

有限射影幾何を用いたソフトウェアテスト向けの直交表 自動生成プログラムの開発とその応用

須田 健二¹ 五味 弘^{2,a)}

受付日 2015年10月28日, 採録日 2016年5月17日

概要: 直交表をソフトウェアテストに応用する動きがさかんである。しかし、強さ2の直交表を構成するのは比較的簡単だが、強さ3以上で、多因子、多水準の直交表の構成は簡単ではなく、そのことが、多因子、多水準を持つソフトウェアテストの組合せテスト設計を困難にしていた。そこで我々は、多因子(50因子)、多水準(2, 3, 4, 5, 7, 8, 9水準)、強さ2, 3, 4に対応し、さらに混合水準(因子の水準が異なり、その因子の水準が2や3のべき乗)の直交表を有限射影幾何を用いて組織的にしかも自動的に構成できるソフト **GaloisSoftTest** を開発した。したがって、**GaloisSoftTest** によって生成される直交表は、2因子間、3因子間、4因子間のすべての組合せのチェックを保証するテストが可能であり、またソフトウェアテストでよく用いられる混合水準にも対応しており、より少ないテスト回数で高品質な組合せテスト設計に応用が可能である。さらに本論文では、**GaloisSoftTest** と代表的なオールペア法の生成ソフトである PICT に対して、因子数やその水準数、強さが与えられたときの効率性の尺度であるテスト回数と品質を保証する尺度である網羅率を求め比較分析した。その結果、**GaloisSoftTest** が優れているテスト対象もあり、この2つのツールを並行運用して、効率的でしかも品質レベルを保証する組合せテストを運用する方針や施策などについても提案している。

キーワード: ソフトウェアテスト, 直交表, 生成アルゴリズム, 有限射影幾何, テスト戦略, 組合せ理論

The Development of a Program to Automatically Construct Orthogonal Arrays for Software Testing by Using Finite Projective Geometry and its Applications

KENJI SUDA¹ HIROSHI GOMI^{2,a)}

Received: October 28, 2015, Accepted: May 17, 2016

Abstract: An orthogonal array is widely used in software testing. But either orthogonal array which are strength 2 is used in software testing. In a practical use, it is necessary to construct orthogonal arrays which are strength 2, 3 and strength 4. In such situations, we developed the computer program **GaloisSoftTest** by using finite projective geometry that can automatically construct orthogonal arrays which supports the strength of 2 and over, and multi-factor, multi-level, mixed-level. Furthermore we studied to the number of test and covering efficiency which are cost performance, to compare the tool **GaloisSoftTest** for generating orthogonal arrays with PICT which is tool for generating pairwise testing. We make a guideline for selecting a method and its strength, factor, level in orthogonal arrays and pairwise testing, and then we execute a suitable software-testing method.

Keywords: Software testing, orthogonal array, finite projective geometry, testing strategy, combinatorial theory

¹ 元群馬工業高等専門学校
Former workplace; Gunma National College of Technology,
Maebashi, Gunma 371-8530, Japan

² 沖電気工業株式会社
Oki Electric Industry Co., Ltd., Warabi, Saitama 335-8510,
Japan

a) gomi521@oki.com

1. はじめに

直交表は、実験計画法で培われた手法であるが、近年、これがソフトウェアテストに応用されてきている。わが国では田口 [1] が開発した直交表と線点図を利用して必要な

直交表を構成する方法が広く行われてきている。しかし利用できる直交表は、水準数が2か3、強さも2に限定されるなど幅広い実験に対応できず、また因子数が多くなると実際大変な手間がかかるなど、実験の初期の目的にかなうような直交表を構成することは簡単ではなかった。

このような問題に対して、高橋 [2], [3] は直交表を有限体上の射影幾何（有限射影幾何）を利用して構成する方法を提唱した。しかし基礎理論はあっても実際に直交表を構成することは簡単ではなかった。そこで、Fuji-hara [4] や須田ら [5] が実験計画法向けの直交表をコンピュータで自動的に構成するためのアルゴリズムを開発し実用化した。これによって実験計画法向けの直交表の生成は非常に容易になったが、ソフトウェアテスト向けの直交表に必要な多因子、多水準で強さも2だけでなく3や4にも対応するような直交表の生成は困難な問題であった。

そこで我々は、まず多水準に対応させるため、最初に素数のべき乗水準に対応できるように必要な原始規約多項式をいくつか新規に求めた。また、条件が与えられたときにテスト回数が最小という意味で最良な直交表を構成するために必要な有限射影幾何の次元を決定する方法について検討し、有限射影幾何を利用して強さ2, 3, 4の直交表を生成できるアルゴリズムを開発した。さらに、因子の水準が異なる混合水準の直交表も有限射影幾何を利用して自動的に生成できるアルゴリズムを開発した。そのうえで、ソフトウェアテスト向けに多因子（50因子）、多水準（2, 3, 4, 5, 7, 8, 9（素数と素数のべき乗水準））、強さ2, 3, 4に対応し、さらに、混合水準の直交表を有限射影幾何を利用して自動的に生成できるソフト **GaloisSoftTest**（ガロアソフトテスト）を開発した。**GaloisSoftTest**によって生成される直交表は、2因子間、3因子間、4因子間のすべての組合せを保証するテストが可能であり、またソフトウェアテストでよく用いられる混合水準にも対応しているので、より広いテスト対象に対し、より少ないテスト回数で高品質な組合せテストが可能である。

さらに本論文では、**GaloisSoftTest**と代表的なオールペア法の生成ソフトである PICT に対して、因子数やその水準数、強さが与えられたときの効率性の尺度であるテスト回数と品質を保証する尺度である網羅率を求め比較分析した。その結果、テスト回数と網羅率において **GaloisSoftTest** が優れているテスト対象があることが分かり、この2つのツールを並行運用して、効率的でしかも品質レベルを保証する組合せテストを運用する方針や施策などについても提案している。

本論文は、2章で有限射影幾何を用いる直交表の構成法、3章でその具体的な生成アルゴリズム、4章で直交表の自動生成プログラムと生成例、5章で生成された直交表の良さを示すためにオールペア法との比較、そして6章で直交表とオールペア法の選択基準と並行運用について記述した構成となっている。

2. 有限射影幾何を用いる直交表の構成法

2.1 直交表とは

直交表は次のように定義される。S を q 個の値からなる集合で、 $S = \{0, 1, \dots, q-1\}$ とする。強さ t の直交表は、 $m \times N$ 行列 **A** (**A** の要素の値は水準と呼ばれ S の元である) で、次の条件を満たすとき、**A** は実験回数 N, 因子数 m, 水準数 q, インデックス λ の直交表 (**Orthogonal Array**) であるといい、 $OA(N, m, q, t)$ と記す。

