International Journal of GEOMATE, Jan., 2017, Vol. 12, Issue 29, pp. 66-72
Geotec., Const. Mat. & Env., ISSN: 2186-2982(P), 2186-2990(O), Japan

AN ALGEBRAIC AND COMBINATORIAL APPROACH TO THE CONSTRUCTION OF EXPERIMENTAL DESIGNS

Julio Romero[1] and Scott H. Murray[2]

[1]Department of Information Sciences and Engineering, University of Canberra, Australia;
2Department of Mathematics and Statistics, University of Canberra, Australia.

ABSTRACT: Experimental design is a well-known and broadly applied area of statistics. The expansion of this field to the areas of industrial processes and engineered systems has meant interest in an optimal set of experimental tests. This is achieved through the use of combinatorial and algebraic approaches. As such, the present study states the theoretical basis to construct and enumerate experimental designs using non-isomorphic mathematical structures in the form of matrix arrangements called orthogonal arrays (OAs). These entities are characterized by their number of rows, columns, entries (symbols), and strength. Thus, each different column could represent some measurable feature of interest (temperature, pressure, speed). The runs, expressed through OA rows, define the number of different combinations of a particular design. Similarly, the symbols allocated in OAs' entries could be the distinct levels of the phenomenon under study. During the OA construction process, we used group theory to deal with permutation groups, and combinatorics to create the actual OAs following a particular design. The enumeration process involved the use of algebraic-based algorithms to list all possible combinations of arrays according to their isomorphic equivalent. To test isomorphism, we used graph theory to convert the arrays into their corresponding canonical graph.

The outcomes for this study are, firstly, a powerful computational technique to construct OAs from 8 to 80 runs; and secondly, additions in the published list of orbit sizes and number of non-isomorphic arrays given in [1] for 64, 72, and 80 runs.

Keywords: Orthogonal arrays, Combinatorics, Experimental design, Engineering parameter design

1. INTRODUCTION

The engineering or scientific method is the approach of solving problems through the efficient application of scientific principles based upon a well-structured theoretical knowledge [2]. In applying the aforementioned approach, engineers undertake experiments or tests as an intrinsic and natural part of their jobs. As such, sound statistical-based experimental designs are extraordinarily useful within the engineering profession aiming to improve process and systems characterized by suitable engineering specifications. These specifications are described as several "controllable" variables associated with the overall system/process performance. Therefore, knowing the main factors and their interactions, the engineer is able to analyze, for instance, process yield, process variability, development time, and operational costs [3], [4].

Latest implications in using experimental tests are related with *engineering parameter design*. Within this discipline, it is possible not only to develop new products/processes, but also to enhance the existing ones. To cite some examples in parameter design, we have the evaluation and comparison of basic design configurations; the appraisal of different alloys in strength of materials; the selection of design parameters to make the product / process to work within a specified tolerance range under a wide variety of field conditions (we say the design is being robust); and the determination of the key factor combinations affecting a particular product performance.

Besides their flexibility of being easily implemented in engineering design [2], our arrays are mathematically made so we can perform their combinatorial "enumeration". This last idea is the one presented in this paper. Our interests in following a purely mathematical point of view in dealing with experimental designs are, firstly, the development of algorithms which actually calculate the objects we are defining theoretically; secondly, to answer concrete questions in group theory to explore more of it; and thirdly, to deal with complexity theory such as to able to answer the problem whether two graphs, given by their adjacency matrix, are in fact, isomorphic [5].

We initially present the main concepts of group theory which underlie the algorithms used to construct the arrays. We move further with the tools provided by the theory of graphs, to map an array into its equivalent canonical graph to be able to indirectly assess isomorphism between the arrays. We then present a pseudo code to implement a backtrack search algorithm to look through the different *orbits* which conform the *orbit space* of a particular array of design type *t*.

