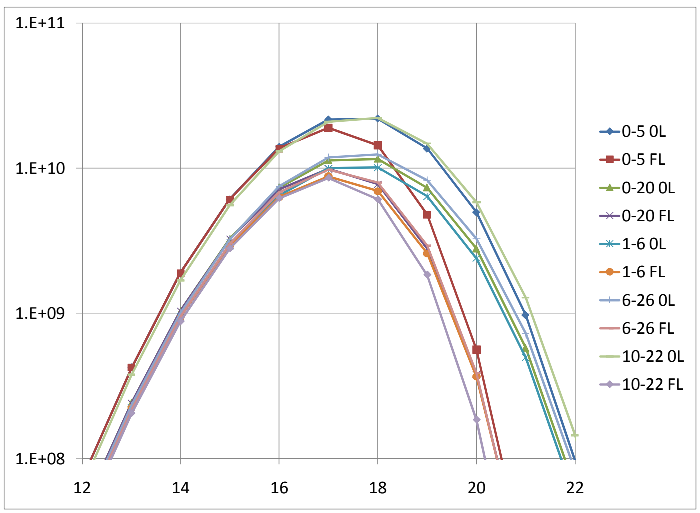
CISS
2010

Costas array search technique that maximizes backtrack and symmetry exploitation



Jon C Russo

Keith G Erickson

James K Beard

March17, 2010 ,11:00 AM

Abstract

CISS 2010

- Costas search techniques
  - Generators don't find all of them
  - We present two innovations that improve speed
- Innovations presented here
  - Essentially full exploitation of symmetry
  - Multiple level look-ahead preclusion
- Advantages gained
  - Well-known symmetry gains a factor of two
  - New symmetry gains approach a factor of four
  - Look-ahead gains approach a factor of two
  - Overall, factor of four over older methods

Topics Today

CISS 2010

- Background
  - Backtrack programming with preclusion
    - The difference table
    - Generating the preclusion table
  - Symmetry and sets related by rotation and transposition
    - Sets of eight Costas arrays, when asymmetrical
    - Symmetry causes duplications; only four to a set
- Innovation: full symmetry exploitation
- Innovation: multi-level look-ahead preclusion
- Results
  - Overall a factor of four improvement
  - The Last Costas Array?
- Costas arrays of large orders

The Difference Table

CISS 2010

- Example: {4,2,5,1,3}
  - Row *nr* is difference between columns *nc+nr* and *nc*, given as *D<nr>(nc)=CA(nc+nr)-CA(nc)*

Col	**1**	**2**	**3**	**4**	**5**
CA	4	2	5	1	3
D1	-2	3	-4	2	
D2	1	-1	-2		
D3	-3	1			
D4	-1				

The Preclusion Table

CISS 2010

- Getting Started: Given two column indices

Col	1	2	3	4	5
CA	4	2	5	1	3
D1	-2	3	-4	2	
D2	1	-1	-2		
D3	-3	1			
D4	-1				

- Use the one available difference to preclude values of the third column index that would result in a duplication

How the Preclusion Table Works

CISS 2010

- Begin with the difference matrix

Col	1	2	3	4	5
CA	4	2	5	1	3
D1	-2	3	-4	2	
D2	1	-1	-2		
D3	-3	1			
D4	-1				

- Note that D1(2) is

$$D1(2) = CA(3) - CA(2) \neq D1(1)$$

- So, we have

$$CA(3) \neq D1(1) + CA(2)$$

The First Preclusion Table

CISS 2010

- Table of precluded values of third column index

Col	1	2	3	4	5
CA	4	2	5	1	3
D1	-2	3	-4	2	
D2	1	-1	-2		
D3	-3	1			
D4	-1				

Row Index	Reason
0	D1(1)+CA(2)
1	
2	Taken
3	
4	Taken
5	

The Second Preclusion Table

CISS 2010

- Table of precluded values of fourth column index

Col	1	2	3	4	5
CA	4	2	5	1	3
D1	-2	3	-4	2	
D2	1	-1	-2		
D3	-3	1			
D4	-1				

Row Index	Reason
2	Taken
3	D1(1)+CA(3), D2(1)+CA(2)
4	Taken
5	Taken
6	
7	
8	D1(2)+CA(3)

The Third Preclusion Table

CISS 2010

- Table of precluded values of the fifth column index

Col	1	2	3	4	5
CA	4	2	5	1	3
D1	-2	3	-4	2	
D2	1	-1	-2		
D3	-3	1			
D4	-1				

Row Index	Reason
-3	D1(3)+CA(4)
-2	
-1	D1(1)+CA(4), D3(1)+CA(2)
0	
1	Taken
2	Taken
3	
4	Taken, D1(2)+CA(4), D2(2)+CA(3)
5	Taken
6	D2(1)+CA(3)

Bitmasks for Rows of the Difference Matrix

CISS 2010

- The difference table consists of numbers that may be anywhere from –(N-1) to (N-1)
- The Costas condition is that no entry is ever allowed to repeat
  - Bit 32 of a 64-bit register represents a difference of zero
  - Bit positions in a 64-bit register can be used to represent a row of a difference matrix for searches up to order 33

Bitmask for the Preclusion Table

CISS 2010

- The preclusion table is shown for all values
- Only row indices from 0 to N-1 are significant in the implementation
  - A 32-bit mask is sufficient for searches up to order 32
  - Initialize with the rows used up to that point in the search
- For each available row of the difference matrix
  - Shift the difference mask by the row indices
  - Update the preclusion mask with a logical OR

Architecture-Independent Efficiency Measure

CISS 2010

- Flow of the search method is
  - Available values of the preclusion table are accepted as row indices
  - A new preclusion table for the next row index is constructed
  - Drop back to previous row index and table when all row indices have been used in current table
- The search method is conceptually recursive
- A count of the recursion levels entered is a measure of resources required to perform the search

CISS 2010

Symmetry

- {4,2,5,1,3}

	1	2	3	4	5
1				•	
2		•			
3					•
4	•				
5			•		

- {4,2,3,5,1}

	1	2	3	4	5
1					•
2		•			
3			•		
4	•				
5				•	

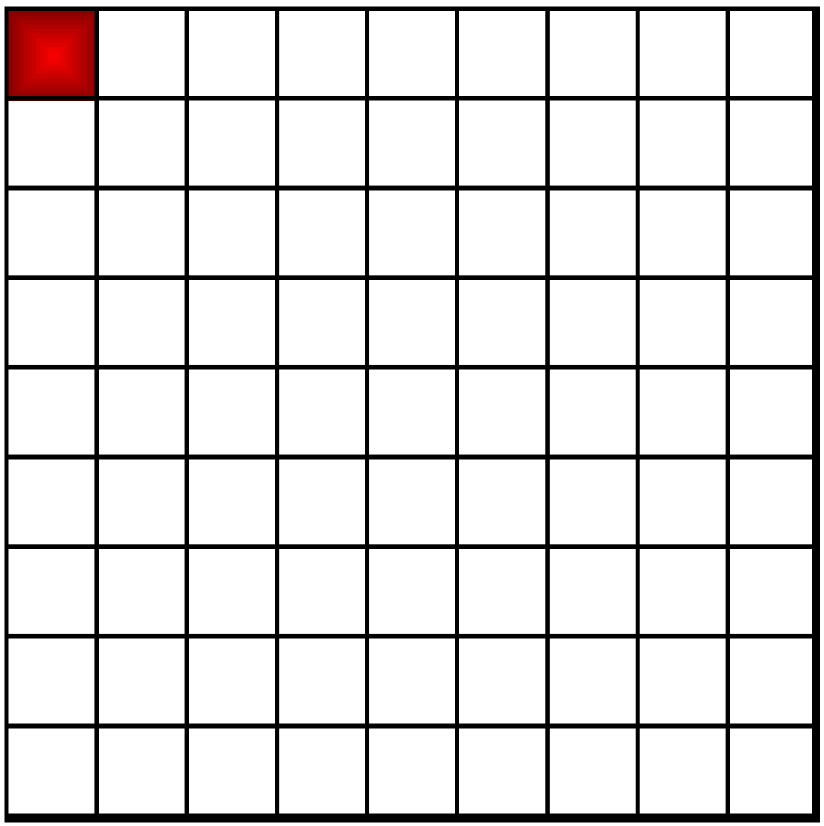
Sets of Four or Eight

CISS 2010

- Costas arrays formed from other Costas arrays by rotation and transposition are defined as being in a set
- We call Costas arrays of the same set are polymorphs of each other
- Symmetrical Costas arrays belong to sets of four
- Non-symmetrical Costas arrays belong to sets of eight