条件：任意の t 行に対し、すべての水準組合せ (S^t の順序対) が同数個 (λ 個) の列に現れる。

なお、この直交表をソフトウェアテストで使用するときには、実験回数はテスト回数、因子数は項目や引数の値、水準数は項目のとり値、強さ t は t 個の因子間の全組合せのテストを保証（網羅率 100%）することを意味する。上記定義は、暗黙のうちに各因子の水準数は一定（揃っている）であることを意味している。この形の直交表をここでは基本形といい、表 1 にテスト回数 8, 因子数 4, 水準数 2, 強さ 3 の直交表である $OA(8, 4, 2, 3)$ の例を示す。表 1 は強さ 3 の直交表の例であり、どの 3 行をとっても、因子の水準組合せである、000, 001, ..., 110, 111 が 1 回ずつ（同数個）列に出現している。

また、各因子の水準数が不揃いであるものを混合形と呼び、次のように記すことにする。

$$OA(N, (m_1, q_1), (m_2, q_2), \dots, (m_n, q_n), t)$$

これは、 m_1 個の因子が q_1 水準、 m_2 個の因子が q_2 水準、 m_n 個の因子が q_n 水準を持つことを意味する。以下の議論ではこの不揃いの水準数は q のべき乗だけのものに制約する。すなわち、 $q_1 = q^{r_1}, q_2 = q^{r_2}, \dots, q_n = q^{r_n}$ であり、 $q_1 > q_2 > \dots > q_n$ とする。これを q 水準系の混合形と呼ぶことにする。

2.2 直交表の構成法

このような直交表を有限射影幾何を用いて構成する方法を高橋 [3] が提唱している。しかし基礎理論はあっても具体的な条件に適した直交表を構成することは容易ではなかった。そこで、我々はソフトウェアテスト向けの直交表を有限射影幾何を用いて組織的に構成する方法を確立したので、以下に述べる。

表 1 直交表の例 $OA(8, 4, 2, 3)$

Table 1 The example of orthogonal array $OA(8, 4, 2, 3)$.

	1回	2回	3回	4回	5回	6回	7回	8回
因子1	0	0	0	0	1	1	1	1
因子2	0	0	1	1	0	0	1	1
因子3	0	1	0	1	0	1	0	1
因子4	0	1	1	0	1	0	0	1

(1) G 行列から直交表を生成

高橋 [3] が提唱したのは、ある条件を満たすような行列 G を作る事ができれば、直交表を構成できる方法である。すなわち、水準数 q が一定で、しかも素数のべき乗であるなら、有限体 $GF(q)$ 上に、ある性質を持つ $m \times n$ 行列 G を考え、その条件を満たす G 行列から、実験回数が q^n である直交表が次式で得られる。

$$A = \{r = G\theta; \theta \in GF(q)^n\} \quad (1)$$

ここで、 r は 1 回の実験 (テスト) の組合せを示す列ベクトルである。

(2) ソフトウェアテストに必要な G 行列

ソフトウェアテストにおける、2 因子間、3 因子間、4 因子間の全組合せテストを可能にするためには強さ 2, 3, 4 の直交表が必要である。したがって、基本形と混合形の直交表を生成するためには、次のような G 行列が必要となる。

- (a) 強さ 2 の直交表を作るには、 G の任意の 2 行は、線形独立である G 行列を作る。
- (b) 強さ 3 の直交表を作るには、 G の任意の 3 行は、線形独立である G 行列を作る。
- (c) 強さ 4 の直交表を作るには、 G の任意の 4 行は、線形独立である G 行列を作る。
- (d) 混合形の直交表は、実用的かつ実装しやすいという観点から q 水準系で強さ 2 に限ることにする。このような直交表を生成するための G 行列の構成法は次のようになる。

(i) 最も大きいべき乗の値 r_1 を持つ因子を、線形独立である r_1 個のベクトルに対応させ G 行列に入れる。これは直交表を生成したあとで q 進 r_1 桁を整数に変換するためである、これを m_1 個の因子に対して行う。ただし、すでに選んだベクトルの集合と互いに線形従属とならないように選ぶ。

(ii) 次に大きいべき乗の値 r_2 を持つ因子に対して同様に線形独立な r_2 個のベクトルを選び G 行列に入れる。これを m_2 個の因子に対して行う。このときすでに選んだすべてのベクトルの集合と線形従属であってはならない。

(iii) 同様な方法で、最も小さいべき乗を持つ因子までベクトルが選べれば、それが求める G 行列である。

(3) 有限射影幾何の点 (0-フラット) の表現と直線 (1-フラット)、平面 (2-フラット) と t -フラット上の点の求め方

有限射影幾何 $PG(n-1, q)$ は有限体 $GF(q)$ 上の拡大体 $GF(q^n)$ を構成することにより代数的に構成できることが知られている [3]。すなわち、有限体 $GF(q)$ 上の n 次原始既約多項式を利用して、その原始根を α として、 α のべき乗表現ですべての拡大体 $GF(q^n)$ の元が表現できる。そこで、 $PG(n-1, q)$ の点を次のように定義すれば、

$$\{[x_1, x_2, \dots, x_n] = [\lambda x_1, \lambda x_2, \dots, \lambda x_n], \lambda \neq 0, \lambda \in GF(q)\} \text{ の非零元} \quad (2)$$

$PG(n-1, q)$ の点は、拡大体 $GF(q^n)$ の元で表現でき、 $GF(q)$ 上の n 次元ベクトルとしても表現できる。また、2 点 ξ_i と ξ_j を通る直線 $\xi_i \xi_j$ 上の点は次式で簡単に求めることができる。

$$\xi_i \xi_j = \{\xi_i + \lambda \xi_j; \lambda \in GF(q)\} \cup \{\xi_j\} \quad (3)$$

平面上の点は、 $PG(n-1, q)$ 上の独立 (同一直線上にない) な 3 点 ξ_i, ξ_j, ξ_k が与えられると、それを含む平面が決定されるが、式 (3) を用いて、まず直線 $\xi_i \xi_j$ を計算し、その直線上のすべての点と、 ξ_k の間ですべての直線を計算することにより、すべての平面上の点を求めることができる。

t -フラット上の点は、次のように再帰アルゴリズムによって、直線の計算のみで求めることができる。 t 個の点が与えられて、 $t-1$ -フラット上の点がすべて求まっているとすると、残りの 1 個の点と $t-1$ -フラット上のすべての点の間で直線を計算すれば、 t -フラット上のすべての点を求めることができる。