2. COMBINATORIAL SETTING

Orthogonal arrays (OAs) are related to combinatorics, finite fields, geometry, and error correcting codes. Fig. 1 shows an example of an OA of strength two (transposed).

$$F^T = \begin{vmatrix} 0 & 0 & 1 & 1 & 2 & 2 & 0 & 0 & 1 & 1 & 2 & 2 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \end{vmatrix}$$

Fig. 1 Strength two OA (transposed)

Each of the four possible rows does appear in the array, and they appear the same number of times (picked them in groups of $t = 2$ columns). This is the fundamental property that defines an OA [6]. In the previous example, only three *symbols* appeared, 0, 1, and 2. Thus, we say the array has 3 *levels*. Note that each level may represent a particular combination of physical properties: fuel with additive or not, 20ºC or 80ºC, machine running or stopped, and so forth. Following the same example, we have 3 columns each representing a variable under study (additive, temperature, machine's condition). In a similar way, our array F has 12 rows or *runs*, each representing a particular combination of the different variable levels. Mathematically, we represent an OA of strength t, s levels, N runs, and a variables as: $OA(N, S_i^{a_j}, t)$. For our particular array F, we have $OA(12, 3^1 2^2, 2)$ or equivalently, a design type $U(3^1, 2^2)$ [6], [7].

A strength-*t* OA assures that all possible combinations of the levels in a variable of up to *t* occur together equally often [6]. Thus, the strength of the array defines the *effects of each individual variable* (factor) that can be taken in consideration when assessing the design [6], [8]. Also, according to the strength *t* required, we will be able to calculate some *interactions* between factors as well. Such experiments are called *fractional factorial designs* [7]. Our particular interest is in OAs of strength 3, which can model all main factors in a particular phenomenon under study and up to two interactions between the factors.

3. MATHEMATICAL BACKGROUND

This section shows how we performed the construction and enumeration of strength three orthogonal arrays. We initially explain the mathematical approach used for the permutation computational implementation.

We applied group theory to define the full group of factor transformations and the corresponding automorphism group. Furthermore, we used graph theory to represent the orthogonal array as a colored graph. This approach allowed us to define a canonical representative of an orbit of OAs using the canonical graph [9]. This leads to improve the computational time and memory usage for the calculations. We also implemented the backtrack search and the lexicographical least run algorithms to find the arrays and as a decision criteria to drop a leaf [1], [10].

3.1 Algebraic Setting

Consider F_{ND} as the set of all elements of the form:

$$F_{ND} = \{F: F \; is \; a \; N \times d \; matrix, F_{ij} \in S_j\} \quad (1)$$

called the set of all *fractional factorial designs* (*FFDs*). We define the *groups* G_ρ, G_γ, and G_σ, acting on F_{ND} as $G_\rho = Sym(N)$, the group of all permutations acting on rows; $G_\gamma = Sym(D)$, the group of permutations acting on columns; and $G_\sigma = Sym(S)$, the permutation group acting on symbols. Thus, $\rho \in G_\rho$ defines an action:

$$(F_{i,j}) \to F_{ND}^{\rho} = (F_{\rho i,j}) \quad (2)$$

similarly, $\gamma \in G_\gamma$ defines an action:

$$\prod_k Sym(J_k) \subseteq Sym(D) \to F_{ND}^{\gamma} = (F_{i,\gamma j}) \quad (3)$$

and $\sigma \in G_\sigma$ defines an action:

$$\sigma = (\sigma_1, \dots, \sigma) \in \prod_k Sym(J_k) \subseteq Sym(S) \to F_{ND}^{\sigma} = \left(F_{i,j}^{\sigma_j}\right) \quad (4)$$

Definition 1. Consider the set F_{ND} of $N \times d$ arrays, and G the group of permutations. We define the *action* of the group G on the set F_{ND} as a map $\phi: G \times F_{ND} \to F_{ND}$ such that:

- $F_e = F, \forall F \in F_{ND}$; e, identity element.
- $(F)(g_1 g_2) = (F g_1) g_2 \; \forall \; g_1, g_2 \in G$.

Corollary 2. Let the map $\emptyset$ define a group homomorphism such that $\emptyset: G_1 \to G_2$ where G_1 and G_2 are both permutation groups. Then $\emptyset$ is a one-to-one map if and only if $ker(\emptyset) = e$

Proof. Assume the map $\emptyset$ is one-to-one. It holds that $\emptyset(e) = e_2$, the identity element of G_2. Thus, e is the only mapped element into e_2 by $\emptyset$, so $ker(\emptyset) = e$. Assume now that $ker(\emptyset) = e$. For all $g \in G$ the element mapped into $\emptyset(g)$ are the elements of the right coset $\{eg\} = \{g\}$ showing that

$\emptyset$ is one-to-one.