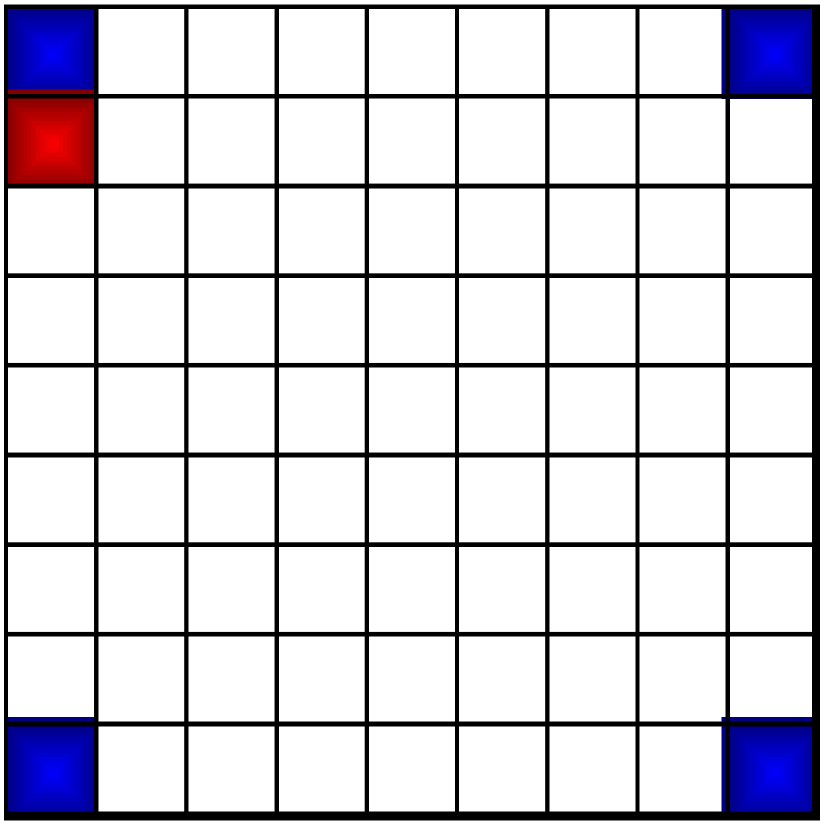
Full Exploitation of Symmetry

CISS
2010



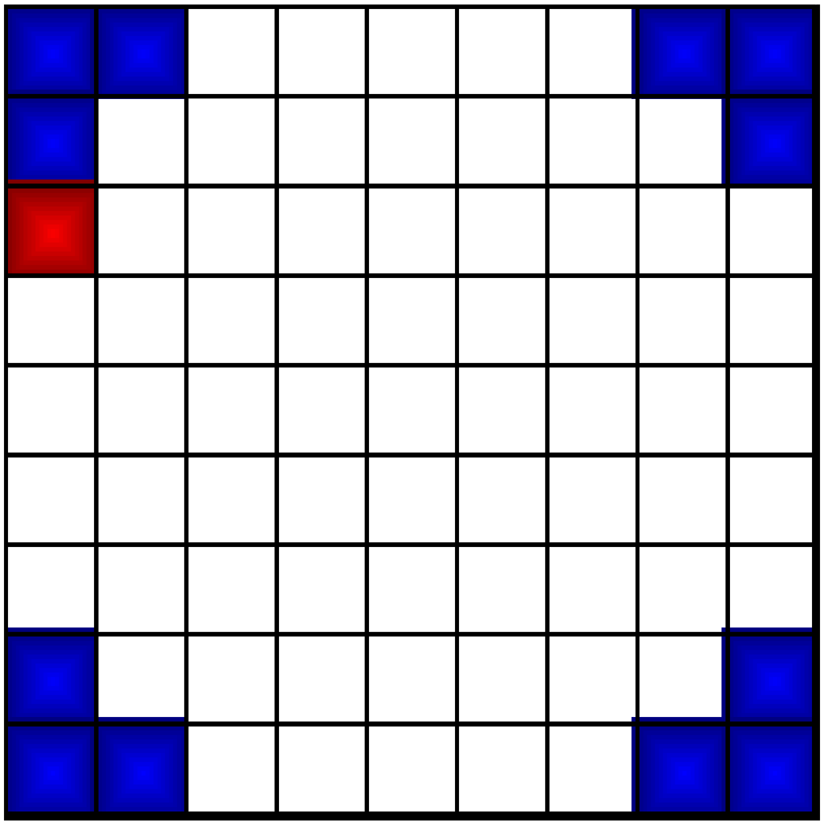
- Search starts with dot in upper left-hand corner
- Finds all possible arrays with this initial dot fixed
- Can be accomplished by corner-dot extending arrays of order N-1
- Omission of this case offers some gains

Exploitation of Symmetry (2)



- Search continues with dot in position (1,2)
- All corner dots may be eliminated from search
- All sets of Costas arrays with a corner dot have already been found

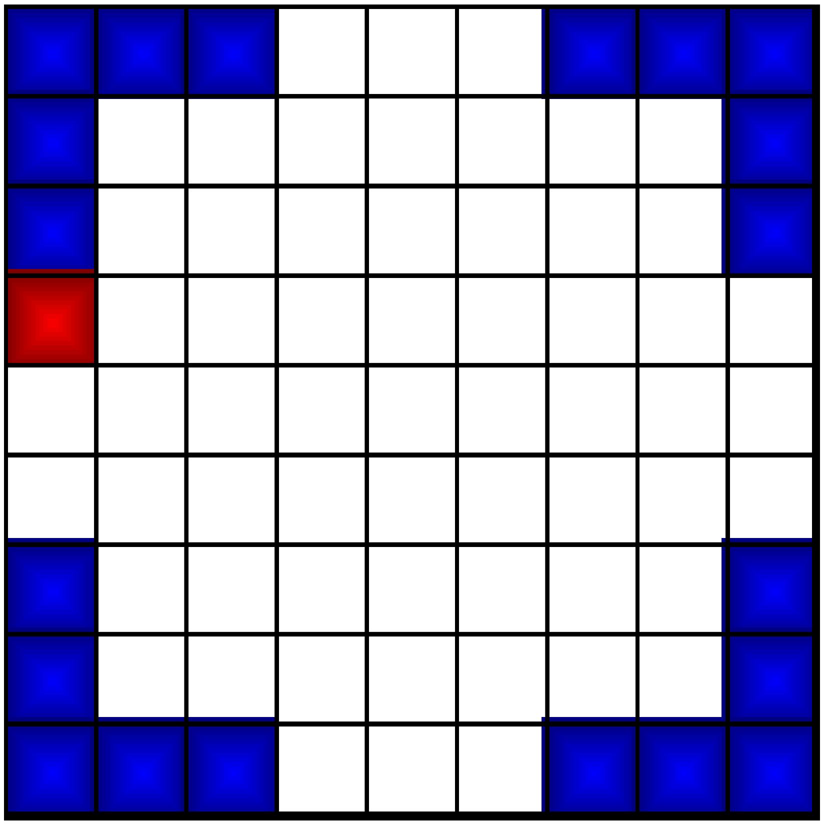
Exploitation of Symmetry (3)



- Continue with dot in position (3,1)
- Corner dots and dots one away from corner are eliminated from search
- Implement by modifying initialization of the preclusion mask

Exploitation of Symmetry (4)

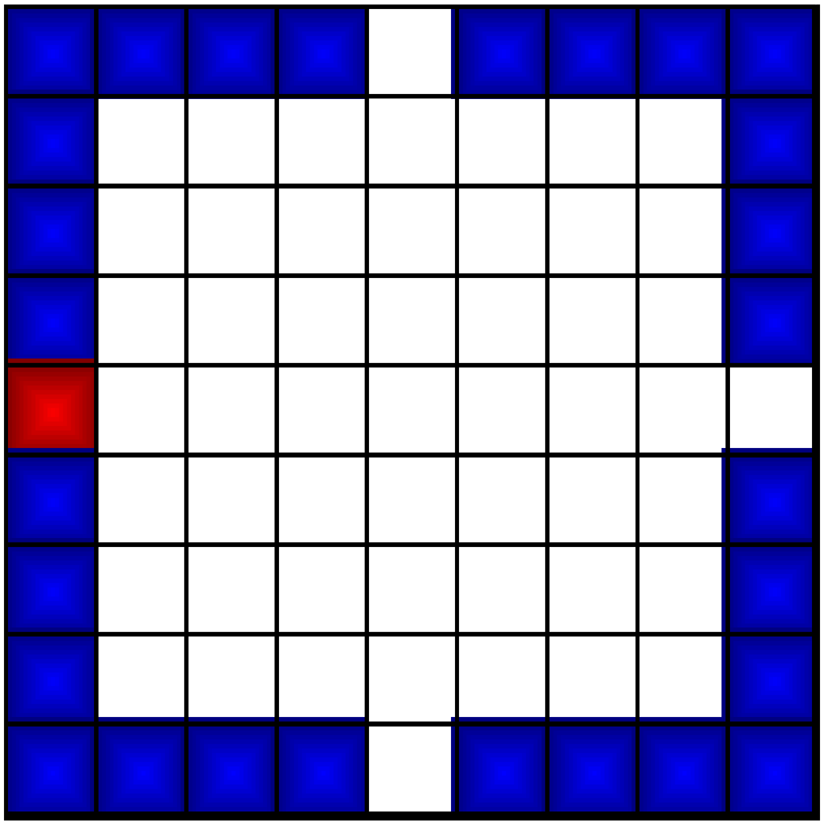
CISS 2010



- Continue with dot in position (4,1)
- All edge dots 0 through 2 away from a corner are eliminated
- In general, edge dots closer to any corner than the current fixed dot are eliminated
- They are guaranteed to have been covered by previous row-1 dot position searches

Exploitation of Symmetry (5)

CISS 2010



- Center dot not necessary
- There is no place for a[N-1]
- The range of a[0] from 1 to [(N-3)/2] is commonly used

Using More Look-Ahead Levels in Preclusion

CISS 2010

- A given recursion level terminates when its preclusion table is exhausted or filled
- The preclusion table is also computed for the next row index, which is checked for no available row indices
- The preclusion table can be computed for the row index after next, too
- We compute preclusion tables to the last available row