(4) G 行列の構成に有限射影幾何を利用

高橋 [3] は G 行列を構成する方法として、因子 F_1, F_2, \dots, F_m を有限射影幾何 $PG(n-1, q)$ の点に割り付ける方法を示した。そこで、我々はソフトウェアテスト向けの直交表を生成するための G 行列を有限射影幾何を用いて組織的に構成する方法を確立した。すなわち、 $PG(n-1, q)$ の点は、 $GF(q)$ 上の n 次元ベクトルで表現できるので、以下のようにして G 行列を構成する。

- (a) $PG(n-1, q)$ の相異なる点 (ベクトル表現) から G 行列を作ると、強さ 2 の直交表が構成できる。したがって、すべての点から G 行列を作れば、因子数 $m = (q^n - 1)/(q - 1)$ なる直交表が構成できる。
- (b) 強さ 3 の直交表を作るには、3 点が同一直線上にない点の集合から G 行列を作る必要がある。
- (c) 強さ 4 の直交表を作るには、4 点が同一平面上にない点の集合から G 行列を作る必要がある。
- (d) q 水準系で強さ 2 の混合形直交表を作るには、べき乗の値 r_1 を持つ因子に対応させる r_1 個の点から生成される点の集合と、他のすべてのべき乗の値 r_2 (r_1 のべき乗のものがまだあればそれも含む) を持つ因子に対応させる r_2 個の点から生成される点の集合が、互いに素となるような点の集合から G 行列を作る必要がある。

2.3 有限射影幾何を用いた直交表の構成例

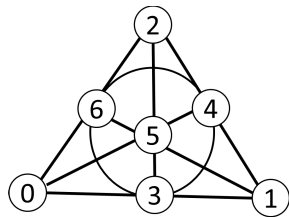
有限体 $GF(2)$ 上の 3 次拡大体 $GF(2^3)$ は、3 次原始既約多項式 $f(x) = x^3 + x + 1 = 0$ の根を α とすれば、 $\alpha^3 = 1 + \alpha$ となり、 α のべき乗で表現すると、 $GF(2)$ 上の 3 次元ベクトル全体 $GF(2)^3$ が表現できる。すなわち、

$$GF(2)^3 = \{[x_1, x_2, x_3] : x_i \in GF(2), 1 \leq i \leq 3\} \quad (4)$$

表 2 3次拡大体 $GF(2^3)$ と $PG(2,2)$ の点の表現

Table 2 The representation of $GF(2^3)$ and the points of $PG(2,2)$.

点	べき乗表現	多項式表現	ベクトル表現
0	α^∞ α^0	$=0$ $=1$	$(0, 0, 0)$ $(1, 0, 0)$
1	α^1	$= \alpha$	$(0, 1, 0)$
2	α^2	$= \alpha^2$	$(0, 0, 1)$
3	α^3	$=1+\alpha$	$(1, 1, 0)$
4	α^4	$= \alpha+\alpha^2$	$(0, 1, 1)$
5	α^5	$=1+\alpha+\alpha^2$	$(1, 1, 1)$
6	α^6 (α^7)	$=1 + \alpha^2$ $=1$	$(1, 0, 1)$



直線
0: 0-1-3
1: 1-2-4
2: 2-3-5
3: 3-4-6
4: 4-5-0
5: 5-6-1
6: 6-0-2

図 1 $PG(2,2)$ の点と直線

Fig. 1 The 7 points and 7 lines in $PG(2,2)$.

表 3 強さ 2 の G 行列

Table 3 G -matrix of strength 2.

因子	点	G 行列
F_1	0	1 0 0
F_2	1	0 1 0
F_3	2	0 0 1
F_4	3	1 1 0
F_5	4	0 1 1
F_6	5	1 1 1
F_7	6	1 0 1

となり、また $GF(2^3)$ は表 2 のように α のべき乗表現、多項式表現、ベクトル表現の 3 種類で表現できる。

表 2 の 3 次拡大体 $GF(2^3)$ において、零元 $\alpha^\infty = 0$ を除いた残り ($\alpha^0 \sim \alpha^6$) が $PG(2,2)$ の 7 個の点を表現しており、これを簡単にべき乗だけを取り (0~6) と整数で表す。また、 $PG(2,2)$ の 2 点 0 と 1 を通る直線上の他の点は上記式 (3) より 3 であることが分かり、 $PG(2,2)$ では直線の数は全部で 7 本である。これらの関係をグラフに表現するとよく知られた図 1 のようになる。次に、上記の有限射影幾何 $PG(2,2)$ を利用して、実際に G 行列を生成し、直交表を構成する。

(1) 強さ 2 の直交表の構成例

有限射影幾何 $PG(n-1, q)$ の点は、その構成からも分かるように、どの 2 点をとっても線形独立なベクトルとなっている。上記の $PG(2,2)$ では、点の数は 7 個あるので、強さ 2 で割り付け可能な最大因子数は 7 つである。この 7 点のベクトル表現を使って G 行列を構成すれば表 3 のよう

表 4 強さ 3 の G 行列

Table 4 G -matrix of strength 3.

因子	点	G 行列
F_1	0	1 0 0
F_2	1	0 1 0
F_3	2	0 0 1
F_4	5	1 1 1

表 5 混合形の G 行列と直交表

Table 5 Mixed G -matrix and Mixed orthogonal array.

因子	点	G 行列	直交表				
			F1	F2	F3	F4	F5
F_1	0	1 0 0	0(00)	0	0	0	0
	1	0 1 0	0(00)	1	1	1	1
F_2	2	0 0 1	1(01)	0	1	1	0
	4	0 1 1	1(01)	1	0	0	1
F_4	5	1 1 1	2(10)	0	0	1	1
	6	1 0 1	2(10)	1	1	0	0
F_5			3(11)	0	1	0	1
			3(11)	1	0	1	0

但し、直交表は転置していることに注意

になり、式 (4) から直交表 $OA(2^3, 7, 2, 2)$ が生成される。この直交表はよく知られた記法で書けば $L_8(2^7)$ である。

(2) 強さ 3 の直交表の構成例

強さ 3 の直交表を作るには、3 点が直線上にのらない点の集合を集める必要があり、上記 $PG(2,2)$ でこの条件を満たすのは図 1 から分かるように、点 0, 1, 2, 5 の 4 点であり、この点の数 4 が割り付け可能な最大因子数となる。これから G 行列を構成すれば表 4 のようになり、式 (1) から 2 ページの表 1 の直交表 $OA(2^3, 4, 2, 3)$ が生成される。