Definition 3. We have the following mappings:

$$\emptyset_\rho: \mathrm{Sym}(\mathrm{N}) \to \mathrm{Sym}(\mathrm{F_{ND}}) \quad (5)$$

$$\emptyset_\gamma: \prod_k Sym(J_k) \to Sym(F_{ND}) \quad (6)$$

$$\emptyset_\sigma: \prod_k \mathrm{Sym}(\mathrm{S_j}) \to \mathrm{Sym}(\mathrm{F_{ND}}) \quad (7)$$

It follows from definitions 1 and 3,

$$\mathrm{Im}\emptyset_\rho, \mathrm{Im}\emptyset_\gamma, \mathrm{Im}\emptyset_\sigma \leq \mathrm{Sym}(\mathrm{F_{ND}}) \quad (8)$$

$$\mathrm{G} = < Im\emptyset_\rho, \mathrm{Im}\emptyset_\gamma, \mathrm{Im}\emptyset_\sigma > \quad (9)$$

$$\mathrm{G} = \mathrm{G}_\rho \times \mathrm{G}_\sigma \rtimes \mathrm{G}_\gamma \quad (10)$$

where $\mathrm{G}_\sigma \rtimes \mathrm{G}_\gamma = \prod_{k=1}^{m} \mathrm{Sym}(\mathrm{S_{j,k}}) \wr \mathrm{Sym}(\mathrm{J_k})$ or equivalently,

$$\mathrm{G} = \emptyset_\rho(\rho) \cdot \emptyset_\sigma(\sigma) \cdot \emptyset_\gamma(\gamma) \leq \mathrm{Sym}(\mathrm{F_{ND}}) \quad (11)$$

Definition 4. Let G be a group of permutations acting on a non-empty set $\mathrm{F_{ND}}$; we define the *orbit* of G containing F to the equivalence class given by:

$$\mathrm{Orb_G(F)} = \{\mathrm{Fg} \mid \mathrm{g} \in \mathrm{G}\} \quad (12)$$

we define the *orbit space* as the family of all equivalence classes obtained from G acting on an element F of the set $\mathrm{F_{ND}}$. Let F and T both elements of $\mathrm{F_{ND}}$. We say they are *isomorphic* if there exists a permutation $\mathrm{g} \in \mathrm{G}$ such that $\mathrm{F} = \mathrm{Tg}$. The previous expression can also be written as $\mathrm{F} = \mathrm{T^g}$.

Suppose now that a group G acts on a set $\mathrm{F_{ND}}$; then, for each element $\mathrm{F} \in \mathrm{F_{ND}}$, we redefine the equivalence class (orbit) containing F as:

$$\mathrm{O_G(F)} = \{\mathrm{F^g} \mid \mathrm{g} \in \mathrm{G}\} \quad (14)$$

Lemma 5. Suppose that a group G acts on a set $\mathrm{F_{ND}}$; then, for each $\mathrm{F} \in \mathrm{F_{ND}}$,

$$|\mathrm{G}| = |\mathrm{stab(F)}||\mathrm{O_g(F)}| \quad (15)$$

Proof. Suppose that $\mathrm{g_1}$and $\mathrm{g_2}$ are in the right coset of $\mathrm{stab(F)}$ and $\mathrm{g_1} = \pi\mathrm{g_2}$ for some $\pi \in \mathrm{stab(F)}$. Thus, $\mathrm{Fg_1} = (\mathrm{F})(\pi\mathrm{g_2}) = (\mathrm{F}\pi)\mathrm{g_2}$. On the other hand, suppose that $\mathrm{Fg_1} = \mathrm{Fg_2}$, then $\mathrm{F} = \mathrm{Fg_2g_1^{-1}}$ implying that $\mathrm{g_2g_1^{-1}} \in \mathrm{stab(F)}$; therefore $\mathrm{g_1}$ and $\mathrm{g_2}$ belong to the same right coset of $\mathrm{stab(F)}$ if and only if $\mathrm{Fg_1} = \mathrm{Fg_2}$. It follows that there is a bijection between the elements in $\mathrm{Orb_G(F)}$ and the right cosets of $\mathrm{stab(F)}$, thus

$$|\mathrm{G}| = |\mathrm{stab(F)}||\#\text{of right cosets of stab(F)}| \\ = |\mathrm{G}| = |\mathrm{stab(F)}||\mathrm{O_g(F)}|$$

and the result follows.

