適化後)の時間モジュール記述の比較に関する考察を行う。時間I/O オートマトンのイベントによる遷移を、時間モジュールの変数の値の変化で表現するときに、等価変換において追加式と追加条件を付加しているため、遷移の数は増えてしまう。ここで、時間モジュールでは、複数の変数値による条件と、複数の変数の値の更新が可能である。最適化では、複数の入力や複数の出力を一つの更新式で表現し、遷移の数を減らすことができた。考察の結果、時間モジュールの方が、等価変換後の時間モジュールより変数 state の数、更新式の数が少なく記述できることがわかった。

3 今後の展望

本論文では、列車制御システムの仕様と Kellner らの例題仕様のソフトウェアプロセスの記述に変換を適用した。他の実時間システムの仕様や、他のソフトウェアプロセスを、時間I/O オートマトンで記述し、本論文の等価変換と、最適化が適用可能であるかどうかは、まだ確認していない。一つ目の課題として、他の実時間システムの仕様や、他のソフトウェアプロセスへの適用を挙げる。

また、上記の適用が可能でない場合、どのような最適化が可能となるのか、また、その最適化は、今まで適用可能であったものに対しても可能かどうかの確認も必要である。こ

こで、二つ目の課題として、本論文の最適化が適用不可能な例を見つけ出し、それに対する最適化法の開発、本論文の適用例への新しい最適化法の適用の確認、を挙げる。

ソフトウェアプロセス記述に関する最適化として、次のことを検討する予定である。本論文では、ソフトウェアプロセスへの適用において、各部分プロセス毎に記述を行った。しかし、実際には、一人の開発者がこれらの部分プロセスを複数兼任していることがある。このため、部分プロセス間の通信で不要になる箇所も生じてくる。部分プロセスの合成方法は、その開発者がどの部分プロセスとどの部分プロセスの作業を兼任しているかによって異なってくる。このように、どのように兼任していても部分プロセスの合成が行える最適化法の開発を三つ目の課題とする。