

線形拘束オートマトンを用いた組込みソフトウェアの仕様記述に関する考察

2000MT012 二口 裕介 2000MT019 日紫喜 直幸 2000MT110 横井 崇人
指導教員 野呂 昌満

1 はじめに

旧来、組込みソフトウェアは系統的な設計がされていなかった。系統的な開発方法とは、システムの仕様を厳密に定義、記述することができ、あいまいさのない設計をすることが可能な開発支援の方法のことである。近年、組込みソフトウェアの開発の大規模化に伴い組込みソフトウェアの系統的な開発方法が必要とされている。一般に、有限オートマトン (Deterministic Finite Automaton, 以下 DFA) のモデルを用いて設計することがよく知られている。しかし、DFA を用いて記述を試みても、対象によっては状態数が組合せ爆発を起こす。

本研究の目的は、組込みソフトウェアをオートマトン [1][2] のモデルを用いて系統的に記述することである。本研究では入力記憶を必要とするような組込みソフトウェアを対象とする。これを DFA で記述しようとする、状態数が組合せ爆発を起こしてしまう。

解決策として、入力を記憶する必要のある組込みソフトウェアに線形拘束オートマトン (Linear Bounded Automaton, 以下 LBA) を適用する。組込みソフトウェアを一つの状態遷移機械で表現すると複雑になってしまうので、構成要素に分割しそれぞれを状態遷移機械で記述する。記憶を必要とする部分に LBA を、それ以外の部分に DFA を適用する。対象とする組込みソフトウェアは、記憶された入力記号列に対して検索、追加や削除など複数の異なる動作を行わなくてはならない。構造を整理するために状態遷移機械を階層化する。

我々の提案する手法を入力記憶を必要とする組込みソフトウェアに適用することで、系統的に記述することが可能となった。エレベータと携帯電話を事例として LBA の適用手順を説明し、有用性と他の組込みソフトウェアへの適用可能性を検証する。本研究では状態遷移を理論的に表現したものをオートマトンと呼び、それを個々の組込みソフトウェアに適用したものを状態遷移機械と呼ぶ。

二口は状態遷移機械の分割、日紫喜は LBA の階層化、横井は他の組込みソフトウェアへの適用を主に担当した。

2 組込みソフトウェアとオートマトン

我々は、入力を記憶し、記憶を任意に参照することのできる組込みソフトウェアの表現に適しているオートマトンを用いることにする。オートマトンは記述能力によってクラス分けができる。一般に、記述能力が高いオートマトンを用いると複雑な計算が可能であるが、その複雑さに起因して、記述や理解が難しくなる。各オートマトンの特性と、エレベータに適用した場合を考える。

DFA は、現在の状態とイベントで次の遷移先が決定される。しかし入力の記憶を用いた遷移をすることができないので、ボタンの入力を記憶する必要のあるエレベータには適用できない。

プッシュダウンオートマトン (Pushdown Automaton, 以下 PDA) は、プッシュダウンスタックを用いることで入力の記憶が可能となる。記憶の操作の仕方が先入れ後出しに限定されるので、ボタンの入力の記憶を任意に参照する必要があるエレベータには適用できない。

LBA は、有限の記憶テープと、テープを読み書きできる有限状態制御部で構成されている。テープを任意の位置で読み書きできるので、複数の入力を記憶し、入力された順序に関係なく任意に参照する必要があるエレベータに適用可能である。

チューリング機械 (Turing Machine, 以下 TM) は、LBA の有限の記憶テープを無限の記憶テープにしたものである。無限の記憶を扱う事が可能で記述能力が高くなるが、停止問題が決定不能な問題になる。

エレベータのような入力の記憶が必要な組込みソフトウェアは、LBA の記述能力で十分表現可能と考え、我々は LBA を適用することにした。

3 事例研究

我々は入力に記憶が必要な組込みソフトウェアを、以下の二種類の特性に分類した。

- 入力を記憶しておく必要があり、その記憶された複数の入力と外部の状態を比較して出力を決定する必要がある組込みソフトウェア。(以下：外部の状態に依存した特性)
- あらかじめ入力に優先順位がついていて、入力の順序とは関係なく、優先順位の高い入力から実行される組込みソフトウェア。(以下：入力に優先

