

第7分科会 Aグループ

機能の組み合わせテスト(シソーラスに基づくテスト)

Features Combination Testing(Thesaurus based Testing)

分科会メンバー

- 主査 保田 勝通 (つくば国際大学)
副主査 西 康晴 (電気通信大学)
高橋 寿一 (ソニー株式会社)
研究員 宇野 也寸志 (株式会社ダイキンシステムソリューションズ研究所)
大森 慈浩 (東芝メディカルシステムズ株式会社)
鈴木 三紀夫 (T I S 株式会社)
竹本 昇司 (株式会社野村総合研究所)
(50音順)

概要

組込システムにおいては機器の性能向上に伴い、複数の機能が同時並行的に稼動するようになった。システム総合テストでは、実運用において機能間に不整合が発生しないことを確認する必要がある。現実には、増え続ける機能に対して、システム総合テストですべての組み合わせを実施、検証することは時間的に不可能である。

本研究では、既存のシステムに対して機能追加を行った際、既存機能との組み合わせテストケースを絞るためにシソーラス(分類語彙表)を用いる手法を考案し、適用方法について検討した。

Abstract :

As for the embedded system, with the performance improvement of the equipment, more than one feature got to operate concurrently. At the integration testing, it is necessary to confirm that the discordance doesn't occur during feature in operation.

Actually, it is impossible to verify all combinations of the feature which continues to increase with the integration testing.

We develop one of the software testing techniques, which is the thesaurus based testing. The thesaurus based testing is to use thesaurus that can choose combination test cases: between existence feature and additional feature on existing system. We have investigated way of the techniques and its application in this paper.

1. まえがき

1.1 研究の背景

近年では、CPU性能の向上により、組込系ソフトウェアでも機能数・規模ともに急激に大きくなってきている。それに伴いすべての機能の組み合わせテストを実施するのが困難

になっている。一方で、組込系ソフトウェアは高い信頼性を求められており、テストの重要性が増してきた。

1.2 研究の目的

最低限テストを実施しておきたい組み合わせを選択するために、シソーラスに基づくテストを提案する。

1.3 本論文の流れ

2章で現状の問題点に関して一般論と事例を挙げる。3章で事例に対する問題点とその対応策を提案する。最後に、今後の課題を示す。

2. 現状の問題点

2.1 機能の組み合わせテストに関する一般的な問題点

ソフトウェアテストでは、機能毎の確認は当然実施するが、それ以外に複数の機能を同時に動作させるといった機能の組み合わせテストが必要となる。

例として、機能数が1000個あるシステムを考える。機能の組み合わせテストでは、

- ・機能が同時に動いても問題なく動作することを確認すること
- ・同時に動いてはいけない機能が仕様どおり動作しないことを確認すること

の2つの確認が必要になる。並行して動作する機能数が100個の場合、2機能の組み合わせでも1万通りの試験が必要となり、テスト時間を確保するのが現実的に困難になってしまう。実際には、同時に動作する機能の数が2つとはかぎらないので、同時に動かせる機能の数に応じて、さらに組み合わせは増え、すべての組み合わせテストをすることは現実的ではない。

このような現状を踏まえ、テストケースを絞り込む方法を検討した。このような現状を踏まえ、テストケースを絞り込む方法として、ユースケースから絞り込むことを検討した。ユースケースの抽出方法として次の2つの方法を検討した。

(1) 顧客のユースケースから絞り込む方法

顧客のユースケースからテストケースを絞り込むことを検討した。しかし、組込系ソフトウェアは、顧客の環境(周辺機器などのオプション構成、ネットワーク環境など)やその使い方によりユースケースが多岐にわたる。顧客の操作ログを記録する仕組みを備えるシステムもあるが、収集した操作ログが多種にわたるため分類・分析が難しく、ここから顧客のユースケースを網羅した操作を導き出すことは難しい。

(2) 仕様を理解している人がテストケースを絞り込む

システムの仕様を完全に理解している人であれば、テストケースを上げる際に漏れることはなくなる。しかし、先に述べたようにシステムが複雑なものや、従来のソフトウェアに機能を追加して開発するなど、仕様を完全に理解している人は少ない。

