



# 公開質問会

## 直交表 VS All-pair

---

秋山 浩一（富士ゼロックス）  
土屋 達弘（大阪大学）  
町田 仁司（松下電器）

# 手法の比較表

	直交表	HAYST法 (秋山発表)	All-pair (土屋先生発表)
一番の違い	直交表にテストを割り付けて実験後、一因子実験を実施する	2水準系直交表(L4 ~ L256)を変形し、多水準が入る列を作って使用。禁則処理を持つ	都度、評価対象に合わせて最小のテスト項目数になるように生成
使用する表	L18とL36がメイン	2水準系直交表 L4 ~ L256を変形しながら使用	都度、評価対象に合わせて生成
使用できるパラメータ(因子)数	乱数で行を乱しながら表を横に連結すること(乱塊法)で可能だが、網羅率は落ちる	300程度迄(L256使用時)	特に制限なし
ひとつのパラメータ内で取り扱える値(水準)数	基本的に最大3	最大16。それ以上も網羅率は落ちるが可能	特に制限なし
禁則処理	なし	あり	あり
2因子間の網羅率	水準をピックアップしているので多水準因子が多いと網羅率が落ちる	多水準と禁則が無ければ100%だが通常は70% ~ 90%	100%(多水準と禁則があるとテスト項目数は増える)
3因子間の網羅率	同上	L8で50% L16で強度3の列が5個 L64で強度3の列が8個	因子数, 水準数等に依存
メリット	36回でテストが終わる	テスト項目数をコントロールしやすい。簡単な入力で禁則処理が可能。テスト対象の仕様変更に対応可能	最小のテスト項目で生成
デメリット	多因子・多水準になると網羅率が落ちるのでバグ出しには不適(最終確認には良い)	100%の網羅率を目指すとテスト項目が若干多くなる	テスト対象の仕様変更時には作り直し。テスト項目数のコントロールは難しい

# < 9つの質問 >

Q 1 ~ Q 3 : 直交表

Q 4 ~ Q 7 : All-pair

Q 8 ~ Q 9 : 実運用

Q 1 テスト数の増減法

Q 2 連続値を取る因子の  
水準の選び方

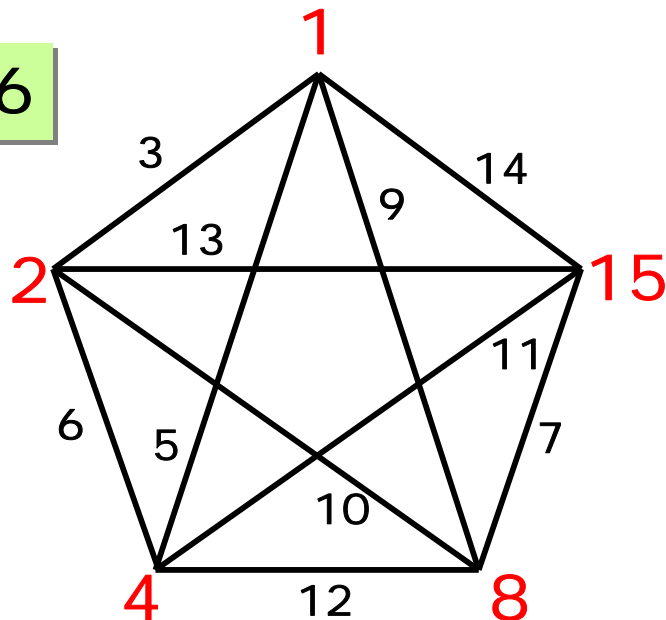
### < テスト数を増やす方法 1 >

因子Aの水準が $a_1, a_2$ の時に  
 $a_1, a_1, a_1, a_2$ と4水準として割り  
り付ける(ダミー水準の付加)  
     $a_1$ が $a_2$ の3倍テストされる  
(この時、テスト数は増える)