(3) 混合形の直交表の構成例

$PG(2,2)$ 上で、4 水準が 1 個、2 水準が 4 個の直交表 $OA(2^3, (1, 4), (4, 2), 2)$ を構成することを考える。これは 4 (2^2) 水準の因子が 1 個なので、この因子に 2 点 0 と 1 を割り付け、0 と 1 の直線上の点は 3 なので、この点 3 は他の因子に割り付けられない。さらに 2 水準の因子が 4 個あるので、これをまだ割り付けてない点 4 個に割り付け、これで G 行列を構成すれば表 5 のようになり、式 (1) から直交表 $OA(2^3, (1, 4), (4, 2), 2)$ が生成できる。

3. ソフトウェアテスト向けの直交表の自動生成アルゴリズム

有限射影幾何 $PG(n-1, q)$ の点を生成するためには有限体 $GF(q)$ 上の原始規約多項式が必要である。そのために我々は q が素数である 2, 3, 5, 7 に対しては、文献 [3], [4] の原始規約多項式を利用したが、 q が素数のべき乗である $2^2, 2^3, 3^2$ に対しては新規に求める必要があった。そこで、高橋 [3] の示す方法を参考にして我々が新たに求めた原始規約多項式を表 6 に示す。最初の式は $GF(q^m)$ 上の拡大

表 6 新規に求めた原始規約多項式

Table 6 primitive irreducible polynomials.

PG(1, 2 ²)	$x^2 = y + x, y^2 = 1 + y$
PG(2, 2 ²)	$x^3 = y + y^2x + x^2, y^2 = 1 + y$
PG(3, 2 ²)	$x^4 = y + yx + yx^2 + x^3, y^2 = 1 + y$
PG(1, 2 ³)	$x^2 = y + y^3x, y^3 = 1 + y^2$
PG(2, 2 ³)	$x^3 = y^4 + x + y^6x^2, y^3 = 1 + y$
PG(3, 2 ³)	$x^4 = y + y^2x + y^2x^2 + y^3x^3, y^3 = 1 + y^2$
PG(1, 3 ²)	$x^2 = y^5 + y^2x, y^2 = 1 + 2y$
PG(2, 3 ²)	$x^3 = y + y^5x + y^2x^2, y^2 = 1 + 2y$

表 7 強さ 3 で選ぶことができる最大因子数

Table 7 The maximum number of points which is 3-linearly independent set in $PG(n-1, q)$.

水準数 q	次元 n	テスト回数 q	最大因子数
3	3	3 ³	4*
3	4	3 ⁴	10*
3	5	3 ⁵	20
3	6	3 ⁶	56
5	3	5 ³	6*
5	4	5 ⁴	26*
5	5	5 ⁵	60
7	3	7 ³	8*
7	4	7 ⁴	50*

*付きは最大が保証されていることを示す。表 8 も同じ。

体 $GF((q^m)^n)$ を構成するためのもので、次の式が $GF(q)$ 上の拡大体 $GF(q^m)$ を構成するための原始規約多項式である。

3.1 基本形の直交表を生成するアルゴリズム

ここでは、因子数 m 、水準数 q 、強さ t が条件として与えられたときのアルゴリズムについて述べる。

最初に (1) の有限射影幾何 $PG(n-1, q)$ の次元を求め、点を生成して、次に直交表の強さにより (2) または (3)、(4) を実施して直交表を生成する。

(1) 有限射影幾何 $PG(n-1, q)$ の次元の決定と点の生成

各因子を割り付ける有限射影幾何の次元は次のように決定される。

(A1) 有限射影幾何 $PG(n-1, q)$ の点の数は $(q^n-1)/(q-1)$ であるので、強さ 2 の場合は、因子数 m がこの点の数を超えない最小の次元 n として決まる。強さが 3 の場合は、 $PG(n-1, 2)$ の場合のみ解決しており、因子数 m が 2^{n-1} を超えない最小の次元 n として決まる。すなわち、因子数が 4、水準数が 2、強さが 3 の直交表は、テスト回数 8 で構成できるが、因子数が 5~8 になると、テスト回数は 16 となることを意味する。その他の強さ 3 や強さ 4 に対しては、必要な次元の決定は難しい問題であり、ほとんど未解決の問題となっている。しかし、我々はコンピュータサーチによって以下の表 7、表 8 に示すような、テスト回数によって割り付けできる最大の因子数を求めており [6]、これにより、割り付けに適した次元を決定することができる。

表 8 強さ 4 で選ぶことができる最大因子数

Table 8 The maximum number of points which is 4-linearly independent set in $PG(n-1, q)$.

水準数 q	次元 n	テスト回数 q^n	最大因子数
2	4	2 ⁴	5*
2	5	2 ⁵	6*
2	6	2 ⁶	8*
2	7	2 ⁷	11*
2	8	2 ⁸	17*
3	4	3 ⁴	5*
3	5	3 ⁵	11*
3	6	3 ⁶	14
3	7	3 ⁷	21
5	4	5 ⁴	6*
5	5	5 ⁵	12
5	6	5 ⁶	23
7	4	7 ⁴	8*
7	5	7 ⁵	16

さらに必要に応じて次元を直接に入力できる機能を持たせている。

(A2) $GF(q)$ 上の n 次原始規約多項式から、 $PG(n-1, q)$ の点を生成する。

(2) 強さ 2 の直交表

強さ 2 の直交表を作るには、 G の任意の 2 行が、線形独立である G 行列を作ればよく、 $PG(n-1, q)$ 上の各点はどの 2 点をとっても線形独立であるので、アルゴリズムは次のようになる。

(B1) 各因子 F_i ($i = 1 \sim m$) を $PG(n-1, q)$ 上の相異なる点 ξ_i に割り付けて G 行列を構成する。すなわち、必要な因子数 m 個分の点を選んで G 行列を作ればよい。

(3) 強さ 3 の直交表

強さ 3 の直交表を作るには、 G の任意の 3 行が、線形独立である G 行列を作ればよく、 $PG(n-1, q)$ 上の 3 点が直線上にない点の集合を G 行列とすればよいので、アルゴリズムは次のようになる。

(C1) G 行列を空にして、 $PG(n-1, q)$ の点のすべての covering 数を 0 に初期設定する。

(C2) G 行列に $PG(n-1, q)$ の点で小さい順に n 個の点 (0, 1 から $n-1$) を入れる。表 2 では F_1 に 0 を F_2 に 1 を、 F_3 に 2 を割り付けて G 行列に入れている。

(C3) G 行列の任意の 2 点 ξ_i, ξ_j を選ぶ。

(C4) 2 点 ξ_i, ξ_j を通る直線を計算する。

(C5) その直線上にある点の covering 数を 1 増やす (なお、 G 行列の中のすべての 2 点の組合せについて、(C3)~(C5)を繰り返す)。

(C6) covering 数が 0 の点を 1 つ G 行列に入れる。covering 数が 0 の点がいくつかある場合の、どの点を選ぶかという問題については、文献 [5] で述べているが、通常は点の小さい順に選ぶ。

(C7) 今選んだ点 ξ_k と G 行列の任意の点 ξ_i ($\neq \xi_k$) に関して, (C4)と(C5)を繰り返す (これをすべての G 行列の点に対して繰り返す).