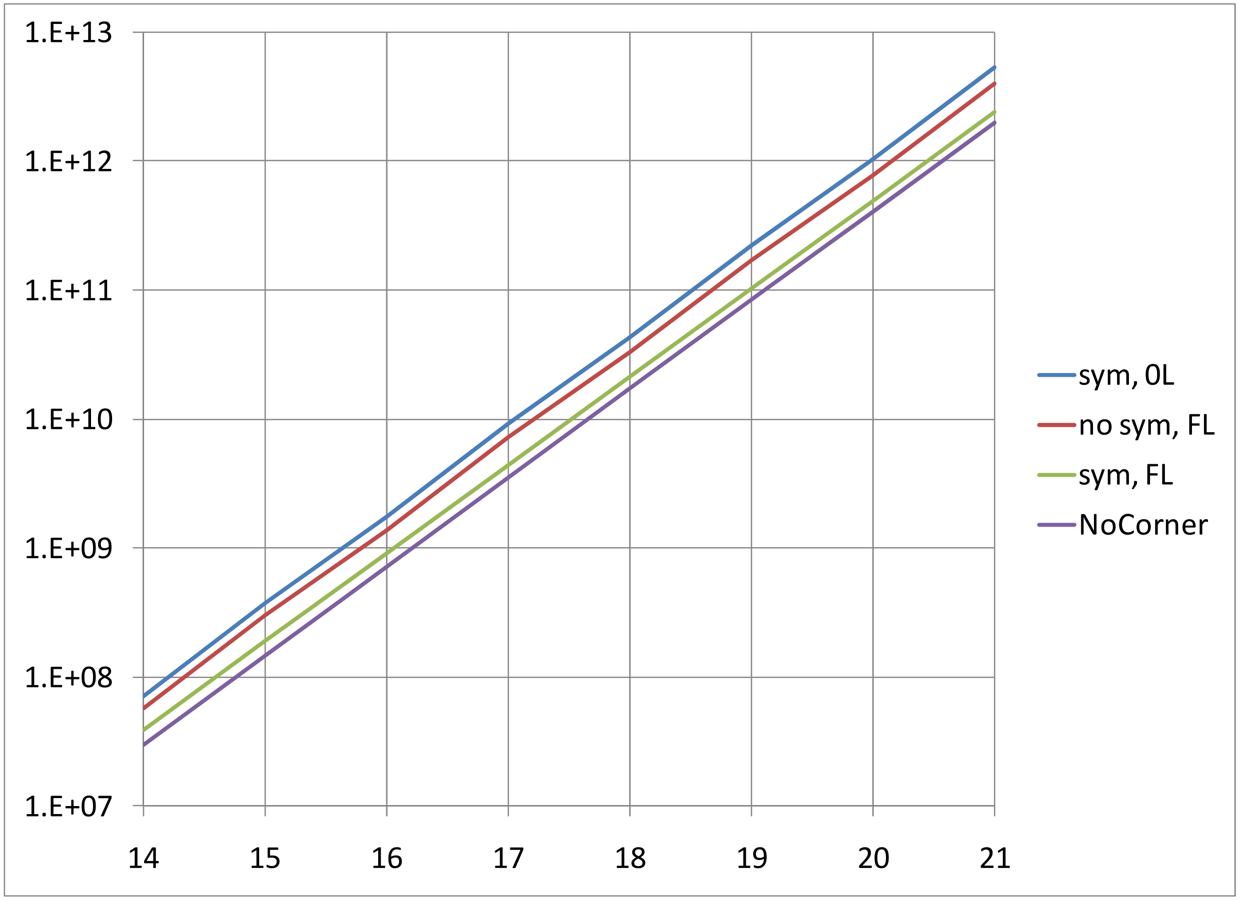
Recursion Counts

CISS 2010

Recursion Counts versus Order

With and Without Symmetry Exploitation, Full Look-Ahead

Counts include Starting with Corner dot Except Lowest Plot



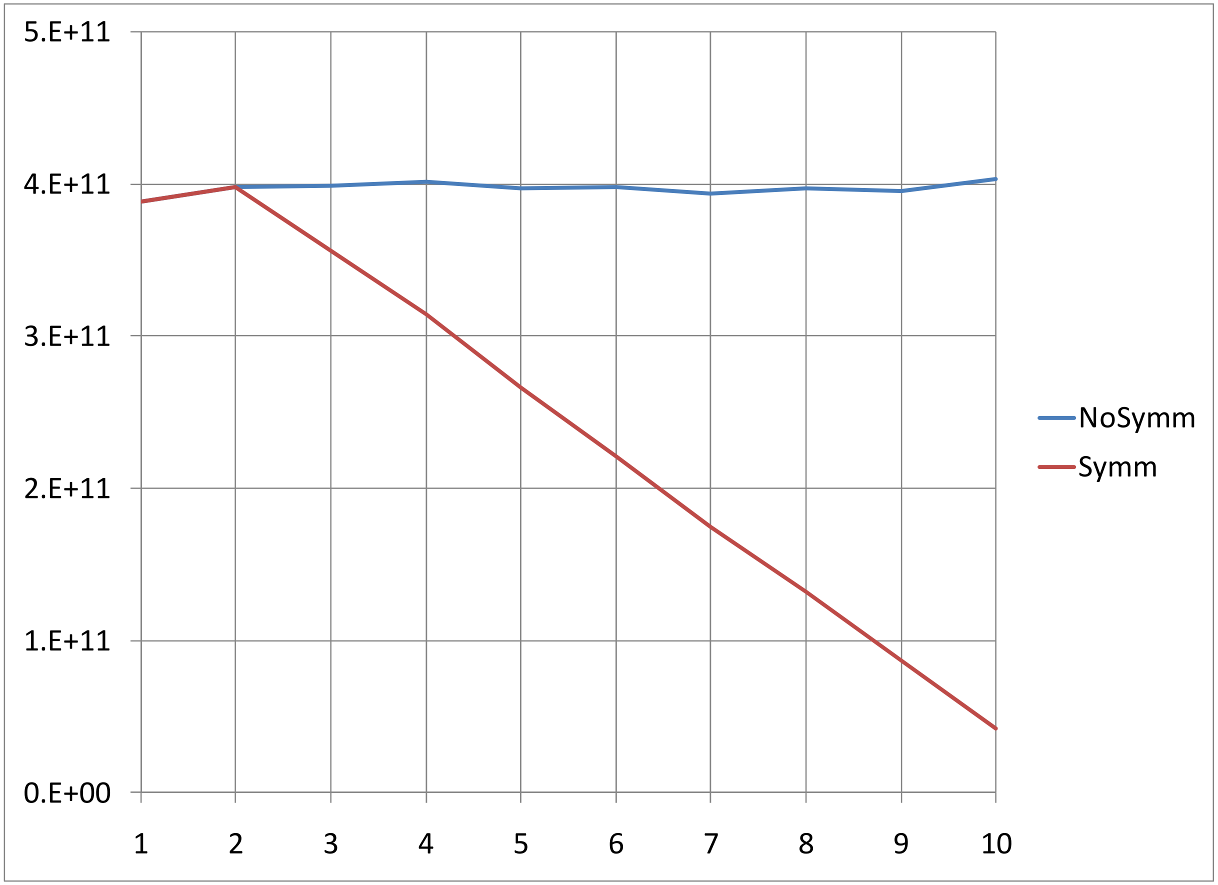
How Does Symmetry Exploitation Do the Job?

CISS 2010

Recursion Counts Versus Row 1 Dot Position for Order 21

Abscissa: a[1] is corner dot

Overall factor of 0.45

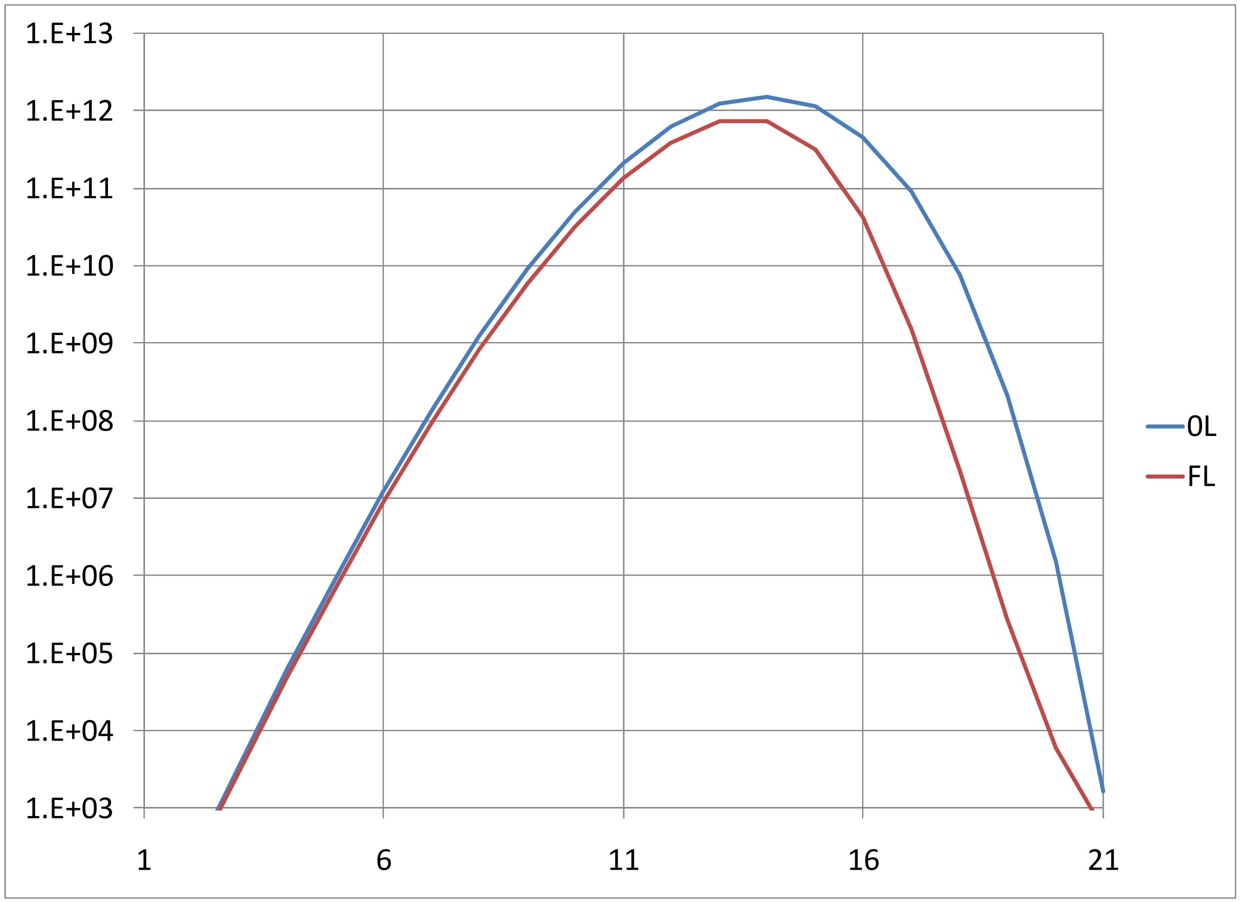


How Does Look-Ahead Do The Job?