以上のようにテストケースを絞り込む方法がないことが問題である。本研究では、既存の機能に対して機能追加・変更を行うようなバージョンアップ開発を前提とした。既存機能すべての仕様を完全に理解していなくても効率的且つ、効果的にテストケースを絞り込

む方法について検討した。

まず始めに、機能追加を行った場合に、既存機能との組み合わせの検討が漏れてしまったことにより障害が発生した事例を紹介する。次に原因を探り、最後に対応策を検討する。

2.2 事例

図1の様な構成において、コントローラから機器A～Cに対して、自動、或いは、手動にて運転又は、停止指令を出す機能がある。その時、10～20までの機能が並行して動作を行うが、仕様上、特定の場合において、ある機能が別の機能に影響を与える事がある。そのケースを図2を用いて説明する。

同図には、2つの機能（自動発停機能と順次運転機能）の制御パターンを記述している。「自動発停機能」

は、ある制御条件が成立した場合、コントローラは対象とする機器A～Cの何れかに対して（例の場合は、機器B）運転、或いは、停止指令を出す。一方、「順次運転機能」は、ユーザがコントローラの操作ボタンにより、機器A～Cを順番に動かす機能である。機器A～Cを1つのゾーン（複数の機器が同じ制御（動作）を行うグループのこと）とする。このゾーンに対してユーザのボタン操作により運転指令を出した場合、ゾーン内のすべての機器を一斉に運転せず、任意の間隔（例の場合は、2秒）で順番に運転指令を出す機能である。