(C8) G 行列の中の点の数が, 条件として与えられた因子数に達していれば, そのときの G 行列が求めるものである. もし達していなければ, 次の(C9)へ進む.

(C9) まだ, covering 数が 0 の点があれば, (C6)と(C7)を繰り返す. covering 数が 0 の点がなければ, 入れ替えアルゴリズム (バックトラック) に進み, 1つ前の点の選択を covering 数が 0 の他の点に置き換えて(C7)からやり直す.

(4) 強さ 4 の直交表

強さ 4 の直交表を作るには, G の任意の 4 行は, 線形独立である G 行列を作ればよく, $PG(n-1, q)$ 上の 4 点が同一平面上にない点の集合を G 行列とすればよいので, アルゴリズムは, (3) の強さ 3 の直交表を求めるアルゴリズムで 2 点を 3 点に置き換え, さらに直線の計算を平面の計算に置き換えればよいので詳細は省略する.

3.2 混合形の直交表を生成するアルゴリズム

ここでは q 水準系の混合形で強さ 2 の直交表を生成するアルゴリズムについて述べる. ここで混合形的水準数はある素数 q のべき乗の形のものに限り, $q_1 = q^{r_1}$, $q_2 = q^{r_2}$, \dots , $q_n = q^{r_n}$ ($r_1 > r_2 > \dots > r_n$) とし q_1 水準の因子数が m_1 個, q_2 水準の因子数が m_2 個, \dots , q_n 水準の因子数が m_n 個とする.

(1) 有限射影幾何 $PG(n-1, q)$ の次元の決定と点の生成

各因子を割り付ける有限射影幾何 $PG(n-1, q)$ の次元 n は次のように決定する.

(A1) $PG(n-1, q)$ の点の数は, $(q^n - 1)/(q - 1)$ である.

(A2) 水準数 $q_1 = q^{r_1}$ とすると, この因子 1 つに必要な $PG(n-1, q)$ の点の数は $(q^{r_1} - 1)/(q - 1)$ であり, 因子が m_1 個あるので, 必要な点の数は $(q^{r_1} - 1)/(q - 1) \times m_1$ である.

(A3) 同様に, 水準数 $q_n = q^{r_n}$ の m_n 個の因子を $PG(n-1, q)$ の点に割り付けると必要な点の数は $(q^{r_n} - 1)/(q - 1) \times m_n$ である.

(A4) したがって, 有限射影幾何 $PG(n-1, q)$ の次元 n は次の不等式を満たす最小の n として決定される.

$$(q^n - 1)/(q - 1) \geq (q^{r_1} - 1)/(q - 1) \times m_1 + \dots + (q^{r_n} - 1)/(q - 1) \times m_n$$

(A5) $GF(q)$ 上の n 次原始既約多項式から, $PG(n-1, q)$ の点を生成する.

(2) 必要な G 行列を求めるアルゴリズム

(B1) G 行列を空にして, $PG(n-1, q)$ の点のすべての covering 数を 0 に初期設定する.

(B2) まず最大の水準数 $q_1 = q^{r_1}$ を持つ因子 1 を $PG(n-1, q)$ の点に割り付ける. そのやり方は covering 数が 0 の点か

ら, r_1 個の点を選び, $(r_1 - 1)$ -フラットを生成し, そのフラット上の点を求め, その点の covering 数を 1 増やす. r_1 個の点を選んだ順に G 行列に入れる. $q_1 = q^{r_1}$ 水準を持つ残りの因子 2 から因子 m_1 まで同様に割り付ける. その際に $(r_1 - 1)$ -フラット上の点の covering 数がすべて 0 なら, その点を選んだ順に G 行列に入れ, その点の covering 数を +1 し, もし covering 数が 0 でない点があれば点を選び直す.

(B3) 次に大きい水準数 $q_2 = q^{r_2}$ を持つ因子を (B2) と同様に covering 数が 0 の点から, r_2 個の点を選び, $(r_2 - 1)$ -フラットを生成しすべての点を求める. その際に $(r_2 - 1)$ -フラット上の点の covering 数がすべて 0 なら, その点を選んだ順に G 行列に入れ, その点の covering 数を +1 し, もし covering 数が 0 でない点があれば点を選び直す. 次に, $q_2 = q^{r_2}$ 水準を持つ残りの因子 2 から因子 m_2 まで同様に割り付け, その点を選んだ順に G 行列に入れる.

(B4) 残りの水準数を持つ因子を (B2), (B3) と同様に $PG(n-1, q)$ の点に割り付け, 割り付けた点の covering 数を 1 増やす. また点を選んだ順に G 行列に入れる.

(B5) すべての水準数のすべての因子が $PG(n-1, q)$ の点に割り付けられれば, そのときの G 行列が求めるものである. まだ割り付けができない因子があれば, (B6)へ進む.

(B6) すべての因子を割り付ける前に covering 数が 0 の点が無くなってしまった場合は, 入れ替えアルゴリズム (バックトラック) に進み, 1つ前の因子を割り付ける点の選択を選び直し(B4)から, やり直す.

4. 直交表の自動生成プログラムとその生成例

前章で述べてきたアルゴリズムに基づいて, 直交表の自動生成プログラムを開発したので, 生成した直交表の例とともに紹介する.

4.1 基本形の直交表を生成するプログラムと生成例

基本形の直交表を生成するプログラム「GaloisSoft Test [7], [8]」を Visual C++ で Windows 上に開発した. ソフトウェアテストで利用しやすいようにコンソールから引数を入力し, それを画面出力とともに CSV ファイルを生成するものと GUI 画面で操作するものを両方提供している. CSV で生成している版は PICT 互換にしている. バックトラックとしては総当たりするものと評価値をもとに総当たりする順序を変更するプログラムを実装している. このツールによって生成できる直交表のサイズはソフトウェアテストで十分な大きさであると考えているが, さらに大きいサイズが必要であれば, 対応する原始既約多項式を入れればいくらか大きくできる.