CISS 2010

Recursion Counts Versus Recursion Level For Order 21

Overall factor of 0.60

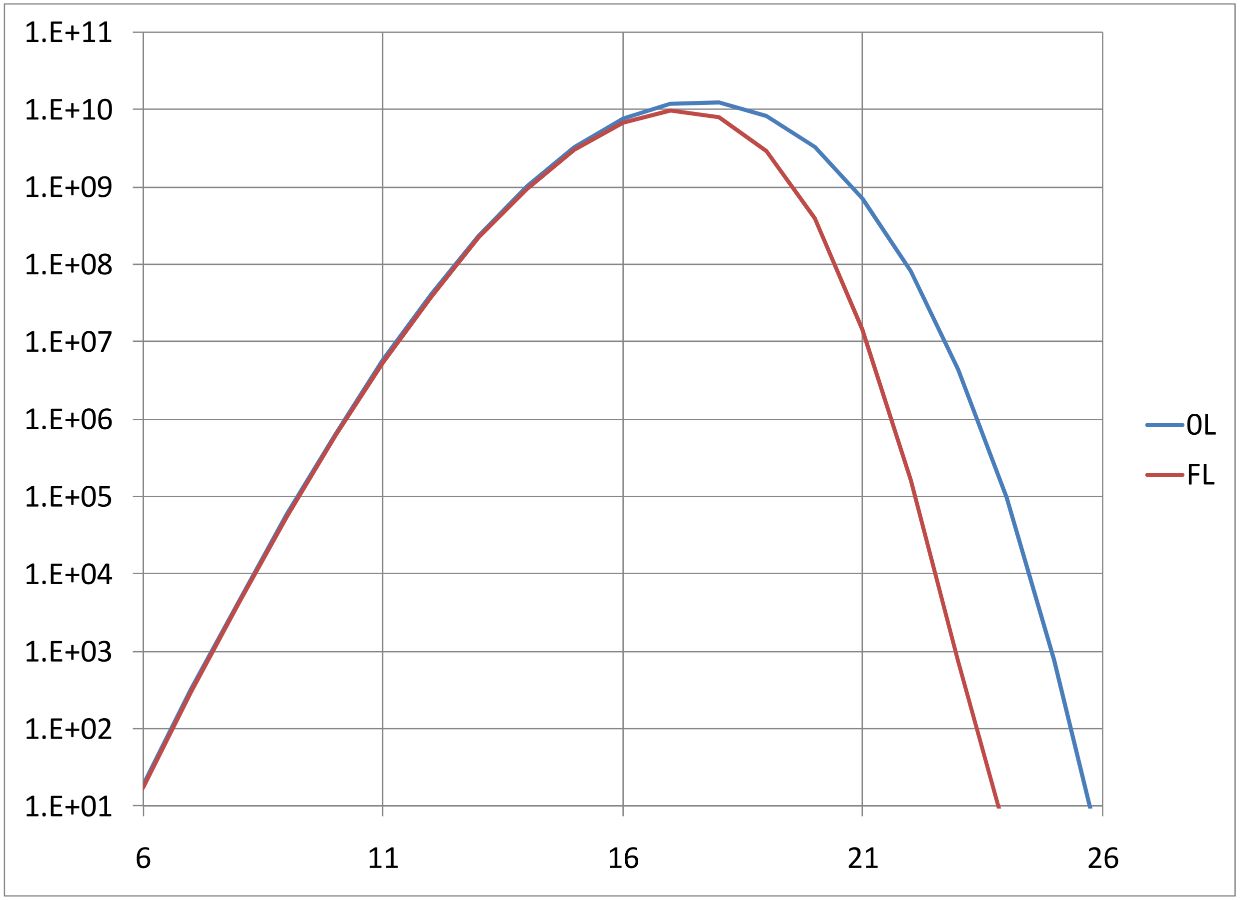


What Can We Expect for Higher Orders?

CISS 2010

Recursion Counts Versus Recursion Level For Order 28

Search only over {6,26,23,13,15…}

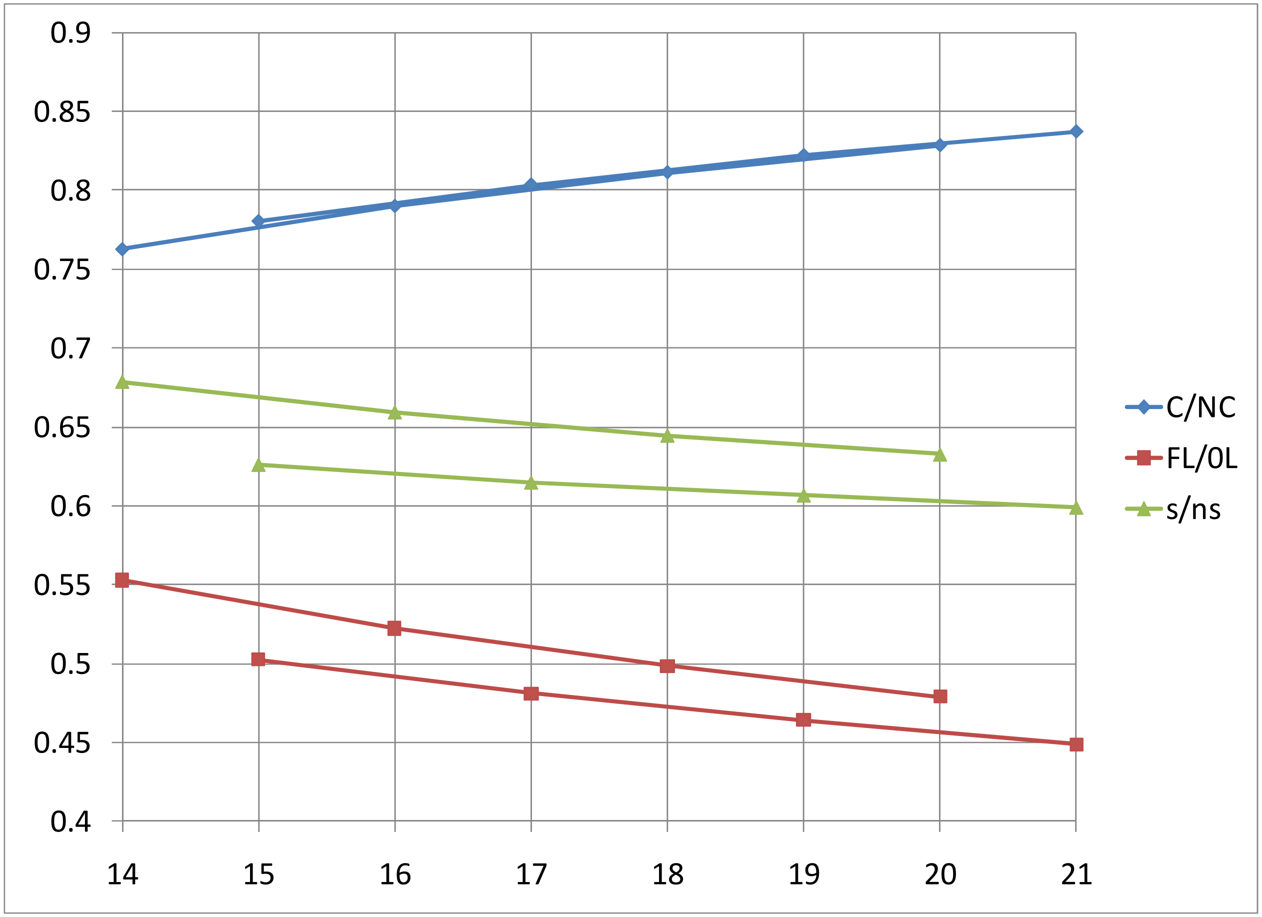


What Gains Are Obtained?

CISS 2010

Gains in Recursion Counts vs. Order

Note that odd and even orders are plotted separately



Analyzing the Curves for Orders 14 through 21

CISS 2010

- Least-squares fit, RMS errors 0.0005 or smaller
- Look-Ahead Gains
  - For even order

$$f_{LE}\left(N\right)=0.3074+\frac{3.437}{N},\quad f_{LE}(28)=0.43$$

  - For odd order

$$f_{LO}\left(N\right)=0.3161+\frac{2.8}{N},\quad f_{LO}\left(28\right)=0.42$$

- Symmetry exploitation gains
  - For even order

$$f_{SE}\left(N\right)=0.5253+\frac{2.147}{N},\quad f_{SE}\left(28\right)=0.60$$

  - For odd order

$$f_{SO}\left(N\right)=0.5319+\frac{1.415}{N},\quad f_{SO}\left(28\right)=0.58$$

- Omission of corner dot

$$f_{C}\left(N\right)=0.98-\frac{3}{N},\quad f_{C}\left(28\right)=0.872$$

Observed Look-Ahead Gains at Order 28

CISS 2010

Case	Gain Ratio
{0,5,25,10,26…}	0.71
{1,6,9,5,20…}	0.73
{6,26,23,13,15…}	0.66
(10,22,26,8,2…}	0.31