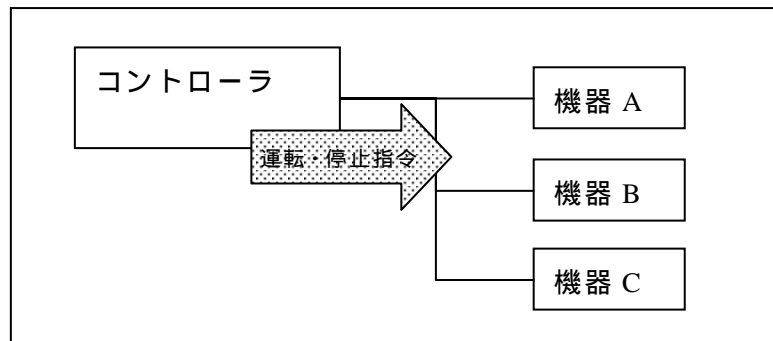


図1 装置の構成

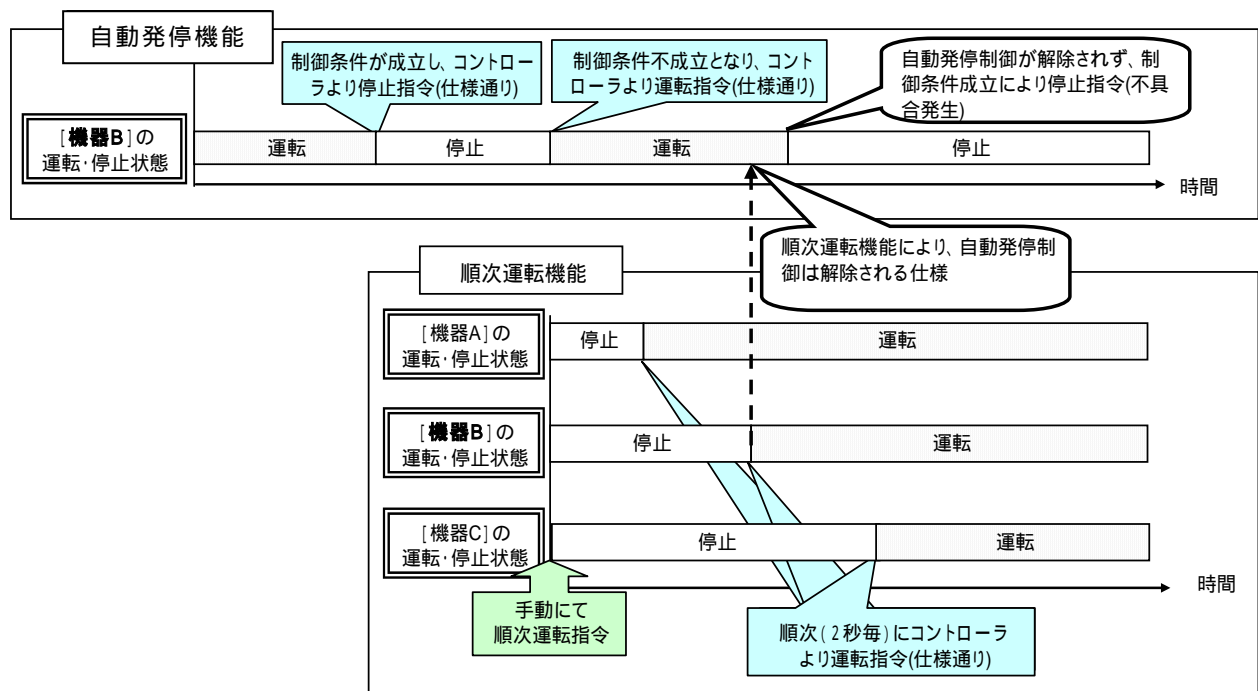


図2 2つの機能の関係

これらの2つの機能が並行して動作した場合、一方のコントローラが運転指令をもう一方のコントローラが停止指令を出すことになり矛盾した指令を機器に出してしまう。このような状況を避けるためには、どちらかの機能をキャンセルさせなくてはならない。この事例ではユーザの操作を優先させる仕様になっているため、順次起動指令が出された場合、自動発停機能の停止指令をキャンセルし、運転指令を優先させるというのが仕様となっている。(図2の点線矢印の部分)

しかし、実装されたシステムは、このキャンセルを行う処理が漏れていた。順次運転指令が出た後も、自動発停機能が動作し続け、再度、機器が停止してしまう不具合が生じていた。

3. 対応策

3.1 事例に関する問題点

テスト仕様書作成時は、それぞれ独立した機能を確認するために仕様書を十分検討して作成した。しかし今回は、自動発停機能の追加機能を確認するテストであったため、複数の機能間の関係までは機能仕様書に明記されていなかった。そのため、複数機能間の検証について、十分に検討できていなかった。機能仕様書に書かれていなかったために、テスト仕様書に反映させることはできず、結果としてテスト項目に自動発停機能と順次運転機能の組み合わせテストを盛り込むことができなかった。

一般的に見ても追加機能についての機能仕様書には関連機能について書かれていないことの方が多い。そのため、機能仕様書を読み込んでも関連機能のテストを想定するのは困難である。事例に書かれた自動発停機能と順次運転機能の組み合わせはアドホックではなかなか思いつかない組み合わせである。

3.2 従来の方法

3.2.1 従来方法の手順

従来は、追加機能を列挙しその機能に関連する要素を組みあわせてテスト項目を抽出していた。その手順を確認する。

手順1：追加機能を列挙する。

手順2：追加機能に関する要素を列挙し、機能と組み合わせテスト項目とする。

3.2.2 適用例

手順1：追加機能を列挙する。

機能1	機能2	
自動発停機能	運転指令	
	停止指令	

手順2：追加機能に関連する要素を列挙し、機能と組み合わせテスト項目とする

機能 1	機能 2	要素
自動発停機能	運転指令	1 台
		2 台
		複数台
	停止指令	1 台
		2 台
		複数台

3.3 従来方法の問題点

追加機能を中心にテスト項目を抽出しているために、既存機能と追加機能との組み合わせテストを想定し難い。既存機能と追加機能のマトリクスを作成し、すべてのテストを実施することは現実的ではない。追加機能と関連のありそうな既存機能を選択し、その機能とのマトリクスを考えることは可能である。マトリクスの作成はできるが、関連のある機能を1000以上ある既存機能から抽出することは難しい。

