

詳説 FL 表

2010/7/11 (日)

秋山浩一

1. FL 表の位置づけ

ソフトウェアのテストを実施する時には、テストオブジェクトを識別し、それを様々な「視点」から扱いやすい粒度になるまで分解する必要がある。

初回は、仕様を目的機能単位に分解し、それらを FV 表によって整理する方法について解説をおこなった。2 回目の前回は、分解された FV 表の 1 行分、すなわち各々の目的機能に対してその要素と構造を明らかにするためのノートーションであるラルフチャートについて詳細に説明した。

今回は、FV 表においてテストテクニックとして、HAYST 法などの組合せ技法を適用すると決めた目的機能に対して、その入力に当たる因子・水準を整理するために考案された FL 表 (Factor Level Table) について述べる。

冒頭、テストオブジェクトを扱いやすい粒度になるまで分解すると書いたが、HAYST 法においては最終的に、この FL 表にまとめる因子・水準まで分解している。これまで、FL 表を用いて行っているような因子・水準の発見作業は「入力条件分析手法」というカテゴリで研究が進んでおり、1976 年に発表された富士通の「テスト要因分析法」、1988 年の Ostrand & Balcer による”Category-partition method”、1993 年の Grochtmann & Grimm による”Classification-tree method”等がある。

FL 表は、それらの入力条件分析手法を踏まえたうえで、結果としては単に因子・水準を羅列した表になっている。つまり、他の入力条件分析手法のように、提供されるフォーマットに条件を埋めていけば結果が得られるといったものではない。そこで、本稿では、FV 表やラルフチャートの結果を基にして、FL 表をどのように埋めていくのか、そのテストテクニックを明らかにすることを目指す。

2. FV 表およびラルフチャートと FL 表の関係

詳説ラルフチャートで述べたとおり、HAYST 法では、図 1 のラルフチャート (“Ralph’s chart”) および、ソフトウェアの機能 f を式 1 のとおりに定義している。

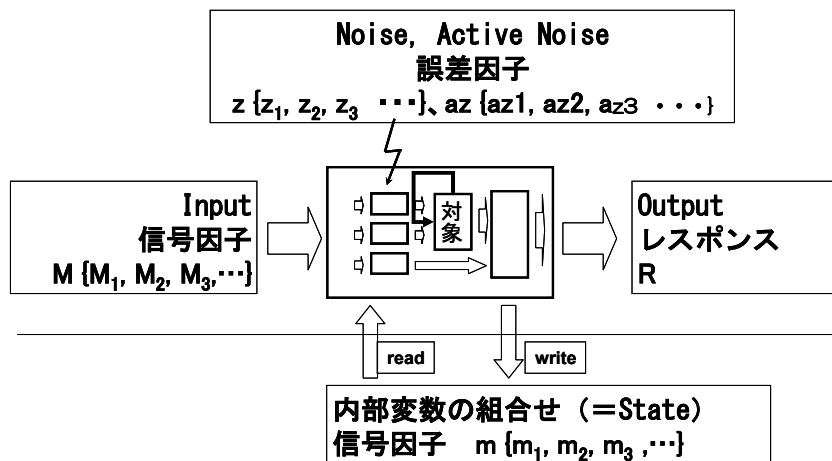


図 1 HAYST 法で使用しているラルフチャート

ソフトウェアの機能 f の定義。

ソフトウェアの機能 f を、出力 R への関数として考えると、式 1 となる。

$$R = f(\text{Input}, \text{State}, \text{Noise}, \text{Active Noise}) \quad \dots\dots\dots \text{式 1}$$

ここで、Input は、テスト対象システムへの入力、State はテスト対象システムの状態、Noise は入出力の関係を妨げる要因、Active Noise は悪意を持ったユーザによるシステム破壊要因（セキュリティの弱点を突いてシステムを破壊する行為など）を示している。

因子・水準とは、式 1 の関数 f の引数である Input, State, Noise, Active Noise を「テストのために具体化」したものとなる。ここで、「具体化」と書いたのは、FV 表およびラルフチャートを用いて発散思考（検討の枠を広げる思考）で広げたテストスコープを集約させることが必要だからである。

機能 f の結果に影響を与える原因には様々なものが存在する。それら結果に影響を与える原因のことを要因と呼ぶ。ソフトウェアテストにおいて、この要因を隈なく洗い出すことは非常に重要である。