順位がある特性)

外部の状態に依存した特性の具体例としてエレベータの制御ソフトウェアを、入力に優先順位がある特性の具体例として携帯電話のモード制御ソフトウェアを題材とする。これらを LBA を用いてモデル化する。

3.1 エレベータのモデル化

エレベータの構成要素と各構成要素の役割を 表 1 に示す。扉はエレベータリフトの動作に影響を与えないので省略する。

表 1: 構成要素とその役割

構成要素	役割
エレベータ	フロアを含むエレベータ全体の制御
ボタンコントロール	入力されたボタンの値を記憶し、次の停止する階、または次に移動する方向を決定する
エレベータコントロール	エレベータリフトの動作を制御する
リフト	人を乗せる部分
中ボタン	行き先階を指定するボタン
外ボタン・上(下)	フロアでリフトを上(下)呼び出すボタン
センサ	リフトの位置を判定する

状態遷移機械の概要

エレベータ制御ソフトウェア全体を一つの状態遷移機械で表現すると複雑になるので、構成要素ごとに状態遷移機械で表現する。構成要素をクラス図で表現し、図 1 に示す。

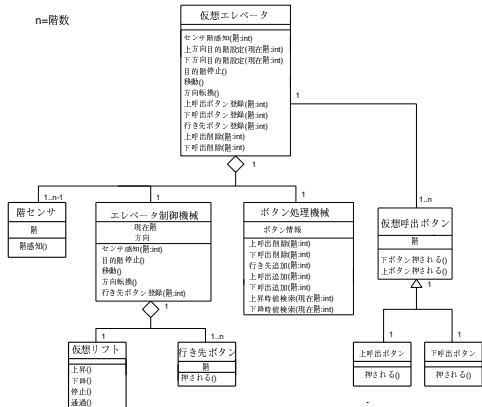


図 1: エレベータ制御ソフトウェアのクラス図

リフトは、移動の方向や現在階などの動作状況と使用者が押したボタンをもとに動作するので、入力されたボタンを記憶する必要がある。この記憶はボタン処理機械で行う。ボタン処理機械は階層化され、親の状態遷移機械を DFA、子となる状態遷移機械を LBA を用いて記述する。ボタン処理機械以外は DFA を用いて記述した。表 2 に構成要素と状態遷移機械の名前とオートマトンの対応を示す。

表 2: 構成要素と状態遷移機械の名前とオートマトンの対応

構成要素	状態遷移機械	オートマトン
エレベータ	仮想エレベータ	DFA
ボタンコントロール	ボタン処理機械	DFA+LBA
エレベータコントロール	エレベータ制御機械	DFA
リフト	仮想リフト	DFA
中ボタン	行き先ボタン	DFA
外ボタン・上(下)	上(下)呼出ボタン	DFA
センサ	階センサ	DFA

ボタン処理機械の状態遷移図

ボタン処理機械を階層化して記述する。ボタン処理機械は、一つのテーブルに対し検索や追加などの複数の書き換え動作を行わなくてはならない。構造を整理するためにボタン処理機械を階層化する。ボタン処理機械は階層の親の状態遷移機械がその子となる状態遷移機械(内部の状態遷移機械と呼ぶ)を自分自身の状態遷移図の一部として持っている。親の状態遷移機械が内部の状態遷移機械を持つ状態に遷移することで、内部の状態遷移機械が動作を開始する。親の状態遷移機械の状態遷移図を図 2 に示す。