Theorem 6. (Burnside's Theorem) Suppose the group G acting on a set $\mathrm{F_{ND}}$; then, the number of orbits in $\mathrm{F_{ND}}$; is given by

$$1/|\mathrm{G}| \sum_{\mathrm{F} \in \mathrm{F_{ND}}} |\mathrm{stab(F)}| \quad (16)$$

Proof. Using lemma 5, we have:

$$1/|\mathrm{G}| \sum_{\mathrm{F} \in \mathrm{F_{ND}}} |\mathrm{stab(F)}| = \sum_{\mathrm{F} \in \mathrm{F_{ND}}} 1/|\mathrm{O_g(F)}| \quad (17)$$

suppose there are $\mathrm{O_S}$ orbits in $\mathrm{F_{ND}}$, and F in the orbit $\mathrm{O_g}$; then, $\sum_{\mathrm{F} \in \mathrm{F_{ND}}} 1/|\mathrm{O_g(F)}| = \mathrm{O_S}$.

Definition 7. Let G be the group of all row, column, and symbols permutations, and $\mathrm{F} \in \mathrm{F_{ND}}$. The set of all isomorphisms from F to F is called the *automorphism group* of F and is denoted by Aut(F) The elements of Aut(F) are called *automorphisms of* F.

Using our previous results from Eqs. (15), (16) and (17); the *length* of the G-Orbit of G is the number of distinct objects isomorphic to it. Thus,

$$|\mathrm{Orb_G(F)}| = \frac{|\mathrm{G}|}{|\mathrm{Aut(F)}|} \quad (18)$$

4. ENUMERATION USING GRAPH THEORY

In the previous section, we explained how we mathematically set up the conditions for the construction and enumeration of strength *t* orthogonal arrays. In this section, we formally explain their construction and how we perform the enumeration by mapping the OAs as canonical graphs.

4.1 Mapping an OA to its Equivalent Canonical graph

Let $\mathrm{F_{ND}}$ the set of fractional factorial designs with $\mathrm{F} \in \mathrm{F_{ND}}$. Let $\mathrm{F_{ij}}$ be a particular entry in the array F, where i: $\mathrm{i^{th}}$ row and j: $\mathrm{j^{th}}$ column. We define $\mathrm{S_\rho}$ the set of all N-tuples which represent the different rows of the arrayF. Then, $\mathrm{S_\rho} = \{\rho_1, \dots, \rho_N\}$ $\{\rho_1, \dots, \rho_N\}$ and $\mathrm{F_{ij}} \in \mathrm{S}$; where S is the set containing the different possible symbols for a particular design *U*.

Definition 8. A *colored graph* is a triple $F_G = (V, E, \Gamma)$, where V is the set of the different vertices, v, of the graph; E is the set of edges, and Γ a map from V to a set of colors C.

Definition 9. We define the *neighbor* of a vertex $v_x \in V$ as $\eta(v_x) = \{v_y \in V \mid \{v_x, v_y\} \in E\}$. An isomorphism $F_G \to F'_G = (V, E, \Gamma)$ is a bijection $\eta: V \to V'$ such that: $\{v_x, v_y\} \in E \leftrightarrow \{\eta(v_x), \eta(v_y)\} \in E'$ and $\Gamma(v_x) = \Gamma(v_y) \leftrightarrow \Gamma'^{(\eta(v_x))} = \Gamma'^{(\eta(v_{xy}))} \forall\ v_x, v_x \in V$.

Proposition 10. Let $F \in F_{ND}$. We construct a colored graph $F_G = (V, E, \Gamma)$ for the array F as follows:

a) The set of the vertices V, is made on the elements ρ_i, i: 1,2, ..., N; γ_j, j: 1,2, ..., d and σ_{jx}, j: 1, ... d; $x \in S$; corresponding to the rows, columns, and the distinct levels of the array.
b) The set of the edges, E, made on all of the $E_1 = \{\rho_i, \sigma_{jF_{ij}}\}$ and $E_2 = \{\gamma_j, \sigma_{jF_{ij}}\}$ $\forall$ i: 1, ..., N and j: 1, ..., d.
c) The set of the edges, E, made on all of the vertices ρ_i with color C_ρ; all of the vertices γ with color C_γ; and all of the vertices σ_{jx} with color C_{σ_x}.