FV 表は、開発成果物の中で最も信用がおけるドキュメント（通常は仕様書）を元にして、テストを構築していくことによって、テストに大きな漏れがないことを確認することを目

的としている。ラルフチャートは、さらに、設計・実装情報と市場環境であるノイズ要因を加えることで、要因を出し尽くすことを目的にしている。

さて、このように FV 表とラルフチャートを書くことで、ソフトウェアの動作を左右する要因を多数発見することができる。テストすべき要因としては明確に関係がないことが立証できる要因を除き、ほとんどの要因については組合せテストの因子 (Factor) として採用した方がよい。

しかし、その一方で、テスト時間は限られているため、要因が持つすべての選択肢を組み合わせさせてテストすることはたとえ、HAYST 法を使用したとしても効率的ではない。組合せテストの件数は因子の数よりも因子が持つ選択肢の数に左右されるからである。したがって、FV 表やラルフチャートで挙げた要因について、その選択肢の妥当性を検証する必要がある。

因子は複数の選択肢を持っている。たとえば、スイッチやチェックボックスは、オフ・オンの 2 つの状態を持っている。この因子の持つ状態や値といった選択肢のことを水準 (Level) と呼ぶ。水準には、上記のようにオフ・オンの 2 つだけを持つものから、色 (24bit Color) のような、16,777,216 色といった大きなもの、さらに、路面状況や天候といった分割しようと思えばいくらでも水準を作れるようなものまで幅広く存在する。多数の水準を持つ因子については、同値分割や境界値分析の考え方をを用いて代表値を取り上げて小さな因子にすることが必要である。もし、そうしなければ、色といった大きな因子をテストする時に超巨大なテストマトリクスが必要となる。

このようにして、因子・水準の全体像を明らかにし、実際にテストに用いる因子・水準を一覧表にしたものが FL 表である。FL 表を作ることで組合せテストの検証範囲を明らかにすることができる。

3. FL 表の形式

FL 表とは、”Factor Level Table”の略であり、基本のフォーマットは、表 1 のとおりである。

因子名(Factor)	水準 1	水準 2	...	水準 k	...	水準 n

表 1 FL 表

前章で述べたとおり、因子には、Input, State, Noise, Active Noise の 4 種類がある。FL 表は、種類ごとに作成するが、Noise と Active Noise については、その後の直交表への割り付け方法は同じなので 1 枚にまとめるとよい。つまり、Input, State, Noise(+Active Noise) の 3 枚の FL 表を作成する。

次に、因子の並び順であるが、これは最終的に作成するテストマトリクスの列の並びに対応するため操作（テストオペレーション）がしやすいように並び変えることが重要である。操作がしやすいということは、GUI であれば近くに配置されていることを意味する。近くに配置されているということは、使用目的が近い因子であることが多く、組み合わせで使用するケースが多いものである。

また、あとで、述べるが、場合によっては、複数の因子を一つにまとめるケースがある。したがって、似た因子を近くに配置した方がよい。

次に列方向に水準を配置する。表 2 に具体例を示す。

因子名	水準 1	水準 2	水準 3	水準 4	水準 5
プロトコル	POP	APOP	IMAP	WebMail	
セキュリティ	なし	SSL	TLS		
メールデータ	JIS	UTF-8	HTML	英語添付	日本語添付
送信元のソフト	Outlook	Thunderbird	Becky!		
OS	Win XP	Vista	Windows 7	Win 64bit	

表 2 FL 表（例：メールソフト）

表 2 のとおり、水準は左詰めに埋める。水準の数は因子によってバラバラであるから表の右側は空欄のセルができる。また、水準 2 と 3、水準 4 と 5 の間の罫線が二重になっているのは、HAYST 法の因子の大きさ（水準数）が 2 のべき乗という制約があるためである。同様に水準 8 と 9、16 と 17、32 と 33 といったように、 2^k と 2^{k+1} の間の罫線を二重で書い

ておくとよい。

表 1、表 2 において、列タイトルの「水準 1」にのみ下線が引かれている。この理由は HAYST 法（直交表）では、1 行目にすべて水準 1 が出現するという性質があるからである。

