TNO Science and Industry

TNO | Knowledge for business

Eric Schoen[1,2]
Pieter Eendebak[1]

[1] TNO Science and Industry Delft, Netherlands
[2] University of Antwerp, Belgium

Very fast enumeration of orthogonal arrays

Definition

- Rectangular arrangement of symbols
  Rows // runs
  Columns // factors
  Symbols // factor levels
- Strength *t*:
  every *t* columns form a full factorial, possibly replicated

Cheese	Bacon	Eggs	Broccoli
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0
2	0	0	1
2	0	1	0
2	1	0	0
2	1	1	1
3	0	0	1
3	0	1	0
3	1	0	0
3	1	1	1

Enumeration algorithm

- OA $(8;2^3;2)$
- $t=2$: start with first 2 columns
- Extend with column 3 such that
  - set D_3 contains all LMC arrays
  - set D_3 is as small as is feasible
- test for all arrays in D_3 whether they are LMC
- discard non-LMC arrays

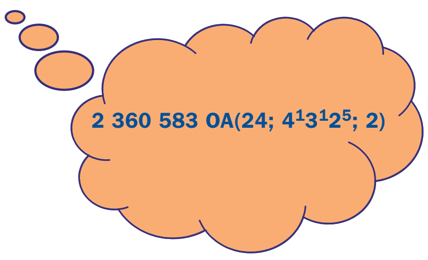
valid extensions

0	0	0	0	0
0	0	0	1	1
0	1	1	0	0
0	1	1	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	1
1	1	0	1	0

isomorphic extensions

Enumeration of inequivalent orthogonal arrays

- For given run size, strength and factor levels: many inequivalent arrays
- Idea: enumerate all inequivalent arrays
- Classify with appropriate criteria
- Choose best array



Features of LMC test

- Single array testing
- Find smaller array by permuting original array
  - column
  - level
  - row — sorting always makes an array smaller
- Two stages:
  1. Root-stage:
     - first *t* columns are always the same
     - each level permutation in root has a stored row sorting that returns the root
  2. Non-root stage: remaining column and level permutations

Equivalence of orthogonal arrays

- Isomorphism class:
  Switching of rows
  Switching of symbols
  Switching of columns
- Idea: keep 1 array per isomorphism class

OA$(8; 2^3; 2)$

0	0	0	0	0	0
0	0	1	0	0	0
0	1	0	0	1	1
0	1	1	0	1	1
1	0	0	1	0	1
1	0	1	1	0	1
1	1	0	1	1	0
1	1	1	1	1	0

Very fast enumeration?

- Factor 7-2122 faster for 36 out of 37 cases in only explicit literature example
- Computing times in literature paper strictly for factor-numbers stated
- Our computing times also include all arrays with less factors

Which array do we keep?

- Lexicographically smallest one:
- array 1 is lexicographically smaller in columns:
- Column 1+2 identical
- Column 3 has smaller symbol in 5th position
- keep LMC array: smallest array of isomorphism class

array 1			array 2		
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	0
0	1	1	0	1	1
1	0	0	1	0	1
1	0	1	1	0	0
1	1	0	1	1	0
1	1	1	1	1	0

Completed cases

strength	exceptions
2: $N \leq 28$	OA$(28; 2^a)$
3: $N \leq 72$	OA$(N; 2^a; 3)$; $N = 56, 64, 72$ OA$(64; 4^1 2^a; 3)$, OA$(72; 3^1 2^a; 3)$
4: $N \leq 168$	OA$(160; 2^a; 4)$

- All cases with run-size up to the stated bounds are enumerated…
- … and some additional cases
- The arrays are currently under investigation

References

• Schoen E.D. (2009). All orthogonal arrays with 18 runs. *Quality and Reliability Engineering International* **25**, 467-480. • Schoen E.D., Eendebak P.T., and Nguyen M.V.M. (2009) 'Complete enumeration of pure-level and mixed-level orthogonal arrays'. *Journal of Combinatorial Designs*, to appear. • Bulutoglu, D.A., and F. Margot (2008) 'Classification of orthogonal arrays by integer programming'. *Journal of Statistical Planning and Inference* **138**, 654-666.