3.4 従来方法の改善（シソーラスに基づくテスト）

追加機能と関連していると考えられる機能との組み合わせは最低限実施したい。既存機能との関連を把握するには、テスト対象となるソフトウェアの仕様をすべて押さえていなくては不可能であるが、既存の機能に対して、定期的に機能追加、変更を行うようなバージョンアップ開発を行う組織において、仕様をすべて知っているという前提は現実的ではない。従来方法を改善するにあたり、仕様をある程度知っていればテストケースを挙げられることが望まれる。

そこで、機能名に着目し機能名の類似語・反対語から機能間の関連を導き出す方法を提案する。

3.4.1 提案する方法の手順

関連機能を導き出し、追加機能とのマトリクスを作り出すまでの手順を以下に示す。

手順1：追加機能に対する説明文を記述する。

手順2：機能名、機能説明文からキーワードを抽出する。

手順3：キーワードから類似語・反対語を着想し、リストにまとめる。

市販の類語辞典等を使用する。なお、複合語はそれぞれ個別の単語にばらす。

手順4：既存機能一覧を用意し、類似語・反対語表と既存機能一覧を見比べ、気になる機能をマークする。

手順5：マークしてある関連機能を抽出し、追加機能とのマトリクスを作成する

手順6：組み合わせを検討しテスト実施の優先順位を付け、順位の高い組み合わせを残す。

手順7：機能に関連する要素を列挙し、機能と組みあわせる。

3.4.2 適用例

提案する方法（シソーラスに基づくテスト）を2章で紹介した事例に適用する。

手順1：追加機能に対する説明文を記述する。

追加機能名：自動発停機能

機能説明文：あらかじめ登録した複数の機器に対して、入力値（電圧）（吸込み温度の変わり）が設定した値よりも大きくなった場合、機器を運転し、設定した値よりも小さくなった場合、機器を停止する。

手順2：機能名、機能説明文からキーワードを抽出する。

追加機能名：自動・発停機能

機能説明文：あらかじめ登録した複数の機器に対して、入力値（電圧）（吸込み温度の変わり）が設定した値よりも大きくなった場合、機器を運転し、設定した値よりも小さくなった場合、機器を停止する。

キーワード：自動、発進（発停の発）、停止、運転

「制御」のようにどの機能説明でも使いそうな名称はあらかじめ排除しておく。

手順3：キーワードから類似語・反対語を着想し、リストにまとめる。

市販の類語辞典等を使用する。なお、複合語はそれぞれ個別の単語にばらす。

類語辞典に書かれている単語は非常に多いため、任意の単語を選択する必要がある。類語辞典に書かれている用語だけでなく、テスト対象となるシステム特有の用語も考慮して記入する。

名詞	類似語	反対語
自動	稼働 機動 活動 始動 変動	手動 他動
運転	操作 作動 稼働 操縦 運用	停止
発進	出動 出発 出立 発向	停止
停止	抑制	進行

	廃止 移行 稼働	継続 発進
--	----------------	----------

手順 4 : 既存機能一覧を用意し、類似語・反対語表と既存機能一覧を見比べ、気になる機能をマークする。

シソーラスリストに記入した単語を自動的に割り付けるのではなく、関連の度合いを判断する必要がある。リストに含まれる単語とヒットしても、明らかに関連性が無いと判断できるものはマークからはずす。

機能 1	機能 2
印刷機能	縮小・拡大機能
	改ページ機能
	グループ化機能
鍵連動機能	運転
	停止
	運転禁止
	停止禁止
スケジュール制御	運転
	停止
	運転禁止
	停止禁止
	グループ化機能
非常 停止 機能	グループ化機能
	自動 停止 解除
	手動 停止 制御
デマンド制御	時限同期信号
	外部同期信号
順次 運転	順次 作動 指令
	起動 間隔設定
計測機能	運転 時間計測
	運転 回転計測
テナント管理	使用時間管理
	使用量計算
	テナントグループ化機能
	使用料金計算機能
異常監視	監視グループ化機能
	異常レベル設定機能
	異常マスク機能

遠隔操作・制御	運転状況監視機能
	異常通報機能
	試運転機能
	日報発報機能
	修理モード機能
.....	