直交表のこの性質を利用するために水準 1 については、その因子のデフォルトの水準をアサインする。こうすることで 1 行目（すなわち 1 回目）のテストがデフォルト状態での動作確認テストとなる。また、単機能テスト（他をデフォルトとして一つだけデフォルト以外の水準を設定したテスト）を除いた組み合わせがより多く出現するためよりよい組み合わせテストマトリクスとなる。

ところで、因子には Input, State, Noise, Active Noise という大きな種類とは別に、水準のタイプを持っている。表 3 は、「水準タイプ」を加えた表になっている。

因子名	水準タイプ	<u>水準 1</u>	水準 2	...	水準 k	...	水準 n

表 3 FL 表

表 4 は水準のタイプ一覧である。

水準タイプ	説明
数値	整数（サイズ、符号付き）、浮動小数など
文字列	ユーザ名など
選択肢	リストやコンボボックスなど
日付	2010 年 7 月 10 日、平成 22 年 7 月 10 日など
時刻	午前 8 時 47 分、22:30 など
通貨	8,000 円、€300、\$2,500 など
指数・対数	2^x や $\log(x)$ など
特別な関数	ガンマ関数など
複合	複数の要因が複合しているもの

表 4 水準のタイプ一覧

注目すべきは、複合(compound)タイプの水準を持つ因子である。表 2 のメールソフトの例でいえば「メールデータ」がそれに当たる。複合タイプの水準については、4.9 節で取り扱う。このように、水準のタイプを明らかにすることで初心者でも過去の FL 表を使って、良い水準を発見することができる。FL 表も FV 表と同様に、EXCEL などの表計算ソフトでまとめることをお勧めする。使い慣れたソフトウェアを用いることで、多くの人が FL 表に目を通し、取り上げた因子・水準に対するレビューすることが重要だからである。

4. FL 表の使い方

本項では、FL 表の使い方について、水準タイプ別に解説する。

4.1 数値

数値については、同値分割した後に境界正常値を取り上げる。たとえば、 $0 < x \leq 10$ （ただし x は整数）であれば、1 と 10 を選択する。というのは、組合せテストであるので異常値である 0 を入れてしまうと、 $x=0$ のテストがすべてエラーになってしまい他の組合せ結果を隠してしまうからである。一般に組合せテストの場合、水準に異常値を含んではならない（異常値については、一つだけ異常値を含むテストを別途実施すること）。

ところで、数値には整数値だけでなく実数値も存在する。実数値の場合、境界値がわかりにくい場合もある。たとえば、 $x < 10.0$ といったケースである。異常境界値として 10.0 (on 点) を取り上げるのは素直であるが、対応する正常境界値 (off 点) について、9.9 でよいのか 9.9999 がふさわしいのか仕様に明確でない場合も多い。このようなときには、変数 d を導入し「 $10.0 - d$ 」としておいて、あとで、 d について 0.1 でよいのか、0.0001 にすべきか検討すればよい。

4.2 文字列

たとえば、ユーザ名として、大文字アルファベット A~Z、小文字のアルファベット a~z、数値 0~9 が許されているとする。4.1 で述べたとおり異常値は含まないので A~Z、a~z、0~9 を用いて水準を作成する。

この時に、長さという観点を忘れずに考慮することが大切である。0 文字（ユーザ名なし）は許されるのか？ 最大長は何文字かを確認して、その文字列長の水準を作成する。

4.3 選択肢

選択肢については、通常はすべて水準とする。というのは、選択肢は 4.9 で述べる複合水準の場合が多いからである。もし、選択肢について取り上げる水準を減らしたいのであれば実装に立ち入って同じとみなしてよい水準を見極める必要がある。

4.4 日付

日付についても、入力可能な日付の両端をテストする。また、閏日について、400 で割り切れる閏年（例：2000 年 2 月 29 日）、100 で割り切れ 400 で割り切れない平年（2100 年 2

月 28 日)、4 で割り切れ 100 で割り切れない閏年 (2012 年 2 月 29 日)、平年 (2011 年 2 月 28 日) は入れてもよいかもしれない。それから、西暦・和暦については、別途因子にした方がよい。

また、日付については期間を指定するものについて注意が必要である。特に、XX 日までといった場合 XX 日は含むのか含まないのか? 終日 (一日中) を指定する方法は、その日とその日+1 日なのか、その日とその日なのかといった点である。外部処理的には後者であるが内部処理的には前者をデータとして持っている場合が多い。9 月 1 日から 9 月 3 日の会議と指定したときには 9 月 3 日に会議は行われるだろう。