次にこのプログラムで生成した直交表の一例を示す.

(1) $OA(243, 20, 3, 3)$

テスト回数が 243 で 20 因子 3 水準で, 強さが 3 の直交

G 行列				
1	0	0	0	0
0	1	0	0	0
0	0	1	0	0
0	0	0	1	0
0	0	0	0	1
2	1	0	0	2
1	1	1	0	0
0	1	1	1	0
0	0	1	1	1
2	1	0	1	1
1	1	2	2	0
0	1	1	2	2
2	2	0	2	1
1	1	1	2	1
2	0	2	1	1
1	0	2	0	1
2	1	1	1	2
1	0	2	1	0
1	2	1	0	2
0	1	0	2	1

図 2 直交表ツール GaloisSoftTest の実行例 1
 Fig. 2 The execution example 1 for GaloisSoftTest.

表を生成した。これは $PG(4,3)$ 上の点に、20 個を超える因子を割り付けることができないという意味で最良の直交表である。生成した 20 個の点の G 行列を図 2 に示す（直交表は大きいので省略する）。

(2) $OA(128, 11, 2, 4)$

11 因子 2 水準強さ 4 テスト回数 128 の直交表であり、 $PG(6,2)$ 上の点に、11 個を超える因子を割り付けることができないという意味で最良の直交表である。

4.2 混合形の直交表を生成するプログラムと生成例

混合形の直交表を生成するプログラムを前章のアルゴリズムをもとに Java で実装した [9]。直交表のサイズ内で多くの点を使う直交表を生成する場合はバックトラックが多くなり時間がかかるが、バックトラックを工夫して高速に実行している。つまりバックトラックアルゴリズムは単純な順番どおりの総当たりから評価値により動的にアルゴリズムを変更するものまで多種のバックトラックプログラムを実装している。また生成した直交表が妥当であるかのチェックをするプログラムも作成して確認している。

またこのプログラムのインタフェースは PICT [10] 互換にしている。PICT 互換にしたことにより、PICT のフロントエンドプロセッサである PictMaster [11] からも操作できる。実行例を図 3 に示す。また図 3 にテストケースを生成するとき生成した直交表（ただし行と列を転置している）も示す。この生成した直交表をもとに入力の水準を割り当ててテストケースを生成している。この例では、4 水準が 2 因子、2 水準が 2 因子を $PG(3,2)$ の点に割り付け、直交表 $OA(16, (2, 4), (2, 2), 2)$ を自動生成している。また、水準に不足があるときは、水準の組合せが同数個になる（これを均一と呼ぶ）出現を満たさなくなるが、自動的に水準を挿入する。

このプログラムではもっと大きな混合形の直交表も生成でき、その例を示す。

(input.csv PICT互換の入力ファイル)			
在住: 市内, 市外			
性別: 男性, 女性			
年齢: 20代, 30代, 40代, 50代			
趣味: 漫画, 旅行, ブログ, 写真			
> GaloisSoftTest input.csv /o:2 (PICT互換のコマンド)			
年齢	趣味	在住	性別
20代	漫画	市内	男性
20代	ブログ	市内	女性
20代	旅行	市外	男性
20代	写真	市外	女性
40代	漫画	市外	女性
40代	ブログ	市外	男性
40代	旅行	市内	女性
40代	写真	市内	男性
30代	漫画	市内	女性
30代	ブログ	市内	男性
30代	旅行	市外	女性
30代	写真	市外	男性
50代	漫画	市外	男性
50代	ブログ	市外	女性
50代	旅行	市内	男性
50代	写真	市内	女性
(PICT 互換の出力)			

生成した直交表				
0	0	0	0	
0	2	0	1	
0	1	1	0	
0	3	1	1	
2	0	1	1	
2	2	1	0	
2	1	0	1	
2	3	0	0	
1	0	0	1	
1	2	0	0	
1	1	1	1	
1	3	1	0	
3	0	1	0	
3	2	1	1	
3	1	0	0	
3	3	0	1	

図 3 直交表ツール GaloisSoftTest の実行例 2
 Fig. 3 The execution example 2 for GaloisSoftTest.

(1) $OA(32, (1, 8), (8, 4), 2)$

テスト回数が 32 で、8 水準の因子が 1 個、4 水準の因子が 8 個の強さ 2 の直交表であり、 $PG(4,2)$ 上の点をすべて使っている最良の直交表である。

(2) $OA(128, (1, 16), (12, 8), (8, 4), (4, 2), 2)$

テスト回数 128, 16 水準の因子が 1 個、8 水準が 12 個、4 水準が 8 個、2 水準が 4 個で強さが 2 の直交表であり、 $PG(6,2)$ 上の点をすべて使っている最良の直交表である。

なおこのツールによって生成できる直交表のサイズはソフトウェアテストで十分な大きさである 2048 まで生成できる。ただし原始既約多項式を入れればいくらかでも大きくできる。

5. 直交表とオールペア法の比較

ここでは、我々が開発した直交表のツールである GaloisSoftTest の良さを示すために、代表的なオールペア法の生成ソフトである PICT に対して、効率性の尺度であるテスト回数と品質を示す尺度である網羅率について比較検討する。網羅率とは、いくつかの因子間のテストすべきすべての水準組合せに対する、実際にテストする水準組合せの比率である。したがって、強さ 2 は、2 因子間の網羅率 100%を保証することを意味する。一方で、強さ 2 の直交表では、2 因子間の網羅率 100%を保証しているが、実は 3 因子間の網羅率も高い値で保証しているのである [12]。さらにテスト回数が大きく異なる場合は網羅率だけでは良さの比較が困難となるので、新たにテスト回数あたりの網羅率である網羅効率を導入することによりソフトウェアテストにおけるテスト回数と品質の関係を述べることにする。

表 9 GaloisSoftTest と PICT のテスト回数の比較

Table 9 The comparison of GaloisSoftTest and PICT in test size.