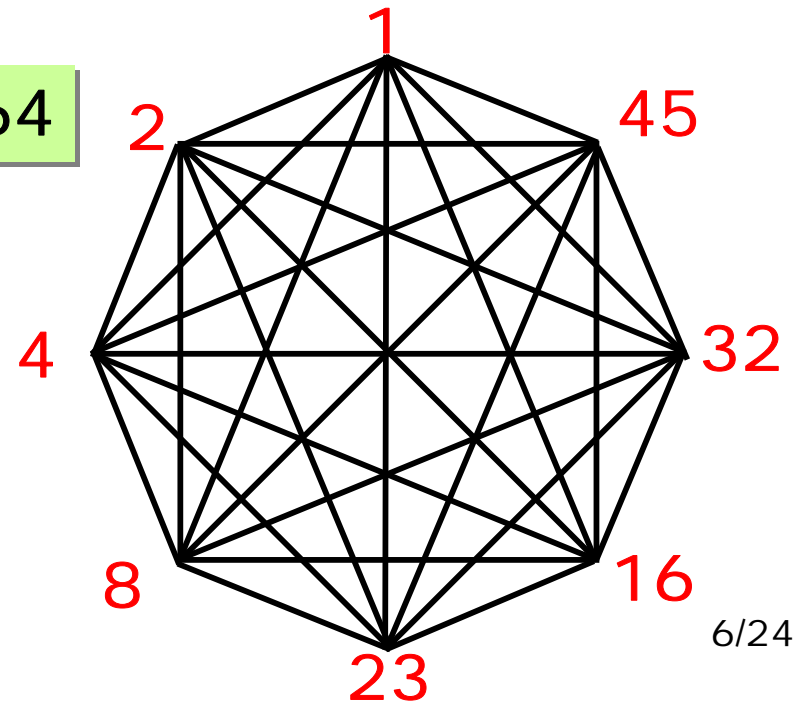
## < テスト数を増やす方法 2 >

強度 3 の列のみを利用して  
3 因子間網羅率を 100% とする

L16



L64



### < テスト数を減らす方法 >

因子Aの値が $a_1, a_2, a_3, a_4, a_5$ の時に $a_4$ と $a_5$ を抽象化して割り付ける  $a_4, a_5$ は $a_1, a_2, a_3$ の半分の出現率になる  
(この時、テスト数は減る)

## A 2 連続値を取る因子の水準の選び方

---

### < 同値分割/境界値分析 >

正常境界値を水準とする

- 倍率なら 25 % と 400 %

異常境界値は水準に入れない

- 24 % と 401 % は単機能  
テストで検査する



## A 2 連続値を取る因子の水準の選び方

---

### < 注意 >

部数 ( 1 部 ~ 9 9 9 部 ) の場合

- 9 9 9 部は選ばない！！

補正 ( 1 . 0 ~ 2 . 6 ) の場合

- ユーザが使う代表値を選ぶ

1 . 0 , 1 . 4 , 1 . 8 , 2 . 2 , 2 . 6 を選ぶ

Q 3 多因子・多水準で  
網羅率 = 100%を狙う  
とテスト項目が激増！？

## A 3 大規模で網羅率100% テスト激増！？

---

### < テスト数が激増する要因 >

因子A ( 40水準)、因子B ( 30水準)の時、網羅率100%を狙うと、

$$40 \times 30 = 1200 \text{項目}$$

のテスト量を必要とする

## A 3 大規模で網羅率100% テスト激増！？

---

### < 対策：水準の抽象化 >

用紙サイズなら A 系 ( A 4 , A 5 )、B 系 ( B 4 , B 5 ) と抽象化  
論理レベルの網羅性を 1 0 0 %  
とする指針をとる。  
全体としては 8 0 % を目安

Q 4 All-pair法の適用事例

Q 5 All-pair法のツール

紹介希望！！

## A 4 All-pair法の適用事例

Bellcore (現Telcordia)製品への適用

事例報告\*

### Integrated Services Control Point (ISCP)

プログラマブルな電話サービスを実現するシステム

1. 算術操作, テーブル操作  
3水準12因子 + 4水準5因子, 5水準12因子 + 2水準30因子
2. メッセージのパーシング
3. ルールベースシステム
4. GUI