4.5 時刻

時刻についても、入力可能な時刻の両端をテストする。また、時刻の場合、12 時間表示と 24 時間表示があるので特に昼と夜の 12 時 (0 時) に注意する。時刻の期間についても日付同様に注意する。会議時間を 1 時から 3 時と指定したときに 3 時 10 分は会議時間に含まないだろう。4.4 の日付のケースと比較しながら考えてほしい。

4.6 通貨

通貨については、日本のように整数値で済むケースと、欧州のように小数値が必要なケースがあることに注意が必要である。また、日本であっても為替の計算のように小数値が必要となるケースも多い。また、桁区切り文字についてもカンマなのかピリオドなのか確認が必要である。

4.7 指数・対数

特に指数の場合、値がすぐに大きくなるため有効値ぎりぎりを選択するとその水準の効果が大きくなりすぎて他の組合せ結果を判定することが困難となる場合がある。そのような場合は、通常使われる値までを組合せテストで実施し、それより大きな値については別途テスト設計するのが良い。

4.8 特別な関数

特別な関数には、特別に使用頻度が高い値や、変化が途切れている場合がある。このような場合は、市場の利用状況を確認する必要がある。

4.9 複合

因子の中には、複数の因子が隠れている場合がある。表 5 に表 2 のメールデータ部分を再掲する。

因子名	水準 1	水準 2	水準 3	水準 4	水準 5
メールデータ	JIS	UTF-8	HTML	英語添付	日本語添付

表 5 FL 表 (例: メールソフトのメールデータ因子: 分離前)

この因子は相手のメールソフトから送られたメールのことであるが、よく見ると、表 6 のような因子・水準が隠れていることがわかる。

因子名	水準 1	水準 2	水準 3	水準 4
本文の文字コード	JIS	UTF-8		
メール形式	プレーン	HTML		
添付データ	なし	あり		
添付ファイル名	英語	日本語		

表 6 FL 表 (例: メールソフトのメールデータ因子: 分離後)

一般的に複合因子は複数の小さな因子に分割した方がよい。今回も元の因子の自由度が 4 であり、分割後の自由度の総和が 4 であり変わらないことから直交表のサイズ見積もりは変わらない。しかも、分割前では出現しない組合せも分割後の因子を使うと組合せが現れることから、より多くのバグが検出できることが期待できる。

というのは、表 5 の水準 1 で示されたメールデータは表 6 でいうと、「JIS+プレーン+なし+ (添付がなしなので禁則)」ということであろう。水準 2 は「UTF-8+プレーン+なし+ (添付がなしなので禁則)」、水準 3 は「JIS+HTML+なし+ (添付がなしなので禁則)」、水準 4 は「JIS+プレーン+あり+英語添付」、水準 5 は「JIS+プレーン+あり+日本語添付」であろう。したがって、たとえば本文が UTF-8 の場合、添付ファイルがついたテストデータはないということである。表 6 のようにすることで、様々な組合せがテストされることになる。

5. FL 表の狙いと効果

5.1 FL 表の狙い

FL 表の狙いは次の 2 点である。

- テスト対象の要因の明確化（因子・水準の抽出）
- テストで選択しなかった要因を明確化する

テスト設計において、なぜその因子・水準を採用したのか、その際に検討に上がったものは他にあるのかといった情報を残すことが非常に重要である。なぜならばリリース後にバグが出たときにテスト設計の不備をレビューできるからである。

5.2 FL 表の効果

FL 表を使用することで、HAYST 法で取り扱う因子・水準が明確になる。また、FL 表を蓄積することによって、水準タイプごとにどのようなテストデータを用意しなければならないのかの情報が共有される。この情報はテストの初心者に対して、水準タイプごとに何をテストすべきかの、よいナビゲーションとなる。

FL 表を使う時に一つ注意点がある。それは、水準の異常値については記載していないという点である。これは、組合せテストで使用する入力としては当然なことであるが、因子について異常水準をテストしなくてよいというわけではない。むしろ、単機能テストとしては異常水準値こそ、十分なテストが必要である。

FL 表が一通り完成したら、それをコピーし、異常水準のみが記載された「裏 FL 表」を作成し異常値テストをすることをお勧めする。

本稿がテストによるソフトウェア品質の向上に少しでも役立てば幸いである。

以上