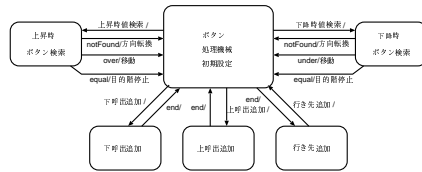


図 2: ボタン処理機械の状態遷移図

LBA の読み書きテーブルの概要

ボタン処理機械の読み書きテーブルの概要を以下に示す。

- LBA の読み書きテーブル内にはあらかじめ、左から上呼出ボタン、区切り記号、行き先ボタン、区切り記号、下呼出ボタンの順に配置しておく。
- LBA の読み書きテーブル内には、各ボタンが ON または、OFF であるかを記憶する。テーブル上の $-i$ は OFF であることを、 i は ON であることを意味する。

ボタン処理機械の読み書きテーブルの概略図を図 3 に示す。

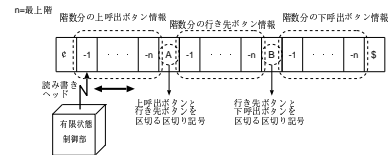


図 3: ボタン処理機械の読み書きテーブルの概略図

内部の状態遷移機械を LBA を用いて記述する。以下に内部の状態遷移機械である上昇時ボタン検索、行き先追加について説明する。外呼出追加は行き先追加とほぼ同様に考えることができるので省略する。

上昇時ボタン検索

上昇時ボタン検索は、外部からリフトの現在位置を受け取り、現在位置より一つ上の階の値がテーブルにあるかを検索する。読み書きヘッドの動作の概略を以下に示す。仮想エレベータから受け取る値を a とし、返す値を $equal$, $notFound$, $over$ とする。 $over$ は現在階より上の階の値があることを示す。

1. テーブルの中を左から見ていき、行き先ボタンか上呼出ボタンのどちらかで a と同じ値が記憶されているかを検索する。

2. a を発見したら、テープの中の a と同じ階の値を全て削除し、equal を返す。
3. a を発見できなかったら、a よりさらに上の階の値を検索し直す。
4. 発見したら over を返す。
5. 発見できなかったら、下呼出ボタンで a または a より上の階の値を検索する。
6. 発見できなかったら、notFound を返す。

行き先追加

外部から押されたボタンの値を受け取り、押されていることを記憶する。行き先追加の動作の概略を以下に示す。追加するボタンの値を a とする。

1. テープの中を左から見ていき、a 階のボタンの情報を記憶する区画までヘッドを動かす。
2. すでに a が記憶されていたときは、なにもしないで終了する。記憶されていないときは OFF を ON に書き換え、終了する。

3.2 携帯電話のモード制御部分のモデル化

携帯電話のモード制御ソフトウェア全体を構成要素ごとに状態遷移機械で表現する。携帯電話は、あるモード中に優先順位の高いモードのイベントが起こった場合、現在のモードを後から復帰させるために記憶する必要がある。モード記憶機械は 3.1 節のボタン処理機械と同様に階層化する。本節では、ボタンが押されると同時に各モードに遷移する携帯電話を考える。以下に構成要素を示し、役割と状態遷移機械とオートマトンの対応を表 3 に示す。

表 3: 構成要素と役割と状態遷移機械とオートマトンの対応

構成要素	役割	状態遷移機械	オートマトン
モードコントローラ	インターネットやゲームなどの様々なモードを制御する。	モード制御機械	DFA
モードメモリ	モードの記憶をし、記憶されたモードの中から優先順位の高いものをモード制御機械に返す。	モード記憶機械	DFA+LBA
ディスプレイ	携帯電話の画面表示。	仮想ディスプレイ	DFA
回線接続装置	インターネットやメール、電話をするために回線につなぐ。	仮想回線接続装置	DFA
受信装置	メールや、電話を受け付ける。	仮想受信装置	DFA
アラーム	設定された時間を知らせる。	アラーム	DFA
ゲームボタン	ゲームを開始する。	仮想ボタン	DFA
インターネットボタン	インターネットを開始する。		
発信ボタン	電話をかける。		
メールボタン	メール読み書きを開始する。		
電源ボタン	各モードを終了させる。長押しした場合、携帯電話の電源を入れる。		
メニューボタン	アドレス登録や携帯電話の各種設定など様々な作業を行う。		