Overall Gain Ratio of About 0.43 is Reasonable

Total Gains for Order 28

CISS 2010

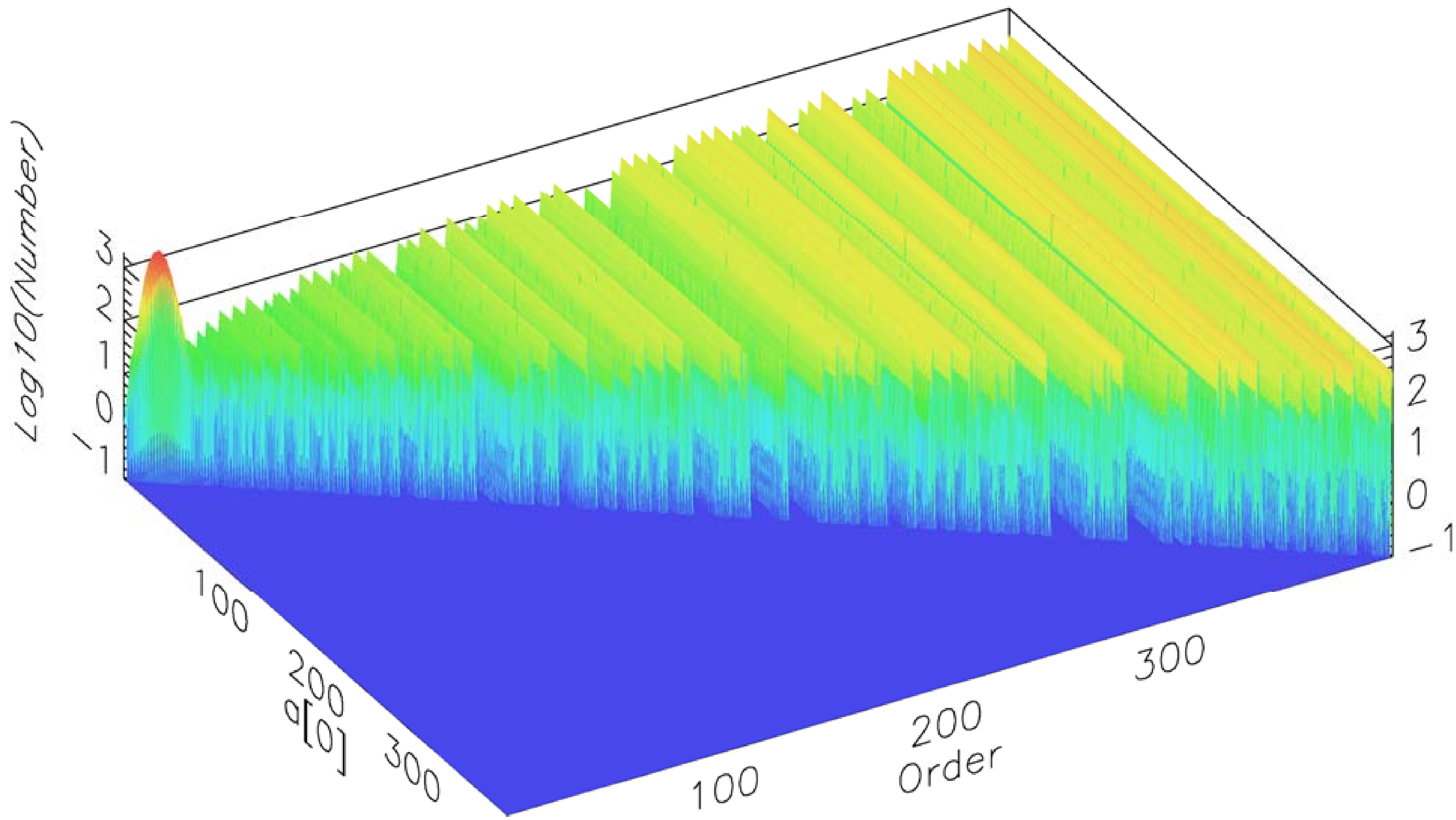
- From Look-Ahead
  - Extrapolation from curves for orders 14 – 21 estimates a factor of 0.43
  - Corroborated by short sections for actual searches
  - About 2:1 is a very conservative estimate
- From Symmetry exploitation
  - Extrapolation from curves for orders 14 – 21 estimates a factor of 0.60
  - Measured at 0.55 for orders 27 and 28
  - Factor of 0.872 from omission of corner dot
  - On top of 2:1 conventionally obtained by limiting search range of a[0] from 0 to [(N-1)/2]
- TOTAL
  - 0.60 X 0.43 X 0.872 = 0.225, not including conventionally obtained 2:1
  - Some gains from omitting corner dot

A Look at Symmetry Filtering

- All Costas arrays, orders three through 400
  - Contour plot, occurrence versus *a[0]*
  - What full symmetry exploitation accomplishes
- Zoom in to look at orders 3 through 28
- A look at “spurious” Costas arrays