Thus, F_G has three partitions: rows, columns, and levels. Mathematically, we express this as $V = C_\rho \cup C_\gamma \cup C_{\sigma x}$. Similarly, we write the set of the edges as $E = E_1 \cup E_2 \subseteq (C_\rho \times C_\gamma) \cup (C_{\sigma x} \times C_\gamma)$. It follows that the cardinalities $|V|$ and $|E|$ are given by

$$|V| = N + \sum_I^d k_i + d \tag{19}$$

$$|E| = dN + \sum_I^d k_i \tag{20}$$

To characterize the graph of an OA, we define the *column-color classes* to the disjoint union of color classes C_ρ, C_γ, and $C_{\sigma x}$. The total number of colors of F_G is $2 + m$ and each row-vertex is adjacent to sv symbol-vertices. Moreover, each symbol-vertex is adjacent to exactly one column-vertex, where $sv = \sum_{i=1}^m |C_i|$

Lemma 11. Let S_{FG} the set of all coloured graphs, and the map $\emptyset$ defined as $\emptyset: F_{ND} \to S_{FG}$ that takes an array $F \in F_{ND}$ to the corresponding colored graph $F_G \in S_{FG}$; thus, $\emptyset$ is an injection.

Proof. The number of vertices of F_G does not depend upon F but only on the design type U and the run size N.

To determine row, symbol, and column vertices, we have the following color partition proposition:

Proposition 12. Let $F \in F_{ND}$ of strength $t \geq 1$ and run-size N. Thus,

a) F_G has a tripartite partition $(C_\rho, C_\gamma, C_{\sigma x})$ with $|C_\rho| = N$, $|C_\gamma| = \sum_{k=1}^m a_k$, and $|C_{\sigma x}| = \sum_{k=1}^m a_k S_k$.
b) Every vertex $v \in V$ has a valence ν denoted by $\nu = V(v)$
c) For the column-vertex set $C_{\sigma x}$, we denote the valency for $c_{\sigma x} \in C_{\sigma x}$ as ν_x, where ν_x is the unique element $\{1, ..., m\}$.
d) Let $s \in S$, the set of different symbols (levels). Then $\exists\ c_{\sigma x} \in C_{\sigma x}$ such that $\{s, c_{\sigma x}\} \in E$, the set of edges for some k in $\{1, ..., m\}$.
e) From d), it follows that the valence of a symbol-vertex is given by

$$\nu(v) = \frac{N}{\nu(c_{\sigma x})} + 1 = \frac{N}{v_x} + 1 \tag{21}$$

f) Let $\rho \in C_\rho$ and $\gamma \in C_\gamma$; thus, there exists a path of length two from ρ to γ through a vertex in V, and this path is unique.

Definition 13. Let an orthogonal array be of design type U and run size N. Consider also a colored graph which satisfies all of the points given in 12 above. Thus, we call the colored graph as being of the type (U; N), which forms a sub-set of F_G.

Lemma 14. Let F_2 and $F_2 \in F_{ND}$ OAs with designs type *U* and *N*. Let also F_{G1} ; $F_{G2} \in S_{FG}$ be their corresponding graphical representations. Thus, F_1 and F_2 are isomorphic arrays if and only if F_{G1} and F_{G2} are isomorphic graphs.

Proof. Suppose F_1 and F_2 are isomorphic arrays $\in F_{ND}$; then $F_1 = F_2^g$; $g \in G$. Because $g \in G$, it follows that g is a product of g_ρ, g_γ, and g_σ permutations; and from c) in proposition (10), we have $F_{G1} = \emptyset(F_1^g) = \emptyset(F_2) = F_{G2}$. Therefore, F_{G1} and F_{G2} are isomorphic graphs. Suppose now that F_{G1} and F_{G2} are isomorphic graphs; thus, $\exists\ g \in G$ such that $F_{G1} = F_{G_2}^g$. Because F_{G1} and F_{G2} satisfy all of the points given in proposition (10), they are tripartite and the permutation g preserves the graph coloring. Therefore, g can be written as the product of g_ρ , g_γ , and g_σ permutations acting on F_1 and the composed map takes F_1 to F_2.

Example. Assume we have an orthogonal array with design type $U(3^1, 2^2)$ and run size $N = 6$ (a fraction of the array shown in Fig. 1); then, the set of vertices is made up on the elements ρ_i, $i \in \{1,\ldots,6\}$; and γ_j, $j \in \{1,\ldots,7\}$; and σ_{jx}, $j \in \{1,2,3\}$, $x \in \{1,2\}$ (see proposition (10)). We assign one color to the set of rows C_ρ; one color to the partition $C_{\gamma 1} \subseteq C_\gamma$ corresponding to the level 3; one color to the second partition $C_{\gamma 2} \subseteq C_\gamma$ corresponding to the two levels of the two columns with the same symbol level $(0,1)$; and one color for the set of symbols $C_{\sigma x}$. We have in total $|V| = 6 + \sum_i^3 r_i + 3 = 16$ vertices, and $|E| = 2 \times 6 + 7$ edges. The colored graph of this array is shown in Fig. 2.