因子数	水準	強さ	PICT	Galois	比較
3	2	2	4	4	△
7	2	2	7	8	
4	3	2	13	9	○
13	3	2	20	27	
5	4	2	22	16	○
21	4	2	39	64	
6	8	2	91	64	○
4	2	3	8	8	△
8	2	3	16	16	△
16	2	3	23	32	
4	3	3	34	27	○
10	3	3	65	81	
6	4	3	111	64	○
17	4	3	207	256	
6	8	3	846	512	○
5	2	4	23	16	○
6	2	4	26	32	
5	3	4	100	81	○
11	3	4	249	243	○
14	3	4	296	729	
5	4	4	333	256	○

混合形の比較

水準と因子数(強さ2)	PICT	Galois	比較
8水準1個+2水準8個	16	16	△
4水準4個+2水準3個	19	16	○
16水準1個+2水準16個	32	32	△
8水準1個+4水準8個	38	32	○
4水準9個+2水準4個	29	32	
16水準1個+4水準16個	74	64	○
8水準8個+4水準1個+2水準4個	101	64	○

5.1 テスト回数の比較

一般的にオールペア法が直交表と比較してテスト回数が少なくなるが、オールペア法ツールではテストの最小回数を保証していない実装になっていることが多く、実際、PICTはGaloisSoftTestよりも多くのテストが必要になることがあった[12].

表 9 に GaloisSoftTest と PICT のテスト回数を示す。なお表 9 で取り上げた因子数や水準数は、GaloisSoftTest が有利になる条件である「一定のテスト回数で最大の因子数と水準数になる組合せ」を中心に取り上げている。表中の○は PICT と比較して GaloisSoftTest のテスト回数が少ないことを表し、△は回数であることを示している。

オールペア法では回数個の出現を保証しなくてよいので本来は直交表より回数が小さくなるが、PICT の実装では全数検査をしていないため、全数検査をしている GaloisSoftTest よりも回数が大きくなる場合があることが表 9 から分かる。このように GaloisSoftTest が同一個数の出現というテストに有用な性質を持っていて、かつ回数が小さくなるという場合があり、GaloisSoftTest に価値がある。

5.2 網羅率の比較

別の比較として 100%出現を保証する組合せの個数 t (強さ) に対して、強さ $t+1$ のときの網羅率の比較がある。たとえば強さ 2 の直交表とオールペア法の表を強さ 3 とみたときの出現の網羅率を比較する。ここで PictMaster は強

表 10 網羅率の比較

Table 10 The comparison in coverage rate.

因子	水準	強さ	PictMaster		Galois		比較	
			目標	網羅率	回数	網羅率		
7	2	2	-	73.9	8	90.0	△	
15	2	2	-	10	83.2	16	96.2	-
15	2	2	95	14	95.1	16	96.2	△
4	3	2	-	13	47.2	9	33.3	-
7	3	2	-	16	53.7	27	88.6	-
7	3	2	80	28	80.6	27	88.6	◎
13	3	2	-	20	60.1	27	87.9	-
13	3	2	87	36	86.7	27	87.9	◎
13	4	2	-	33	45.3	64	87.7	-
13	4	2	87	92	87.7	64	87.7	◎
15	5	2	-	54	38.3	125	88.9	-
15	5	2	88	191	88.6	125	88.9	◎

表 11 テスト回数あたりの網羅率 (網羅効率)

Table 11 The comparison of PictMaster and Galois in coverage rate per test size (coverage efficient rate).

条件			PictMaster			Galois			比較
因子数	水準数	強さ	回数	網羅率	網羅効率	回数	網羅率	網羅効率	
15	2	2	10	83.2	8.32	16	96.2	6.01	
4	3	2	13	47.2	3.63	9	33.3	3.70	
7	3	2	16	53.7	3.36	27	88.6	3.28	
13	3	2	20	60.1	3.01	27	87.9	3.26	
5	4	2	21	32.5	1.55	16	25.0	1.56	
13	4	2	33	45.3	1.37	64	87.7	1.37	
6	5	2	35	27.4	0.78	25	20.0	0.80	
15	5	2	54	38.3	0.71	125	88.9	0.71	

さ t のときに強さ $t+1$ の目標網羅率を設定することができるので、GaloisSoftTest の網羅率に合わせて回数の比較もした。表 10 に PictMaster と GaloisSoftTest の網羅率の比較を示す。直交表は強さ t で同一個数の出現の性質があるため、同一個数の出現を保証しないオールペア法と比較して、強さ $t+1$ の出現がより均等になり、テスト回数が少ないときでも網羅率が高い場合があることが表 10 から分かる。

比較の◎は回数も網羅率も両方とも GaloisSoftTest が優れているものを示し、△は回数の差が 2 個以内で網羅率が高いものを示す。この比較から GaloisSoftTest が網羅率で良くなっている場合が存在することが分かる。なお表 10 も表 9 と同様に GaloisSoftTest が有利な点になっている。

5.3 テスト回数あたりの網羅率 (網羅効率)

テスト回数が大きく異なっているときでも網羅率の比較をするためにテスト回数あたりの網羅率 (網羅効率) で比較したものを表 11 に示す。この網羅効率がソフトウェアテストの良さを表す 1 つの尺度になりうる。両者の網羅効率に大きな差はないが、表 11 の比較に示すように GaloisSoftTest が良くなる条件も存在している。なお表 11 も表 9 と同様に GaloisSoftTest が有利な点になっている。

6. 直交表とオールペア法の選択基準と並行運用

組合せテストをするときに、直交表かオールペア法を決め、そのときの組合せの強さを決める必要がある。

6.1 テスト手法の選択

直交表は同一個数の出現という組合せが均一なテストが実施できる。この均一なテストはソフトウェアテストにとってテスト項目に偏りがなく均等にテストができるという有利な点がある。このためテスト回数に大きな差がない場合は直交表を選択する。またテスト回数に大きな差があるときは対象のソフトウェア種別に応じて均一テストが必要かどうかで選択基準を決める必要がある。

6.2 テストの強さの決定

次に組合せテストの強さを決める必要がある。ここで Kuhn ら [13] によって示されたテストの組合せ数と障害の発見率をソフトウェア種別にグラフ化したものを図 4 に示す。図 4 から分かるように組合せ 1 個では発見率は低く、システムによっては 2 個の組合せ（強さ 2）で飽和しているものもあれば、組合せ 2 個と 3 個で差が大きいものもある。図 4 や他の結果からソフトウェア種別に応じた組合せテストの強さの選択基準を決める必要がある。