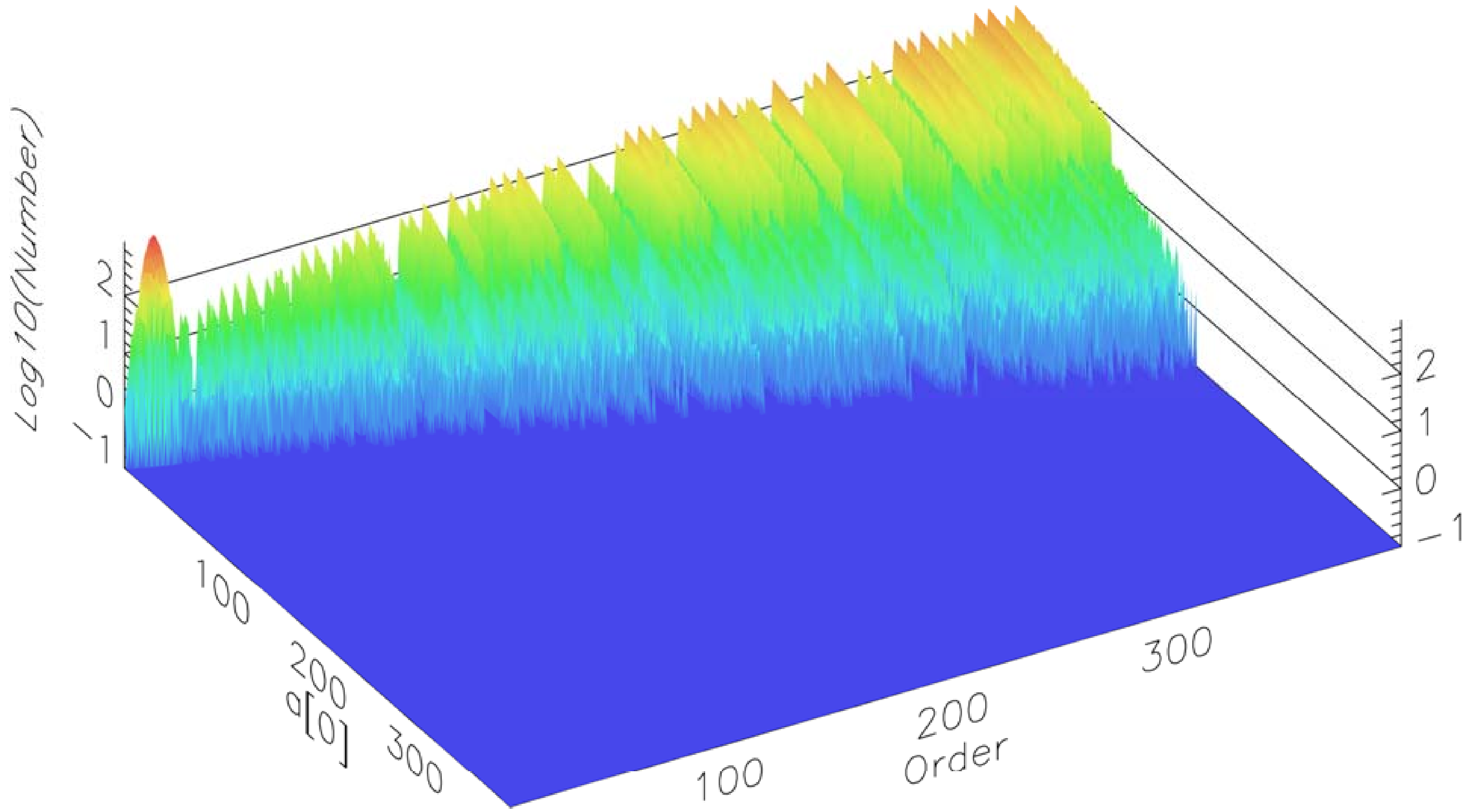
All Costas Arrays

CISS 2010



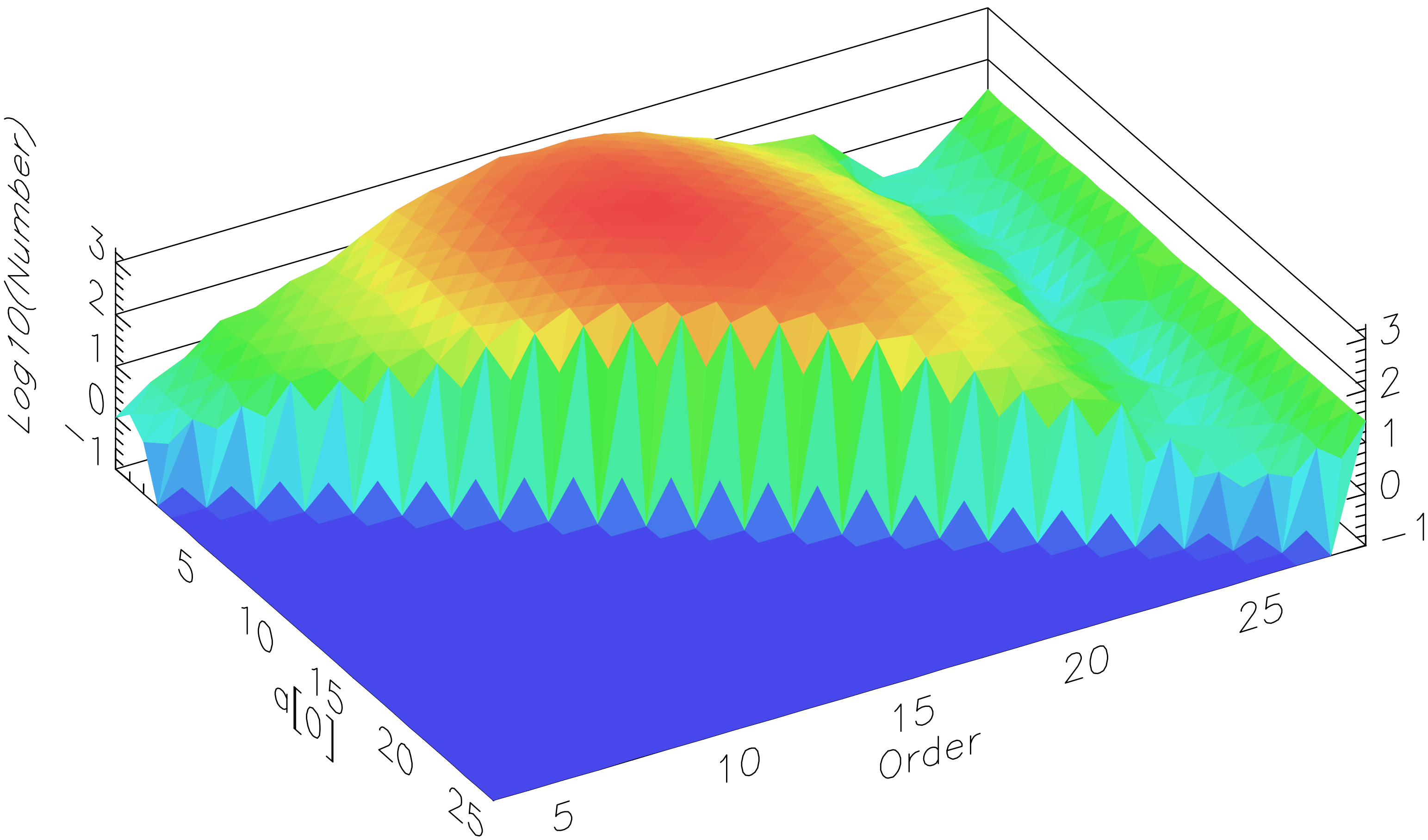
Symmetry Filtered Costas Arrays

CISS 2010



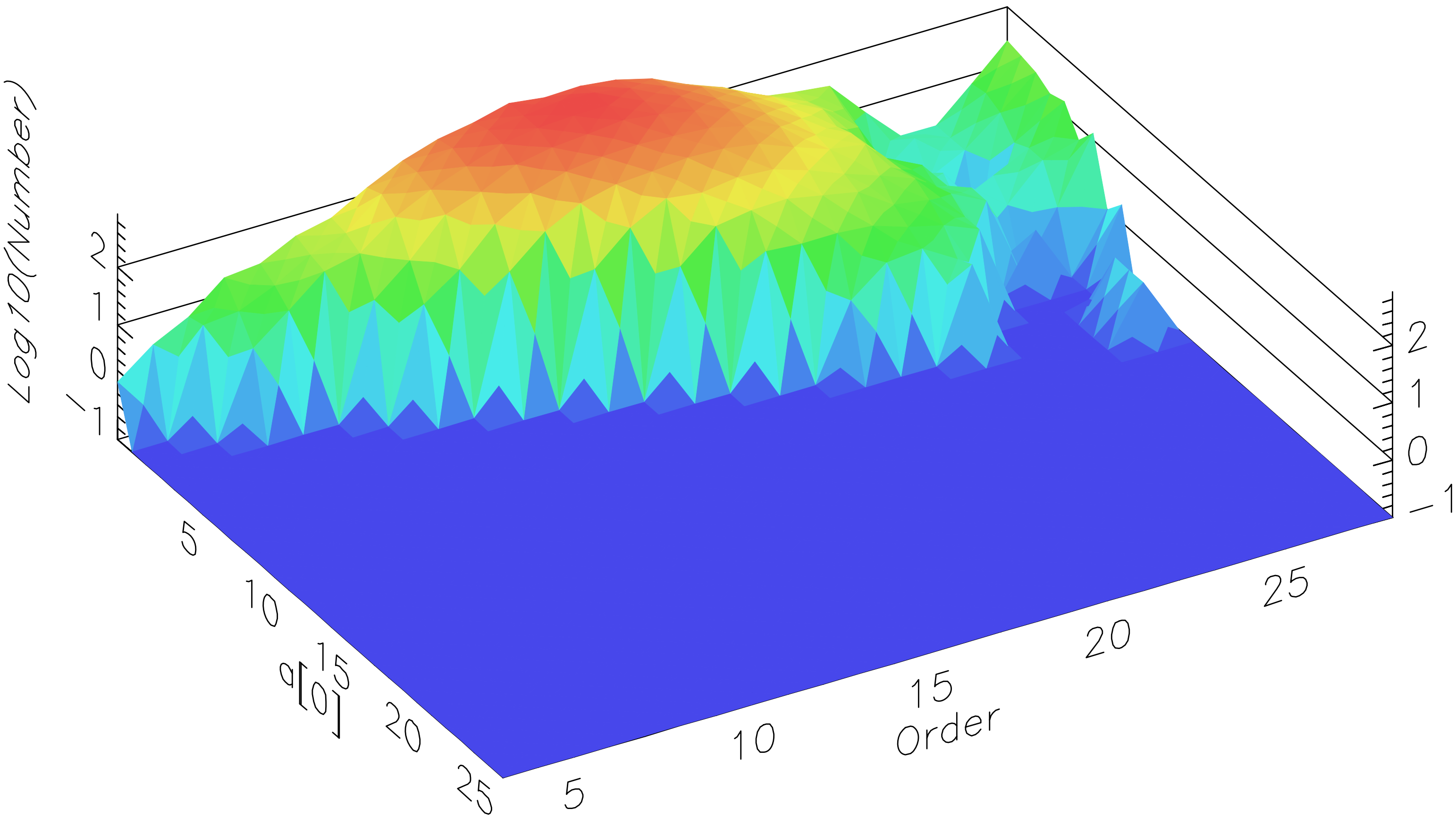
All Costas Arrays

CISS
2010

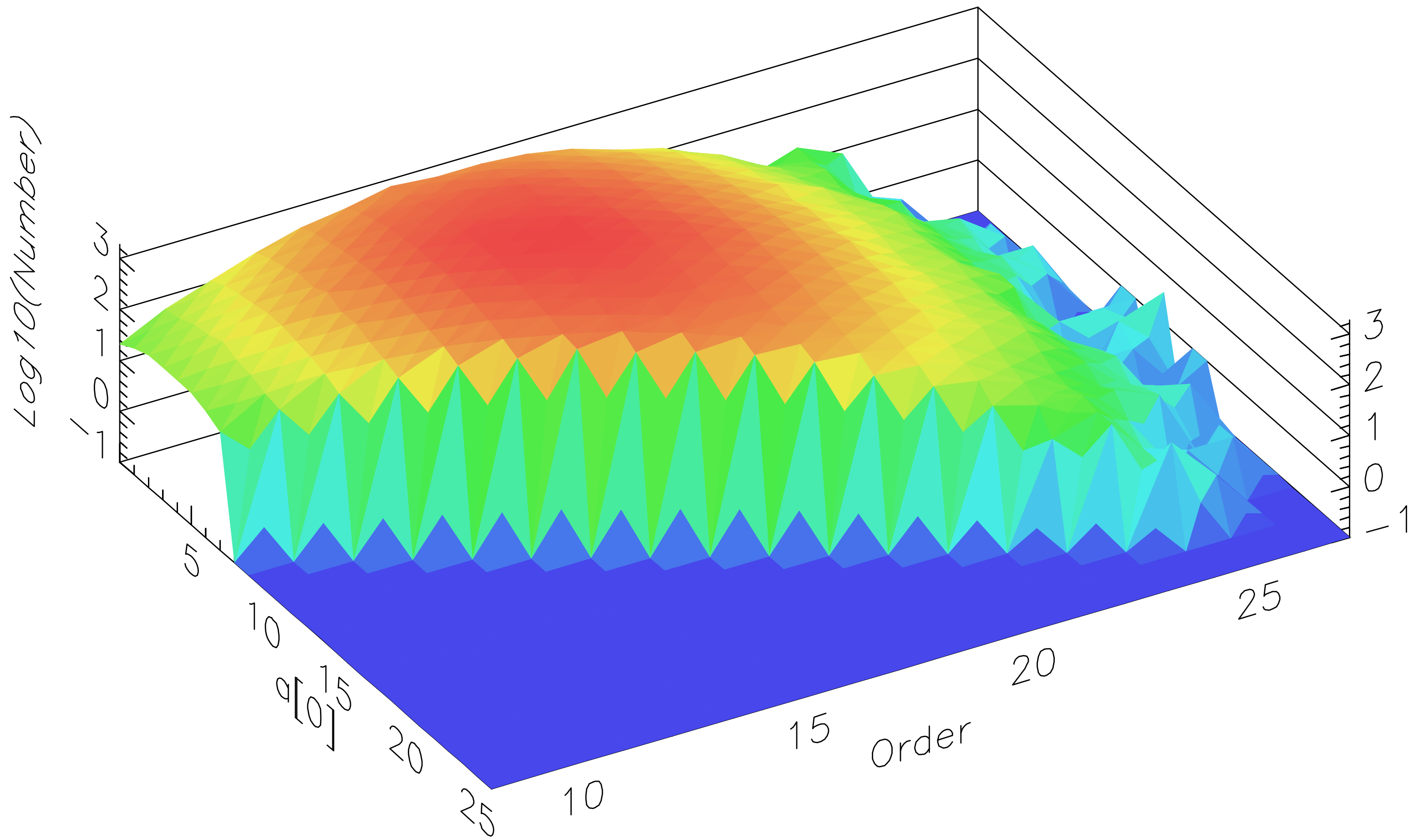


Symmetry Filtered Costas Arrays