	#テスト	#バグ
1	1,601	43
2	4,500	27
3	13	4
4	159	6

\*S.R. Dalal et al., "Model-Based Testing in Practice," ICSE'99. 14/24

<http://www.pairwise.org/>

## A 5 All-pair法のツール

1. <a href="#">CATS (Constrained Array Test System)</a> *)	[Sherwood] Bell Labs.	
2. <a href="#">OATS (Orthogonal Array Test System)</a> *)	[Phadke] AT&T	
3. <a href="#">AETG</a> 	Telecordia	Web-based, commercial
4. <a href="#">IPO (PairTest)</a> *)	[Tai/Lei]	
5. <a href="#">TConfig</a> 	[Williams]	Java-applet
6. <a href="#">TCG (Test Case Generator)</a> *)	NASA	
7. <a href="#">AllPairs</a> 	Satisfice	Perl script, free, GPL
8. <a href="#">Pro-Test</a>	SigmaZone	GUI, commercial
9. <a href="#">CTS (Combinatorial Test Services)</a>	IBM	Free for non-commercial use
10. <a href="#">Jenny</a> 	[Jenkins]	Command-line, free, public-domain
11. <a href="#">ReduceArray2</a>	U.S. Air Force	Spreadsheet-based, free
12. <a href="#">TestCover</a>	Test Cover.com	Web-based, commercial
13. <a href="#">DDA</a> *)	[Colburn/Cohen/Turban]	
14. <a href="#">Test Vector Generator</a>		GUI, free
15. <a href="#">OA1</a>	k sharp technology	
16. <a href="#">CTE-XL</a>	Daimler Chrysler	GUI, free
17. <a href="#">AllPairs</a>	[McDowell]	Command-line, free
18. <a href="#">Intelligent Test Case Handler (replaces CTS)</a>	IBM	Free for non-commercial use

