2J-07

組込みシステムにおける要求分析モデルからのテストケース導出手法

西川　俊†　　松浦　佐江子‡
芝浦工業大学　理工学研究科　システム理工学専攻†‡

１．はじめに

組込みシステムではハードウェア構成ならびにシステムの実行環境という条件によりソフトウェアの制御が変化する．センサやアクチュエータの性能がシステム全体のゴールに大きく影響するため，想定したゴールを満たすかという観点からハードウェアの性能評価実験を事前に行う必要がある．本稿では，システムのゴールをUML(Unified Modeling Language)のアクティビティ図を用いて分析し，ハードウェア構成と動作環境の条件下で，そのゴールを満たすことを効率よく確認するテスト方法を提案する．本研究の対象はAPI(Application Programing Interface)が既知である既存のハードウェアを使った開発であり，またUML を用いたモデルベース開発とする．

２．関連研究

UML のようなモデルからソフトウェアのテストケースを導出する研究が行われており，[1]ではWeb システムを対象として設計モデルからテストケースを自動生成している．一般的な PC やサーバで動作するシステムの場合，ハードウェアの性能は明らかであるが，組込みシステムの場合は，システムによってハードウェア構成や求められる性能が異なり，想定したハードウェア構成ではシステムのゴールが達成できなくなる可能性が十分ある．そのため要求分析段階という開発の早期段階で，ハードウェアに対しシステムのゴールを満たすことを確認するテストを行う必要がある．

３．適用事例

事例は本学の授業課題である「荷物自動搬送システム」とする．このシステムのゴールは「2 台のロボットが規定されたコース上の道を自律走行し，データによって仮想的に生成された荷物を通信によって受け渡すことで荷物を自動搬送すること」である．ロボットは LEGO MINDSTORMS EV3 で作成し，図 1 のコースの受付所から中継所まで荷物を運ぶ「収集担当ロボット」と待機所から中継所へ移動し，荷物を受け取って受取人宅へ届ける「配達担当ロボット」がある．

４．提案手法

４.１．ワークフローの作成

ワークフローとはシステムのゴールを満たす振る舞いを動作系列としてアクティビティ図によって表したものである．事例のワークフローである図 2 では，依頼人から受付所，収集担当ロボット，中継所，配達担当ロボット，受取人宅という順の荷物の流れが表されている．この後の手順は，組込みシステムである収集担当ロボットと配達担当ロボットについて分析する．また，ワークフローを定義する上で前提となる環境の要素をクラス図で表す．事例では，ロボットは「異なる 2 つの地点の間をコース上の道に沿って移動する」というサブゴールを持つことから，移動に使う道があることを想定しているため「道」クラスを作成する．

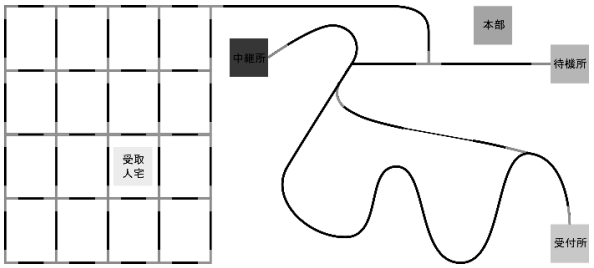

図 1.「荷物自動搬送システム」のコース

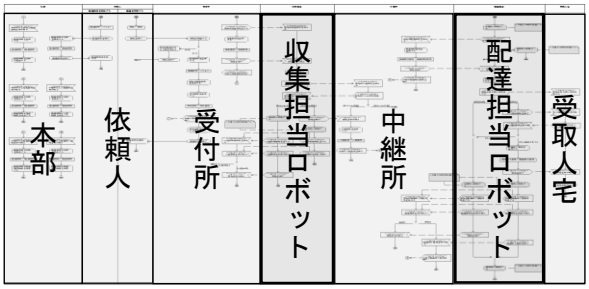

図 2.「荷物自動搬送システム」のワークフロー

４.２．ハードウェア依存アクションの抽出

ハードウェア依存アクションとはワークフローにあるセンサやアクチュエータを用いて実現するアクションであり，この手順ではこれを特定する．事例の収集担当ロボットでは「中継所へ移動する」と「受付所へ移動する」というコース上を走行するアクションが特定できた．

４.３．ハードウェア課題の定義

ハードウェア課題とはハードウェア依存アクションの実現のためにセンサやアクチュエータを用いて解決しなければならない課題である．これをユースケースとしてユースケース図によって定義する．そのユースケースの事後条件として課題を解決して期待する結果も定義する．事例の「中継所へ移動する」のハードウェア依存アクシ

A Test Case Derivation Method from Requirements Analysis Model in Embedded System
†Shun Nishikawa ‡Saeko Matsuura
†‡Systems Engineering and Science, Graduate School of Engineering and Science, Shibaura Institute of Technology

ョンはコース上の受付所から中継所までを移動するアクションである．その経路を走破するには「道に沿って進む」ことと，目的地で停止するために「止まる場所であるか判断し止まる」こと，中継所前の2台のロボットが共有する道で衝突回避をするために「共有の道に他のロボットがいるか判断する」ことが必要であり，この3つをユースケースとして定義した．このユースケース図を図3に示す．

