

## 組込みシステム向け関数リアクティブプログラムに対するテストケース生成手法

内野駿亮, 森口草介, 渡部卓雄 (東京工業大学)

## 概要

本研究では, 組込みシステム向け関数リアクティブプログラミング(FRP)言語に対するテストケース生成手法を提案する. 提案手法では, LTLの拡張であるメトリック時相論理(MTL)を用いる. 具体的には, MTLに不等式等による制約を加えることで, 数値を含むより広範なテストケースの生成を行えるようにする. これにより, ある値が変化してから所定の時間内に特定の振る舞いが生じなければならないといった制約に対するテストを行うことが可能になる.

## 関数リアクティブプログラミング(FRP)

時間とともに変化する値を時変値として抽象化し, 時変値間の依存関係を記述することによりシステムを記述する. Emfrp[1]は小規模組込みシステム向けに開発されたFRP言語であり, 実装したプログラムはDAGを用いて表すことができる. コンパイルされたEmfrpのプログラムはinput-process-outputを繰り返すことで実行される.

[1] Kensuke Sawada and Takuo Watanabe. Emfrp: A functional reactive programming language for small-scale embedded systems., MODULARITY Companion 2016, pp. 36–44, 2016.

## メトリック時相論理(MTL)

線形時相論理(LTL)に対して時間的制約を追加した論理体系. ここではLTLにおけるF,G,Uに対してのMTLにおける拡張のみを使用する

$F_I \varphi$ : 現時点から時区間Iにおいて少なくとも一回 $\varphi$ が成り立つ

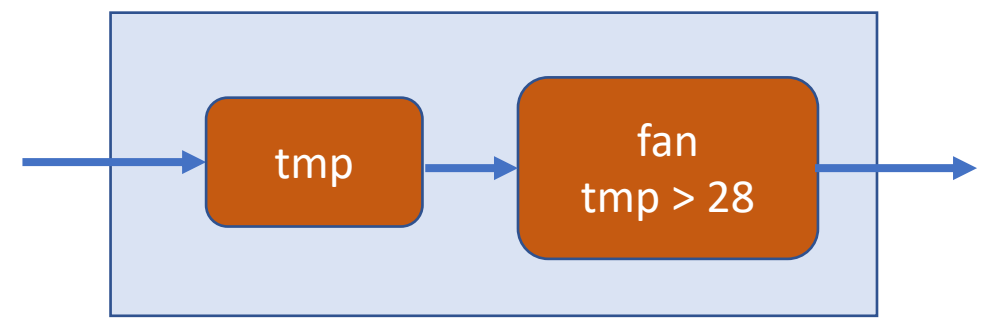
$G_I \varphi$ : 現時点から時区間Iにおいて常に $\varphi$ が成り立つ

$\sigma U_I \delta$ :  $\sigma$ は $\delta$ が成り立つまで常に真

$\delta$ が真になった時とその直前の状態との時間隔が, 時区間Iに含まれる

## FRPで記述されたプログラムに対するテスト手法

- FRPで記述されたプログラムに対するテストのためには以下が必要である
  - テスト対象のプログラム
  - プログラムに対する入力としての配列
    - 長大な入力列が必要であるため自動生成が必要
  - プログラムの実行結果として得られた出力の配列を評価する機構



tmpが28度を超える時にfanが稼働するFanController

右の図のようなFanControllerを例とすると,

- 入力として, 10から20秒の間はtmpが28度となり, それ以外では30度となる入力の配列
  - $(sec, tmp): [(0,30), (8, 30), (11, 28), (17,28), (23,30), (29,30)]$
  - MTLを用いると,  $G_{[0,10]}(tmp == 30) \wedge G_{[10,20]}(tmp == 28) \wedge G_{(20,\infty)}(tmp == 30)$ のように表すことができる
- 出力を評価する機構として与えた配列をMTL式で評価する機構を用いる
  - プログラムが正しく実装されている場合, 上記の入力に対する出力はfan:  $[true, true, false, false, true, true]$ となる
  - MTL式では,  $G_{[0,10]}fan \wedge G_{[10,20]} \neg fan \wedge G_{(20,\infty)}fan$ と書くことができる

## 既存研究: LTLによるテストケース自動生成手法 [2]

この先行研究では, 以下の手順によりLTLを用いた状態列の生成を行う.

- 与えられたLTL式 $\varphi$ および初期条件を規定する $\varphi_0$ を基に状態列の論理変数並びにその添え字間の関係式を生成する
- SATソルバを用いて生成された関係式の充足可能性を判別し, 充足可能な場合はそのモデルを生成された状態列とする

[2] Ingo Pill and Franz Wotawa. Automated generation of (F)LTL oracles for testing and debugging. Journal of Systems and Software, Vol. 139, pp. 124–141, 2018.

## 提案手法: MTLを用いたFRPのためのテストケース生成手法

MTLを用いて既存研究の手法を以下のようにしてFRPに対応させる.

- MTLの各演算子が使用する時間列の生成
  - 時間列の生成は, 自身を除く状態列の生成とは独立に実行
  - 本研究では複数(ランダム, 周期的)の生成手法を用意
- LTLにおけるF,G,Uに関する関係式をMTLにおける $F_I, G_I, U_I$ に対応するように変更
  - $U_I$ については, LTLにおけるUと同様に漸化式を使用した関係式を定義
    - 時区間Iが0を含む場合とそうでない場合で異なる関係式を使用
  - $F_I, G_I$ では生成した時間列をもとに, 一致するような場合を全列挙し関係式を生成

提案手法:  
数値への対応

SATソルバではなくSMTソルバを用いることにより, 数値を含む状態列の生成を可能にした.

SMTソルバは本研究においては, Z3 Proverを用いた.

## 評価

以下のMTL式に対して長さ2000の状態列の生成時間の計測を行った

生成のためのMTL式	生成時間
$Pushed \wedge (F_{[90,110]} Pushed)$	486.7ms
$G_{[0,10]}(tmp == 25)$	159.4ms
$Detected U_{[2,10]}(G Alert)$	600.7ms

## 今後の課題

- より複雑な言語機能への対応
  - 直和型・直積型
- 生成した時間列が充足不能となる場合の対処
  - 時間列をSMTソルバを経由して生成することにより対応可能
- ランダムな状態列の生成
- 既存研究で対応していたLTLの演算子に対するMTLでの生成の対応