モード記憶機械の状態遷移図

モード記憶機械を階層化し、内部の状態遷移図に LBA を適用する。モード記憶機械の状態遷移図を図 4 に示す。

LBA の読み書きテープの概要

モード記憶機械の読み書きテープの概要を以下に示す。

- LBA の読み書きテープ内にあらかじめ、左から優先順位の高いモード順に配置しておく。
- LBA の読み書きテープ内には、各モードが ON または、OFF であるかを記憶する。

モード記憶機械の読み書きテープの概略図を図 5 に示す。

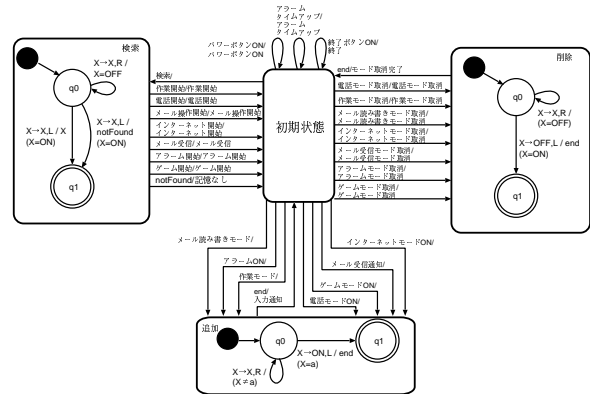


図 4: モード記憶機械の状態遷移図

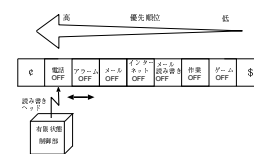


図 5: モード記憶機械の読み書きテープの概略図

図 4 に示されている、内部の状態遷移機械である検索の読み書きヘッドの動作の概略を以下に示す。追加・削除は省略する。

検索

記憶したモードの中から優先順位が最も高いモードを検索する。

1. テープの中を左から順に見ていき、ON になっているモードがないか検索する。
2. ON になっているモードを発見したらそのモードを返す。

4 考察

本節では LBA 適用の妥当性、LBA を用いた組込みソフトウェアのモデル化について考察する。

4.1 LBA 適用の妥当性

従来では、エレベータのスケジューリングの部分まで、状態遷移機械で表現されていなかった。厳密に定義されていなく、あいまいさがあった。我々はエレベータのスケジューリングに記述能力の高い LBA を用いることで、あいまいさのない記述が可能となった。

最初に我々はエレベータを DFA で記述しようと試みた。DFA は押されたボタンを状態で記憶する必要がある。エレベータの中ボタンの押される組合せの数は $\sum_{i=0}^n n C_i$ ($n =$ 階数) で表すことができる。同様に上(下)呼出ボタンの組合せ数は $\sum_{i=0}^{n-1} n-1 C_i$ と表せる。したがってボタンの組合せは $\sum_{i=0}^n n-1 C_i \times (\sum_{i=0}^{n-1} n-1 C_i)^2$ になり、状態数は膨大になる。

PDA を用いてエレベータのモデル化を試みた。記憶の操作の仕方が後入れ先出しに限定される。記憶を自由

に参照するためには、スタックから取り出した値を別のスタックに記憶すればよい。これは PDA の定義に反する。

本研究では LBA を適用することで状態数を減少させることができ、状態遷移機械で記述することが可能になった。さらに階数の変更を行ってもテープの区画数を階数分だけ用意すれば、状態遷移図は変わらない。

4.2 LBA を用いた組込みソフトウェアのモデル化
 入力記憶を必要とする組込みソフトウェアに、我々の提案する LBA の適用手順を用いることで系統的な記述が可能になると考える。以下に適用手順を示す。

1. 組込みソフトウェアを一つの状態遷移機械で記述すると複雑になるので、構成要素に分割しそれぞれを状態遷移機械で記述する。
2. 記憶あり入力に LBA を適用する。
3. 記憶された入力記号列に対して検索、追加や削除など複数の異なる動作を行わなくてはならない。構造を整理するために状態遷移機械を階層化する。親の状態遷移機械を DFA で、内部の状態遷移機械を LBA でモデル化する。
4. テープにあらかじめ記憶の必要がある入力の分だけ区画を用意し、配置を決める。
5. 親の状態遷移機械と内部の状態遷移機械の状態遷移図を記述する。
6. 親の状態遷移機械が状態遷移機械間のイベント通信を行う。