CISS 2010

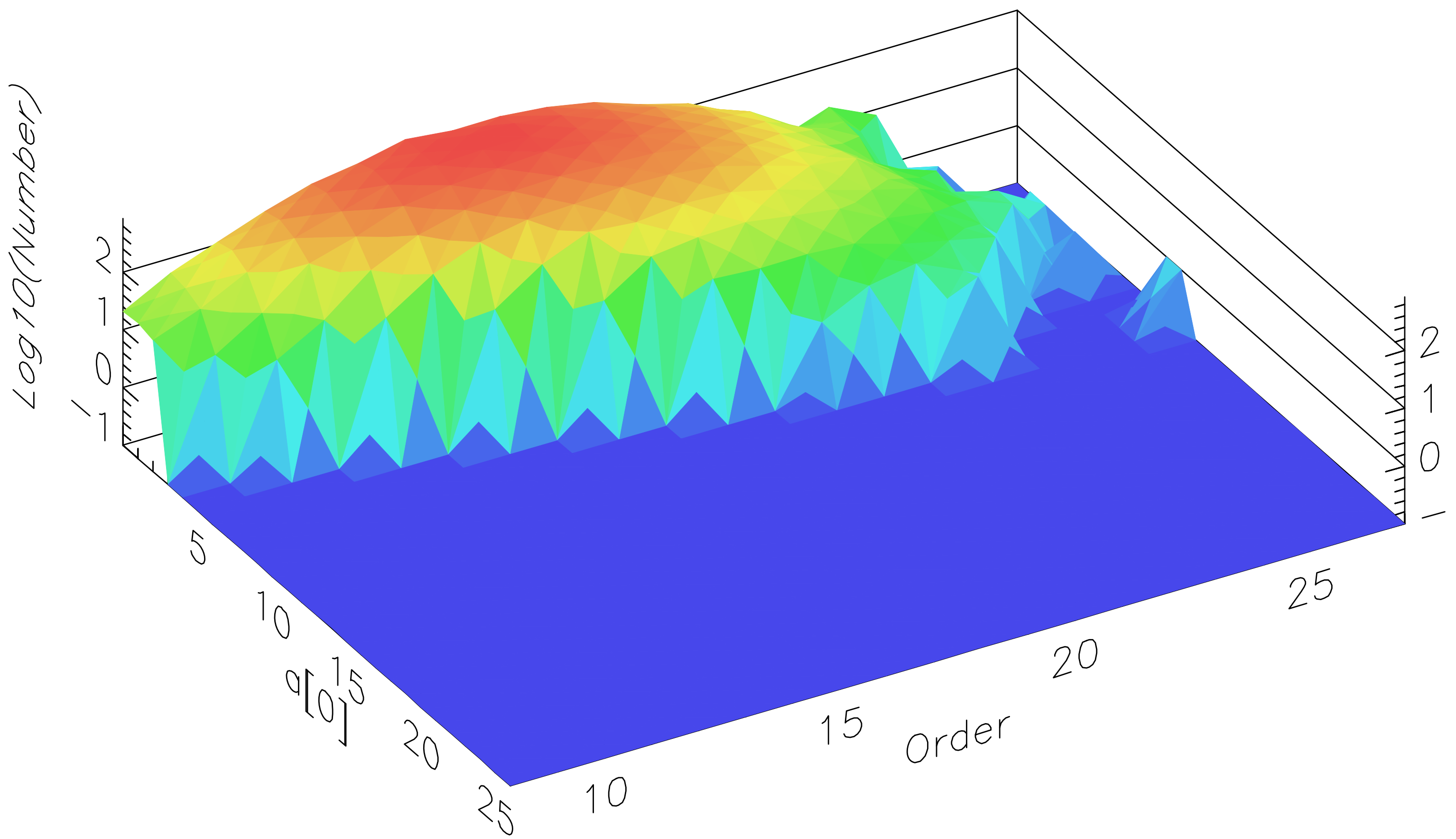


Spurious Costas Arrays



Symmetry Filtered Spurious Costas Arrays

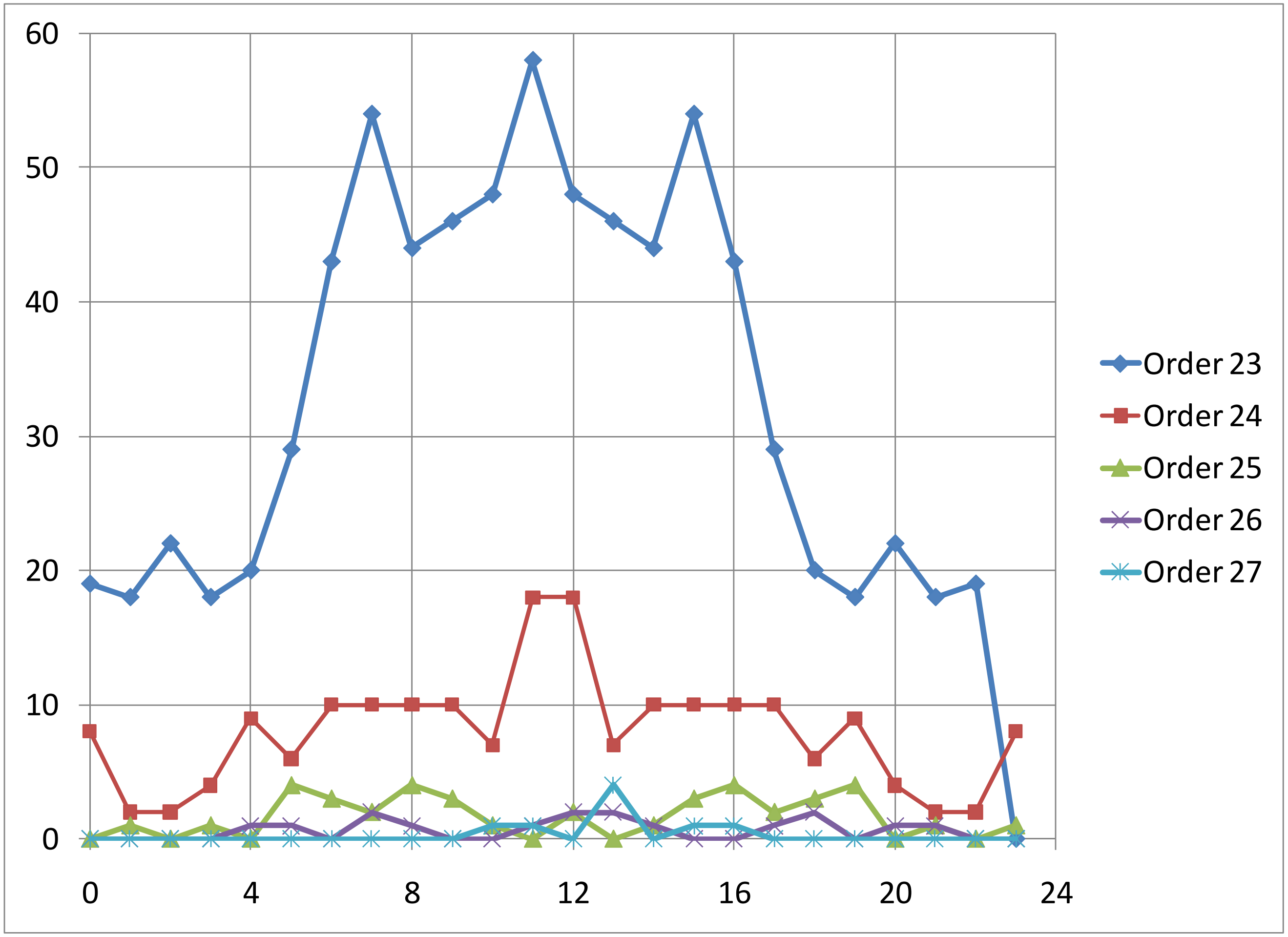
CISS 2010



Costas Arrays Found by Search

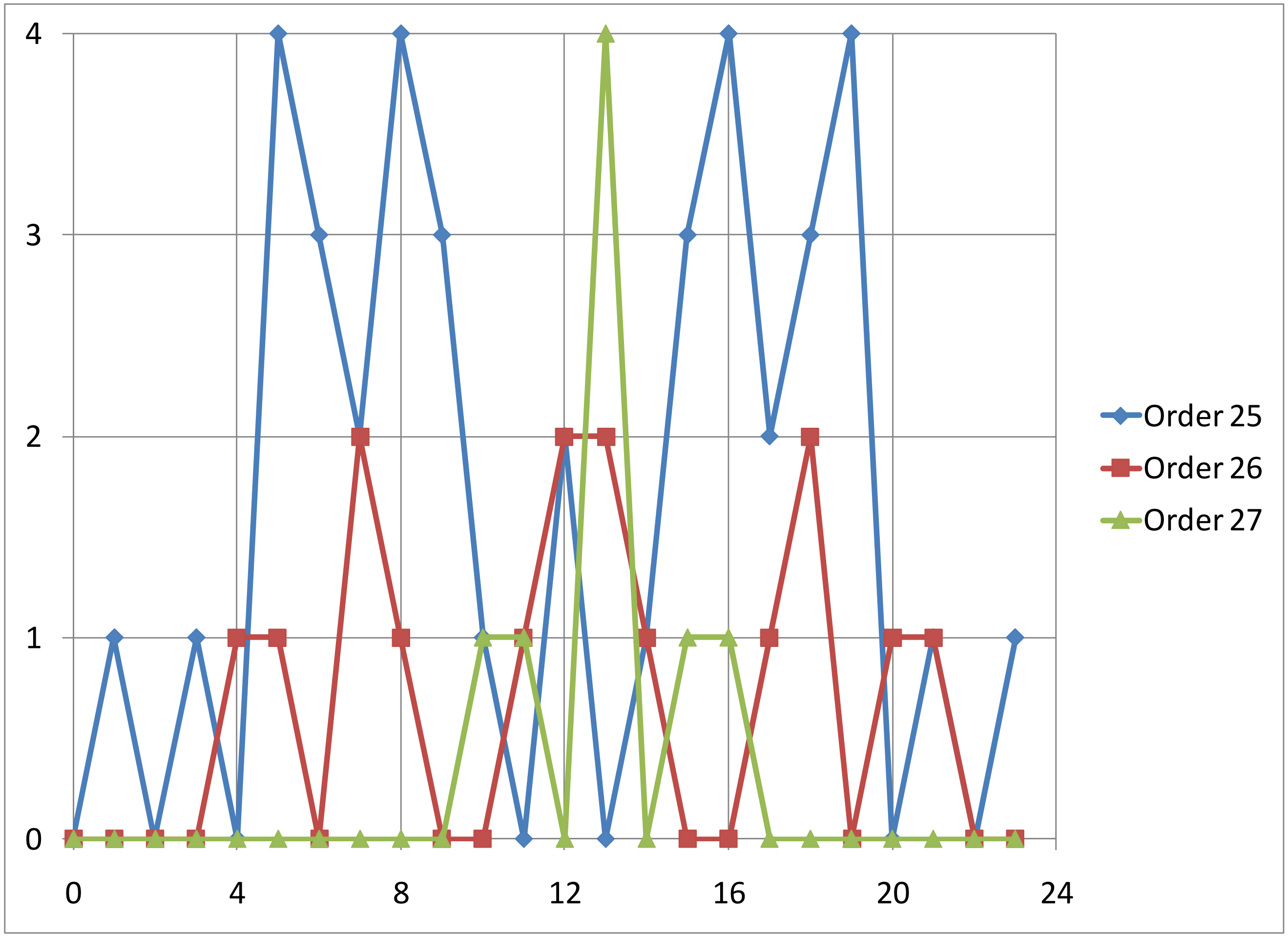
CISS 2010

Number Found vs. a[0]