手順5：マークしてある関連機能を抽出し、追加機能とのマトリクスを作成する。

手順6：組み合わせを検討しテスト実施の優先順位を付け、順位の高い組み合わせを残す。

機能	サブ機能	自動発停機能
非常停止機能	自動停止解除	
	手動停止制御	
順次運転機能	順次作動指令	
	起動間隔設定	
スケジュール制御	運転	
	停止	
	運転禁止	
	停止禁止	

計測機能	運転時間計測	
	運転回転計測	
鍵連動機能	運転	
	停止	
	運転禁止	
	停止禁止	
遠隔操作・制御	運転状況監視機能	
	試運転機能	

手順7：機能に関連する要素を列挙し、機能と組みあわせる。

従来の方法と一緒にするため省略。

3.5 提案する方法の検証

3.5.1 他事例への適用

提案した方法を同じ程度の機能をもつ別の組込系ソフトウェアに試みたところ、次のようなことがわかった。機能仕様書に基づいて機能説明文を書き、キーワードを抽出するが、キーワードの抽出に個人差があり、また、同じ人が実施しても若干の揺らぎがあった。印

刷機能を例にとると、実際にキーワードとして抽出された単語は次のようなものであった。

印刷、印字、プリント、プリント、Print、P R I N T

このような結果を踏まえ、いくつかの決まり事、例えば 全角表記とする 漢字表記を基準とする、などを決めることによってばらつきが少なくなった。

3.5.2 メリット

他事例への試行を振り返って考えられるメリットとして属人性の排除を挙げることができる。最初はばらつきが生じたが、慣れるに従えばばらつきが少なくなった。また、既存機能一覧と類語辞書（シソーラス）があれば、容易に関連していると思われる機能を抽出することが可能になる。そのため、既存機能の仕様を知っているかどうかにかかわらず、関連機能を抽出することができるようになる。

3.5.3 デメリット

しかし、次のようなデメリットもあった。

1. シソーラスの精度に依存してしまい、テスト対象との相性が悪ければ、関連する機能を導き出せない。
2. 関連していない機能でも単純に単語がマッチングしていれば抽出してしまう。
3. 既存機能の名称に使われている単語が標準化されてない場合、類義語、反対語としてのキーワードとマッチングしない危険がある。

などである。

4. 反省と今後の課題

デメリットを踏まえ、提案した方法を拡張しようと試みた。現在の方法では、追加機能から導き出した類義語・反対語を、直接既存機能一覧とマッチングしている。この方法では、既存機能名の精度に左右されてしまう。しかし、既存機能の名称を変更し精度を高める方法は採用できない。それらを改善さらに拡張すべく、次のような案を検討した。

1. 既存機能からのキーワード抽出
2. 機能毎のシソーラス作成
3. 既存機能の類義語・反対語のリスト化

しかしながら、本研究テーマを決めるまでにかかなりの時間がかかったことから、1000機能のシソーラスリストの作成まで着手することはできなかった。そのため案まで検討していないながら、それに続く事例への適用ができなかったことが反省点であり、同時に課題として残った。本成果に関心を持った人たちによって、当成果を足がかりとする有効な方法論の完成を目指して頂ければ幸甚である。

最後に、一年間の研究に適切なお指導、アドバイスいただきました保田主査、西副主査、高橋副主査、ならびにご支援をいただきました日科技連SPC事務局の皆様に厚く御礼申し上げます。

5 . 参考文献

- 「ソフトウェア・エンジニアリング」 : 菅野文友 (日科技連出版社) 1979
「ソフトウェア品質保証の考え方と実際」 : 保田勝通 (日科技連出版社) 1995
「シソーラス (類語) 検索」 <http://www.gengokk.co.jp/thesaurus/>