外部の状態に依存した特性・入力に優先順位がある特性を持つ組込みソフトウェアへの LBA の適用が有効である。特性と具体例と LBA のテープの初期配置の対応を表 4 に示す。

表 4: 特性と具体例と LBA のテープの初期配置の対応

特性	具体例	LBA のテープの初期配置
外部の状態に依存した特性	エレベータ制御ソフトウェア	<ul style="list-style-type: none"> 外部の状態と記憶された複数の入力を比較しやすいように入力される値の配置を決める 入力の種類を区別したいとき、区切り記号を加えることで種類の違う入力一つのテープで扱える
入力に優先順位がある特性	携帯電話のモード制御ソフトウェア 複合複写機のモード制御ソフトウェア	<ul style="list-style-type: none"> 入力に優先順位を付け、優先順位が高いものをテープの左側から配置する テープ内には入力の有無を記憶したいので入力の ON、OFF を記憶させる 入力の種類を区別したいとき、区切り記号を加えることで種類の違う入力一つのテープで扱える

エレベータ群制御機械への LBA の適用

複数基のエレベータを制御するエレベータ群制御機械を考える。3.1 節のエレベータのモデル化では、一基のエレベータを考えた。エレベータを複数基動作させるとき全てのリフトの現在位置を覚えておく必要がある。エレベータ群制御ソフトウェアを考えると、以下の 2 つの状態遷移機械に LBA を適用する必要があると考える。

- ボタンの入力を記憶しておくボタン処理機械

- 各リフトの位置を記憶するエレベータ群制御機械
 エレベータ群制御機械は以下のような特性があるので、外部の状態に依存した特性の組込みソフトウェアに分類される。

- 複数基のエレベータの個々の位置、リフトの方向を記憶し、その記憶を任意に参照する必要がある。
- 呼出しボタンが押された時に複数基のエレベータの中から適切なエレベータを動作させる必要がある。

テープにあらかじめ記憶の必要がある入力の分だけ区画を用意し、エレベータの ID・階・方向の配置を決める。ID が A のエレベータが上向き 2 階にあり、ID が B のエレベータが下向き 3 から 2 階間にあるときのテープの概略図を図 6 に示す。

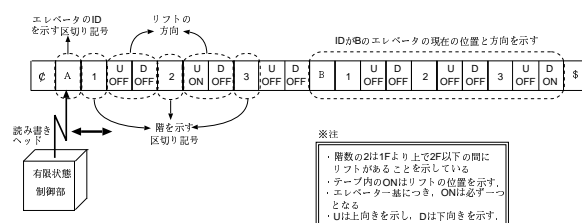


図 6: エレベータ群制御機械の読み書きテープの概略図

テープ内の配置を図 6 のように考えることで、複数基のリフトの個々の現在階・方向を記憶することが可能となった。LBA の適用手順にしたがってモデル化することにより系統的に記述することが可能になった。

5 おわりに

本研究では LBA を用いてエレベータと携帯電話の組込みソフトウェアを系統的に記述することが可能であることを確認できた。

今後の課題は、本研究の方法を他の組み込みソフトウェアに適用し、その有用性を検証することである。

謝辞

本研究を進めるにあたり、二年間御指導いただいた野呂昌満教授、有益なアドバイスをいただいた張漢明助教授、貴重な時間をさいて数多くのミーティングをして下さった蜂巣吉成講師、大学院生の熊崎敦司さん、藤原泰昌さん、森貴彦さん、後藤修平さんに深く感謝致します。

参考文献

[1] A. サローマ, 野崎昭弘, 町田元, 山田秀記, 横森貴: 計算論とオートマトン理論, サイエンス社 (1988).
 [2] 富田悦司, 横森貴: オートマトン・言語理論, 森北出版株式会社 (1992).