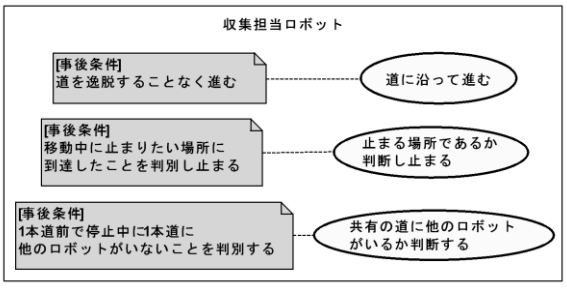

図3.「中継所へ移動する」のユースケース図

4.4．ハードウェア課題解決案の作成

ハードウェア課題解決案とはハードウェア課題を解決するアルゴリズムをアクティビティ図で表したものである．作成にはアルゴリズムの知識と，使用するハードウェアの選定が必要である．選定したハードウェアは，そのAPIにあるフィールドや関数からクラス図で表す．アクティビティ図のアクションノードは選定したハードウェアのクラス図の操作と同じ粒度で記述する．また，アルゴリズムの事前条件も記述する．事例の「道に沿って進む」のハードウェア課題では，PID制御によるライントレースの知識の準備と，使用するハードウェアは光センサ1つとモータ2つという選定を事前に行った．これにより，光センサで取得した道からのズレの値から，PID制御によってモータの出力を変化させる値を算出し，それを左右のモータに反映させるアルゴリズムのアクティビティ図を作成した．このアクティビティ図の一部を図4に示す．

4.5．テストケースの導出

本研究のテストケースはテストプログラムとテスト指示書を組にしたものである．テストプログラムは，ハードウェア課題解決案のアクティビティ図の各アクションを関数として，その動作系列を再現したものである．選定したハードウェアの関数を使うアクションは，その関数を用いてプログラムを記述し，ハードウェアを使わないアクションはそれに対応する関数を作成し，それを用いる．事例では「道に沿って進む」のハードウェア課題解決案のテストプログラムを作成した．その解決案の一部である図4と対応するテストプログラムを図5に示す．

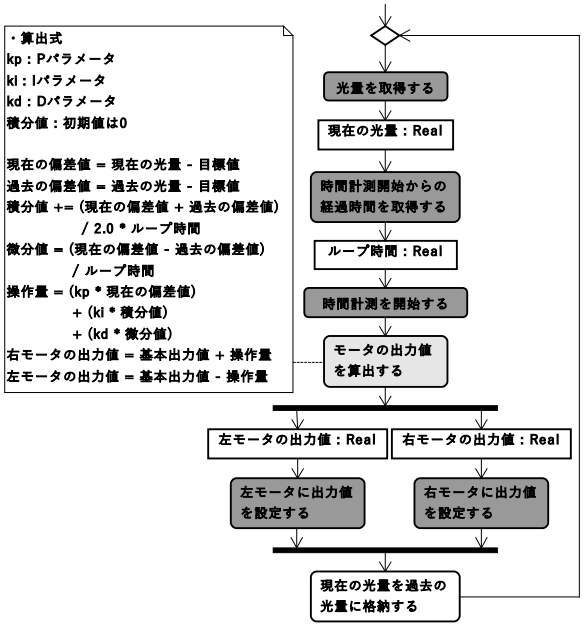

図4.「道に沿って進む」のアクティビティ図

```
while(true){
    光センサ.光量を取得する(現在の光量, offset);
    ループ時間=ストップウォッチ.経過時間を取得する();
    ストップウォッチ.リスタート();
    出力値を算出する();
    左モータ.出力値を設定する(左モータの出力値);
    右モータ.出力値を設定する(右モータの出力値);
    過去の光量=現在の光量
}
```

図5．図4のテストプログラム

5．まとめ

組込みシステムでは環境に合わせたパラメータを用いてハードウェアを制御することが多いため，そのテストではパラメータの調整作業が必要になる．提案手法では実環境下で動作するテストプログラムが作成できるため，その動作を観察してパラメータ調整が可能である．また，他のテストプログラムの動作やログを観察することで，解決できるか確認できるハードウェア課題もあり，その場合は動作やログの観察でテストを済ませることで効率を上げることができる．そうしてテストを行い，ハードウェア課題の事後条件にある期待結果が得られた場合，そのハードウェア課題は解決可能である確認が取れる．全てのハードウェア課題が解決できることが確認できた場合は，想定したハードウェア構成と実行環境の条件下でシステムのゴールを満たすことが確認できる．今後の課題として，パラメータ調整のサポート方法を含んだテスト指示書の作成と，ハードウェアのクラス図を利用してアクティビティ図からのテストプログラムの自動生成を検討する．

6．参考文献

[1]丹野治門，張暁晶，星野隆．結合テストにおけるテスト項目自動生成手法の提案と評価．信学技報，第110巻 of SS2010-34, pp.37-42, 2010