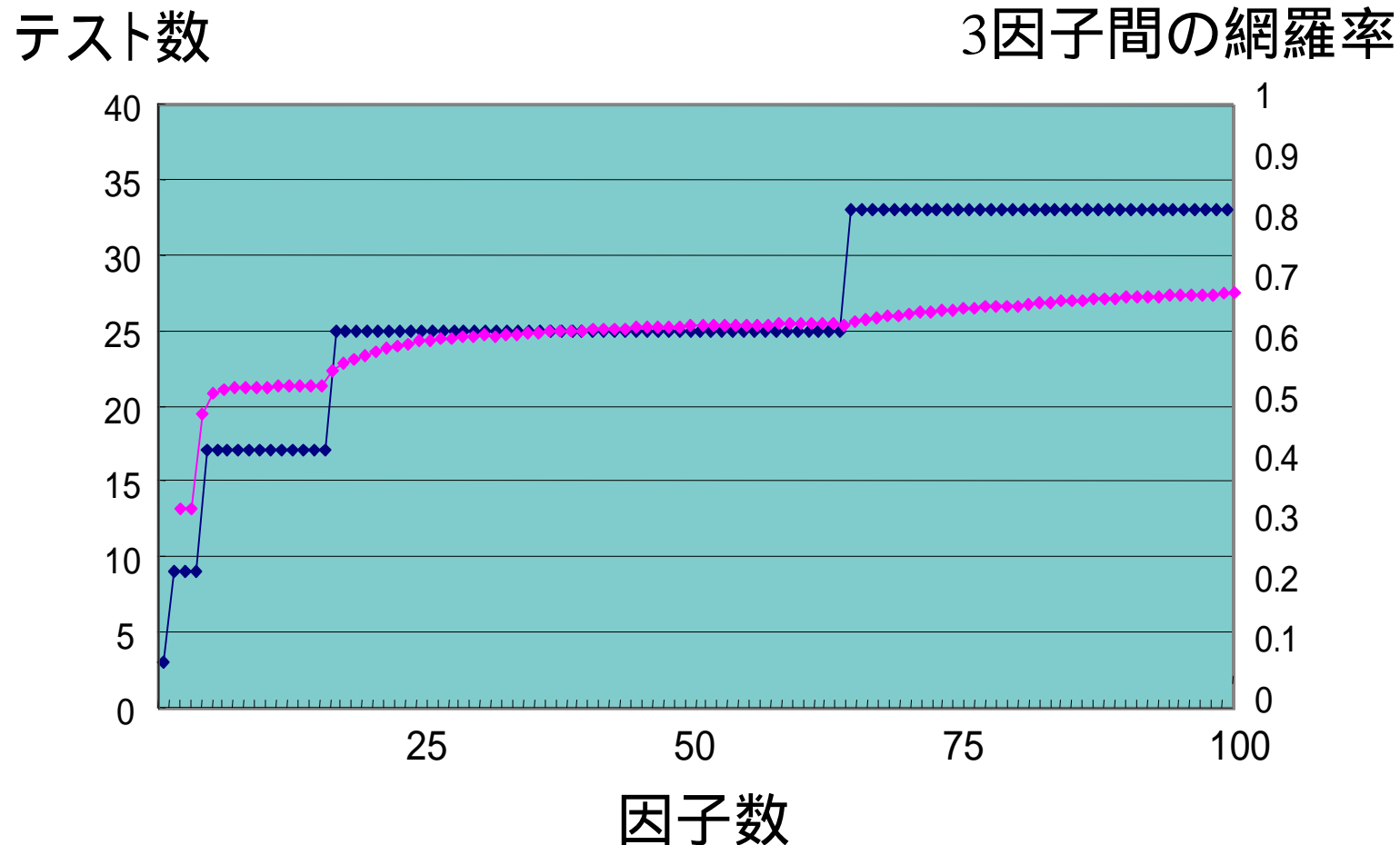
<http://www.swtest.jp/>

# Q 6 All-pair法における 3 因子間網羅率



## A 6 All-pair法における 3 因子間網羅率

### 3因子間の網羅率(3水準の場合)



# Q7 仕様変更が多いプロジェクトへAll-pair法を適用する場合の注意点

## A 7 仕変が多い場合のAll-pair法の適用注意点

---

< 問題点 >

仕様変更時      テスト作り直し

< 対処法 >

予備の因子を予め導入しておく

必要なテスト数はほとんど増えない

Q 8 適用条件は？

「規模」、「部分適用」他

A 8 適用条件は？「規模」「部分適用」他

---

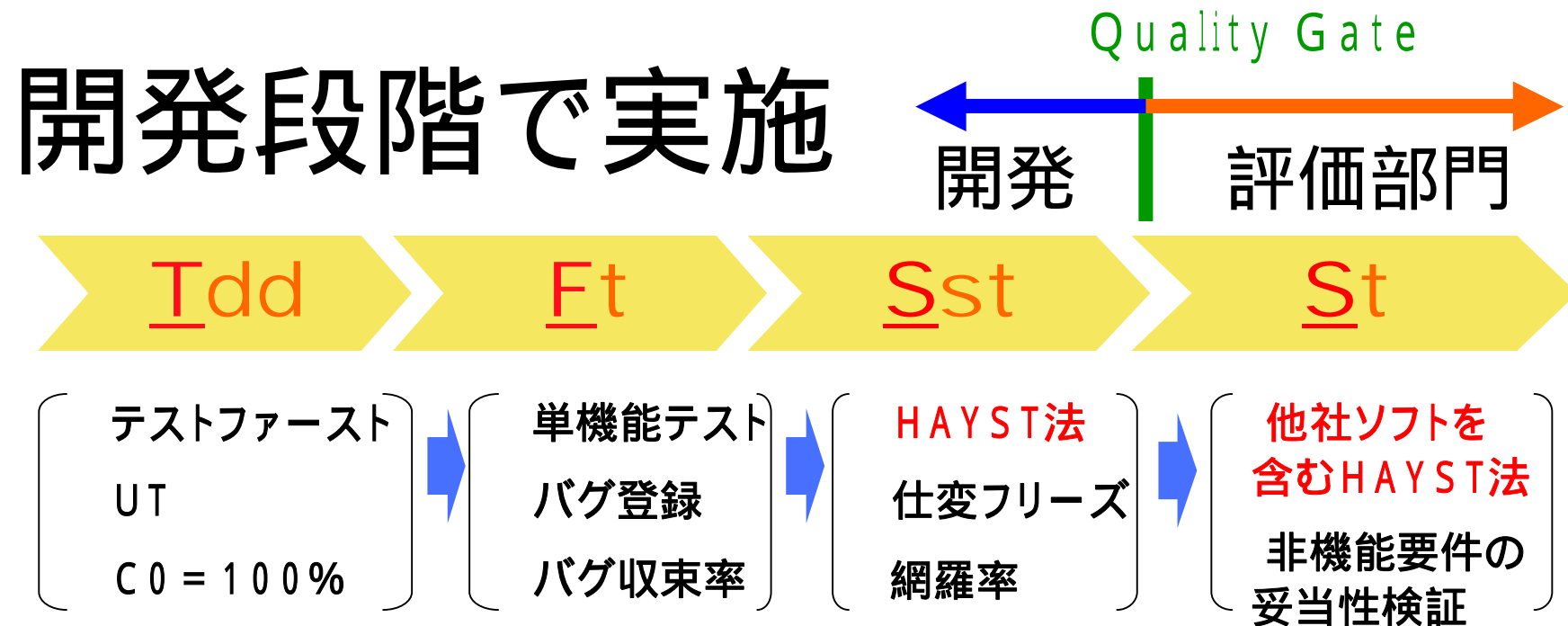
全プロジェクトが適用対象

規模は関係なし

ローレベルの関数を作っている  
グループは上位層で直交表を  
使うため対象外

Q 9 適用工程は？  
「最終テスト」？  
「開発段階」？

## A 9 適用工程は？「最終テスト or 開発段階」



# 品質の経営

全社活動対応(部分最適から全体最適へ)

# 全体のまとめ