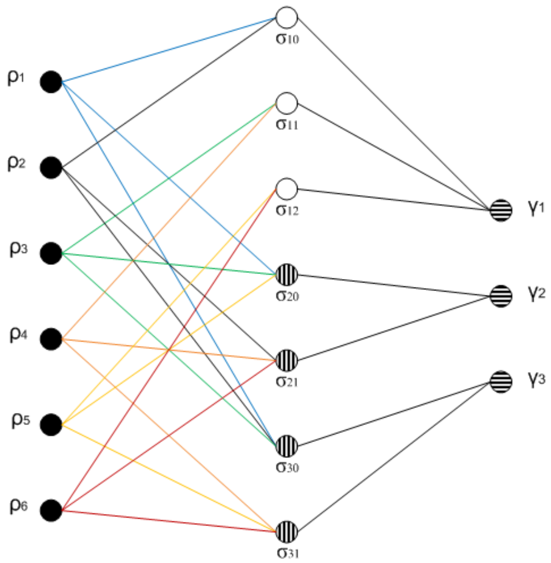

Fig. 2 Canonical graph of the orthogonal array F.

4.2 Algorithm Description

The construction process requires an existing orthogonal array with design type U made using simple combinatorial techniques. We then start adding the different symbols according to the specified design. By adding symbols, we create a new column to the existing array until the new required OA is completed.

During the process of adding symbols, our algorithms check that the conditions of strength t and sequence's lexicographical order are being met; otherwise, the entire OA is discarded. The aforementioned process is carried out using *the backtrack* search algorithm [11].

In addition to the previous criteria, we use our permutation group G with the operations g_ρ, g_γ and g_ρ to prune the tree when we find isomorphic graphs downstream of the search tree (this stage help us to save important computational time and memory-usage resources). Furthermore, the recently made array is then mapped to its equivalent canonical graph in order to be compared against the orbit-representative one. If it turns out that the two graphs are isomorphic, the newest array is counted as part of the orbit-representative array and discarded. However, if the graphs are not isomorphic, the newest array is stored and classified as part of the orbit representatives for the particular design U.

4.3 The Backtrack Search Algorithm

Mathematically, we consider the idea of finding a FFd $\in F_{ND}$ with a design type U. This design specifies, among other parameters, the strength t of the array, which is the main criterion looked in the search tree to drop a leaf.

Definition 15. We define a *partial image* to the set of distinct entries in a row by $S_{eq\rho} = [F_{i1}, \ldots, F_{ir}]$, $0 \le r \le m$. When $r = m$, we call the sequence $S_{eq\rho}$ as *complete*.

According to definition 15, the backtrack search goes through the partial images given by the sequence $S_{eq\rho}$. Within this search, it will use any knowledge of the design to prune the search tree. Note that in constructing a new design we totally order the sequences $S_{eq\rho}$ so that we induce a *lexicographical order* for the set of partial images. This means that only the first point of each k-orbit has to be considered when extending $S_{eq\rho}$.

Figure 3 shows the algorithm we use to traverse a tree created to search orbits and their representatives. The search will go through some of the elements of the permutation group while skipping some leaves according to the previously discussed criteria.

4.4 Isomorphism Classes Enumeration Results

In Table 1, we show some examples of results obtained by Man [1] which were corroborated with our approach. In addition, we introduce new isomorphism classes enumerated using the technique presented in this paper.

The first column of the table represents the run size N for the different designs. The second column corresponds to the actual design type U according to the multiplicity notation for automorphism group orders. The third and fourth columns of the table are the *number* of the automorphism groups and their corresponding *size* respectively. We have indicated with an asterisk the newest designs we have found.

The Backtrack Search Algorithm

Algorithm 1.

Input: Design type, U; run size, N; basis design, F, row position index, Λ; an orbit representative, Δ_F

Output: The automorphism size and number.

