代数仕様言語を用いたマルチカーエレベータの仕様記述と検証

安藤　大貴　　†中村　正樹†　　榊原　一紀†
† 富山県立大学工学部情報システム工学科

はじめに

形式手法は、数学を基盤としたソフトウェアやハードウェアシステムの仕様記述、開発、検証の技術である。代数仕様言語 CafeOBJ は代数をモデルに持つ、形式仕様の記述を可能にする言語である[1]。マルチカーエレベータ(MCE)は一本のシャフト内に複数のかごが独立して運行するエレベータのことである。複雑な運行制御を行う為、通常のエレベータよりも安全に効率的に制御を行う必要がある[2]。そこで本研究では、CafeOBJ によるマルチカーエレベータの仕様記述と検証を書き換え仕様と振る舞い仕様の2種類の方法で行い、比較する。

マルチカーエレベータのシャフト内同一方向制御

本研究で扱う MCE のモデルはシャフト内にかごを 2 つ設置したものである。かごは現在階と進行方向を要素に持つモデルを扱う。運転方式はシャフト内同一方向制御方式を採用した。また、デッドロックを回避する為に退避階にいるかごの方向転換は追い越す側のかごが方向転換した後に行う(図1)。デッドロックとは進行方向が互いに向き合い先に進めなくなってしまう状態である。

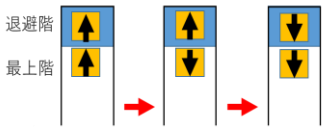

図1:方向転換

書き換え仕様による MCE 仕様の記述と検証

書き換え仕様は、システムの状態を項として明示的に記述し、項の間の書き換え関係によってシステムの動作を記述する仕様である。この仕様では項の取り得る遷移を網羅的に探索することができる。例えば、項 car1(4,up)car2(2,dn)は、上かご(car1)が 4 階で上(up)を向き、下かご(car2)は2階で下(dn)を向いている状態を表す。以下に仕様の一部である、書き換え関係の式を示す。

```
ctrans car1(X,up)car2(Y,up)
    => car1(X+1,up)car2(Y,up) if (X =/= Max1)
```

書き換え関係を表すキーワード ctrans を使って、上かごは最上階(Max1)以外の階(X)で上(up)向きかつ下かごが上(up)向きならば1つ上の階に遷移することを表す。CafeOBJ の処理系では到達可能性を検査する述語==>を使って、左辺から右辺への遷移が存在するか調べることができる。以下は、初期状態から進行方向によらず同時に x 階に存在する遷移があるかどうか、また、デッドロックが起こる遷移があるかどうか検査することを表す。

```
red car1(1,up)car2(0,up)==>car1(X,D1)car2(X,D2)
red car1(1,up)car2(0,up)==>car1(X,dn)car2(Y,up)
```

振る舞い仕様による MCE 仕様の記述と検証

振る舞い仕様は対象とするシステムをブラックボックスとみなし状態を観測する演算と状態を変更する演算を使ってシステムの振る舞いを記述する仕様である。かごの位置と進行方向を観測する演算子 loc、dir と、かごの遷移を表す演算子 next を使って MCE の振る舞いを記述する。以下に next を定義する一部の等式を示す。

```
ceq loc(c1, next(c1,A)) = loc(c1,A) + 1
   if (dir(c1,A) =u p) and (dir(c2,A) = up)
   and (loc(c1, A) < max1) .
```

ceq は条件付き等式を宣言している。事前状態 A に対し loc(c1,A)と dir(c1,A)はそれぞれ、上かごの位置と進行方向を表し、next(c1,A)は上かごが遷移した後の状態を表す。上記の等式は if 以下の条件が成り立つ時、loc は 1 つ上の階に遷移することを表す。

振る舞い仕様の検証は、A ならば B という命題に対して仮定 A を等式(eq)で宣言し、結論 B をそれらの等式のもとに等式推論(red)で確かめる。それらの等式推論の組み合わせで帰納法や場合分けを行い、形式的に証明する。

```
eq (loc(c1,a) > loc(c2,a)) = true
red loc(c1,next(c1,a)) > loc(c2,next(c1,a))
```

上記は、等式(eq)で上かごの位置は下かごの位置より大きいと仮定して、等式推論(red)で上かごが next した状態で位置関係が変わらないことの証明である。

おわりに

書き換え仕様と振る舞い仕様で記述されたマルチカーエレベータの仕様を作成し、衝突及びデッドロックがないことを検証した。書き換え仕様では網羅的に自動で検証することができた。しかし、階数を増やすと結果が返ってくる時間が長くなった。振る舞い仕様ではシステムの規模の大きさによらず、対話的に検証することができた。しかし、人が場合分けをある程度行って、証明する必要があった。今後の課題として、複雑なかごの制御を行う仕様の記述と検証や呼びの概念の実装が挙げられる。

参考文献

[1] 二木ら, CafeOBJ 入門(1)~(3), コンピュータソフトウェア, (1)Vol.25, No.2, pp.1-13, (2)pp.14-27, (3)No.3, pp.69-80, 2008.
[2] 山口ら, CST ソリューションコンペティション 2007 及び 2008 の総括, 電子情報通信学会, 信学技報, CST2009-11, pp.59-64, 2009.