6.3 テストの強さと手法の選択と並行運用

我々は直交表生成ツール **GaloisSoftTest** とオールペア法ツール **PICT/PictMaster** を並行運用している。並行運用のために、(1) 2 個のツールの入力と出力の形式を揃えることで、1 つの入力データで両方のツールを同時に適用し、(2) 2 つのツールの出力結果を見て優れた組合せテストであった方を採用することにした。(1) によって入出力の形式が同じであるので、テスト実施者は 2 つのツールがあたかも 1 個のツールであるかのように扱えるようになった。(2) の 2 つのツールの出力結果の比較であるが、出力結果のテストケース数にそれほど差がなければ、均一出現という優れた性質がある直交表生成ツール **GaloisSoftTest** の結果を採用するようにしている。また網羅率や網羅効率も考慮して選択するようにしている。

また手法と強さの決定は機械的に決めるだけでは不十分で、図 5 のように分析と評価を実施し、その経験を蓄積して選択していく PDCA サイクルを回すべきである。図 4 からソフトウェア種別に適した強さを選択し、次に上記にあるように **PICT** と **GaloisSoftTest** を両方適用し、この結果とソフトウェア種別により手法を決定する。このときに因子と水準を調整して、それをツールにかけることを繰り返して、因子と水準を決定する。このようにして得られたテストケースを従来のテスト基準によるテストケースや

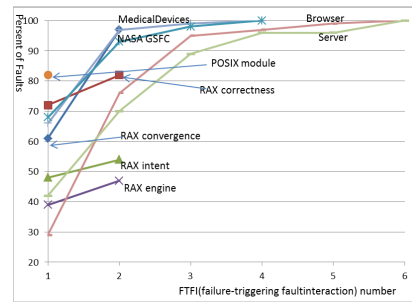


図 4 テストの組合せ数と障害の発見率. 文献 [13] よりデータを引用してグラフ化

Fig. 4 Faults rate per FTFI (Failure-triggering fault intersection) number.

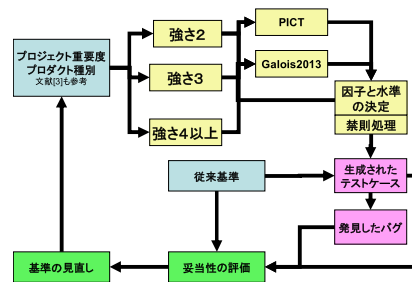


図 5 強さと手法の選択の PDCA サイクル

Fig. 5 PDCA cycle for selection of strength and methods.

過去の経験と比較して、妥当性のあるテストケースであれば、テストを実施する。妥当でない場合やテスト後に多くのバグが見つかった場合は基準を見直す。このサイクルを繰り返すことにより選定基準を良くしていく。

7. おわりに

我々は、ソフトウェアテスト向け直交表生成ソフト **GaloisSoftTest** を開発した。**GaloisSoftTest** は多因子・多水準で、強さも 2 だけでなく、3 と 4 の直交表の生成ができ、さらに、混合形にも対応できる万能型直交表生成ソフトである。**GaloisSoftTest** は、有限射影幾何の理論をベースに構成されており、因子数と水準数、強さが与えられたとき、計算アルゴリズムによって自動的に最適な（テスト回数最小の）直交表が生成できる。

次に、組合せテストの代表的なテスト手法である直交表とオールペア法について、今回我々が開発した直交表生成ツールである **GaloisSoftTest** とオールペア法の生成ツールである **PICT** に対して、そのコストパフォーマンスを左右するテスト回数と網羅率について求め、比較分析した。実際にどちらのテスト手法を選択して運用するのかについては難しい問題もあるが、選択の際の 1 つの指針となるものと考えられる。最後に、この 2 つのツールを並行運用して、効率的でしかも品質レベルを保証する組合せテストを運用する方針や施策などについても記述した。

参考文献

- [1] 田口玄一：実験計画法，第3版，丸善，上（1976），下（1977）.
- [2] 高橋磐郎：組合せ数学を用いた実験計画の自動計画アルゴリズム開発計画について，早稲田大学システム科学研究会報，Vol.29, pp.20–26 (1975).
- [3] 高橋磐郎：組合せ理論とその応用，岩波書店（1979）.
- [4] Fuji-Hara, R.: On Automatical Construction for Orthogonal Designs of Experiments, *Rep. Stat. Appl. Res. JUSE*, Vol.25, No.1, pp.13–25 (1978).
- [5] 須田健二，宮崎晴夫：直交配列を用いた実験計画における要因割りつけのコンピュータ・アルゴリズムについて，日本経営工学会誌，Vol.37, No.6, pp.345–352 (1987).
- [6] 高橋磐郎，藤原 良，杉本英二，須田健二，神保雅一：直交実験の自動計画と Maximal 3,4-linearly independent set, 京都大学数理解析研究所講究録 285, pp.13–23 (1976).
- [7] 須田健二：パソコンによる直交表の自動構成とソフトウェアテストへの応用—多因子・多水準，強さ 2・3・4 対応の直交表生成ソフト Galois の応用，ソフトウェアテストシンポジウム 2007, pp.91–97 (2007).
- [8] 須田健二：直交表と万能型直交表生成ソフト Galois の活用—強さ 2, 3, 4 の直交表の生成事例，ソフトウェアテストシンポジウム 2009, pp.54–56 (2009).
- [9] 五味 弘，辻村 浩，小池宏道，須田健二：直交表とオールペア法の並行運用によるソフトウェアテスト—手法と強さ，因子，水準の選択ガイドライン，ソフトウェアテストシンポジウム 2014, pp.89–96 (2014).
- [10] Czerwonka, J.: Pairwise Testing in the Real World: Practical Extensions to Test-Case Scenarios (Feb. 2008), available from <http://msdn.microsoft.com/en-us/library/cc150619.aspx>.
- [11] PictMaster, available from <http://sourceforge.jp/projects/pictmaster/>.
- [12] 須田健二，五味 弘：直交表とオールペア法のテスト回数と網羅率について，情報処理学会第 76 回全国大会講演論文集，Vol.2014, No.1, 6A-4, pp.239–240 (2014).
- [13] Kuhn, D.R., Wallace, D.R. and Gallo Jr., A.M.: Software Fault Interactions and Implications for Software Testing, *IEEE Trans. Softw. Eng.*, Vol.30, No.6, pp.418–421 (2004).



須田 健二 （正会員）

元群馬工業高等専門学校教授。群馬高専第 1 期卒業，早稲田大学留学。放送大学や群馬大学の非常勤講師を歴任。著書に『電子回路』（コロナ社）等がある。



五味 弘 （正会員）

沖電気工業株式会社エバンジェリスト。三重大学大学院博士後期課程修了，博士（工学）。著書に『プログラミング言語論』等。関数型言語やソフトウェアテスト，組込み系開発に興味。本会シニア会員。