$\Lambda := 0; NE := [\,];\ \ extensions := [\,];\ \ \Delta := [\,];$

  function FillingRows(U,N,V,F,Λ,Δ_F)

    if $\Lambda := (N+1)$ **then**;

      if $\exists\{<\alpha,\beta>: \alpha \in F, \beta \in \Delta |\ CheckIso(\alpha,\beta)\}$ **then**

      $append(\Delta_F, F);$

      end if

      return Δ, F;

    end if

    $F[\Lambda] := F[\Lambda] + NE;$

    if $IsStrengtht(F, \Delta, U)$ **then**

      $F_{NEW}, \Delta := FillingRows(U, N, V, F, \Lambda + 1, \Delta_F);$

      $extensions := extensions + F_{NEW};$

    end if

    $F[\Lambda, \#U] := 1;$

    if IsStrenght **then**F, Λ, U

      $F_{NEW}, \Delta := FillingRows(U, N, V, F, \Lambda + 1, \Delta_F);$

      $extensions := extensions + F_{NEW};$

    end if

    $\Lambda := \Lambda + 1;$

    return $extensions$, Δ;

  end function

end

Fig. 3 Backtrack Search Algorithm.

Table 1 Non-isomorphic OAs of strength three

N	Type	#	Size
64	4^5	1	144^1
64	$4^4\,2^1$	3	256^1, 512^1, 1536^1
64	$4^1\,2^5$	12692	
64	$4^1\,2^6$	$\geq 7865^*$	
64	$4^1\,2^7$	$\geq 10661^*$	
64	$4^1\,2^8$	≥ 1189	
72	$6^1\,2^4$	156	256^{36}, 512^{72}, 3072^{32}, 4096^{12}, 110592^4
72	$6^1\,2^5$	64296	
72	$6^1\,2^6$	$\geq 36550^*$	
72	$6^1\,2^7$	$\geq 54834^*$	
80	$4^1\,2^5$	$\geq 18653^*$	
80	$4^1\,2^6$	$\geq 15283^*$	

Note: new arrays indicated with an asterisk *.

5. CONCLUSION

We have presented a mathematical and computational method for constructing and enumerating strength-t orthogonal arrays given a fixed number of experiments. The technique shown provided a feasible generic framework and has been validated through both, the comparison of designs already listed by several different techniques, and the discovery of some new mixed OAs.

6. ACKNOWLEDGEMENTS

My deepest gratitude to Dr Scott H. Murray for his continuous support throughout the development of this project. Many thanks also to Dr Judith Ascione for her very useful assistance during the early stages of preparing this manuscript.

7. REFERENCES

[1] Nguyen M, “Computer-algebraic methods for the construction of designs of experiments”. Technische Universiteit. Eindhoven, Eindhoven University Press, 2005.

[2] Blanchard BS and Fabrycky WWJ, “Systems engineering and analysis”, Prentice-Hall international series in industrial and systems engineering, 2011.

[3] Ross PJ, “Taguchi techniques for quality engineering”, McGraw-Hill Education (India), Pvt Limited, 2005.

[4] Tsai JT, “Robust optimal-parameter design approach for tolerance design problems”, Engineering Optimization, 42(12):1079 - 1093, 2010.

[5] Dummit DS and Foote RM, “Abstract Algebra”, Wiley, 2003.

[6] Hedayat AS, Sloane NJA, and Stufken J, “Orthogonal arrays: theory and applications”, Springer series in statistics, Springer-Verlag, 1999.

[7] Montgomery DC, “Design and Analysis of Experiments”, Wiley, 2012.

[8] Lekivetz R and Tang B, “Multi-level orthogonal arrays for estimating main effects and specified interactions”, Journal of Statistical Planning and Inference, 2012.

[9] David G and David B, “Graphs for orthogonal arrays and projective planes of even order”, SIAM Journal on Discrete Mathematics, 26(3):1076-1087, 2012.

[10] Schoen ED and Nguyen M, “Enumeration and classification of orthogonal arrays”. Journal of Algebraic Statistics, 2007.

[11] Hulpke A, “Notes on Computational Group Theory”, Department of Mathematics, Colorado State University, 2010.

International Journal of GEOMATE, Jan., 2017, Vol. 12, Issue 29, pp. 66-72.
MS No. 1308j received on July 15, 2015 and reviewed under GEOMATE publication policies.

Pertinent discussion including authors’ closure, if any, will be published in Dec. 2017 if the discussion is received by June 2017.
Corresponding Author: Julio Romero