Costas Arrays Found by Search

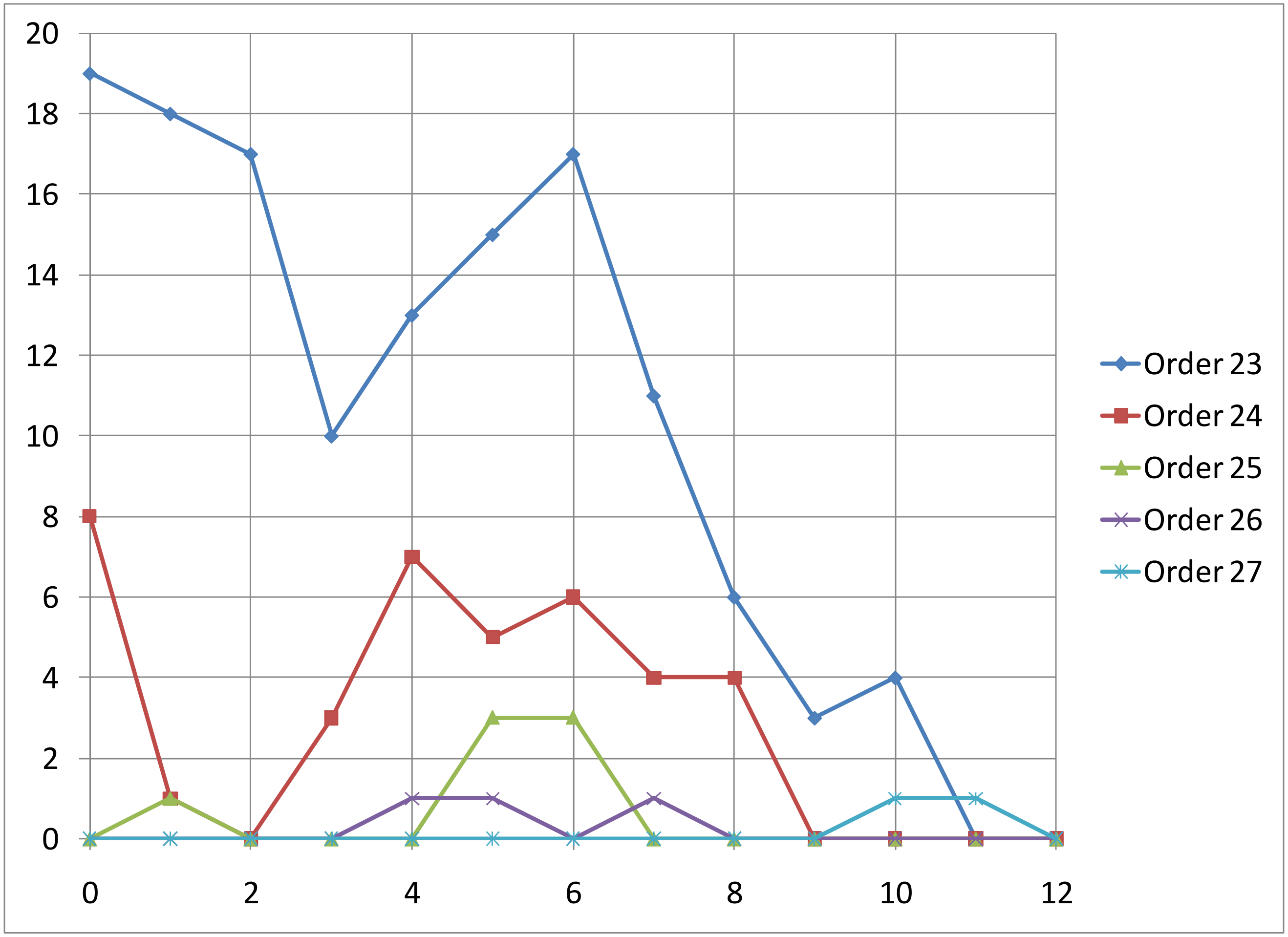
Number Found vs. a[0]



Costas Arrays Found by Search, Symmetry Filtered

CISS 2010

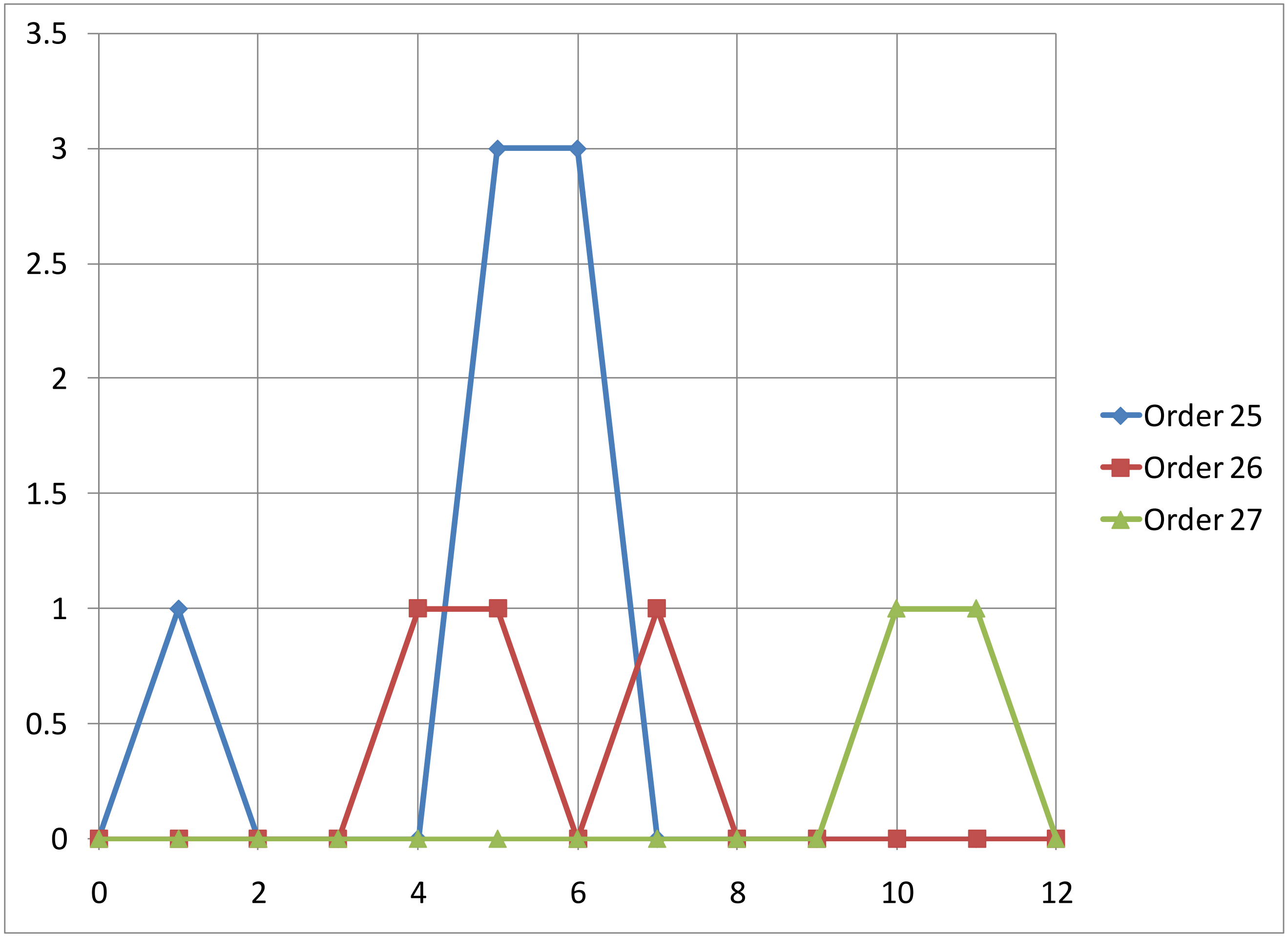
Number Found vs. a[0]



Costas Arrays Found by Search, Symmetry Filtered

CISS 2010

Number Found vs. a[0]



The Last Costas Array

- Found by two teams in Spring of 2008
  - Drakakis, et al., in a log dated March 9 (Euro supercomputer)
  - Beard, et al., found April 8, announced May 29 (personal resources and resources of opportunity)
- Keith G Erickson
  - Identified and accessed resources of opportunity
  - Revised allocations to deal with unique restrictions on the use of this resource
  - Executed allocated searches and found the Costas Array

The Last Costas Array

CISS 2010

- Costas array of order 27
- Few others of order over 26 have ever been found
- Likelihood that any others exist is slight

11	10	4	24	7	23	3	18	21	9	26	16	5	1	15	27	2	25	17	22	19	6	8	12	20	13	14
12	17	10	24	22	8	19	3	7	20	9	16	13	1	2	4	27	26	18	5	23	6	15	25	21	11	14
14	11	21	25	15	6	23	5	18	26	27	4	2	1	13	16	9	20	7	3	19	8	22	24	10	17	12
14	13	20	12	8	6	19	22	17	25	2	27	15	1	5	16	26	9	21	18	3	23	7	24	4	10	11
14	15	8	16	20	22	9	6	11	3	26	1	13	27	23	12	2	19	7	10	25	5	21	4	24	18	17
14	17	7	3	13	22	5	23	10	2	1	24	26	27	15	12	19	8	21	25	9	20	6	4	18	11	16
16	11	18	4	6	20	9	25	21	8	19	12	15	27	26	24	1	2	10	23	5	22	13	3	7	17	14
17	18	24	4	21	5	25	10	7	19	2	12	23	27	13	1	26	3	11	6	9	22	20	16	8	15	14

How Was It Found?

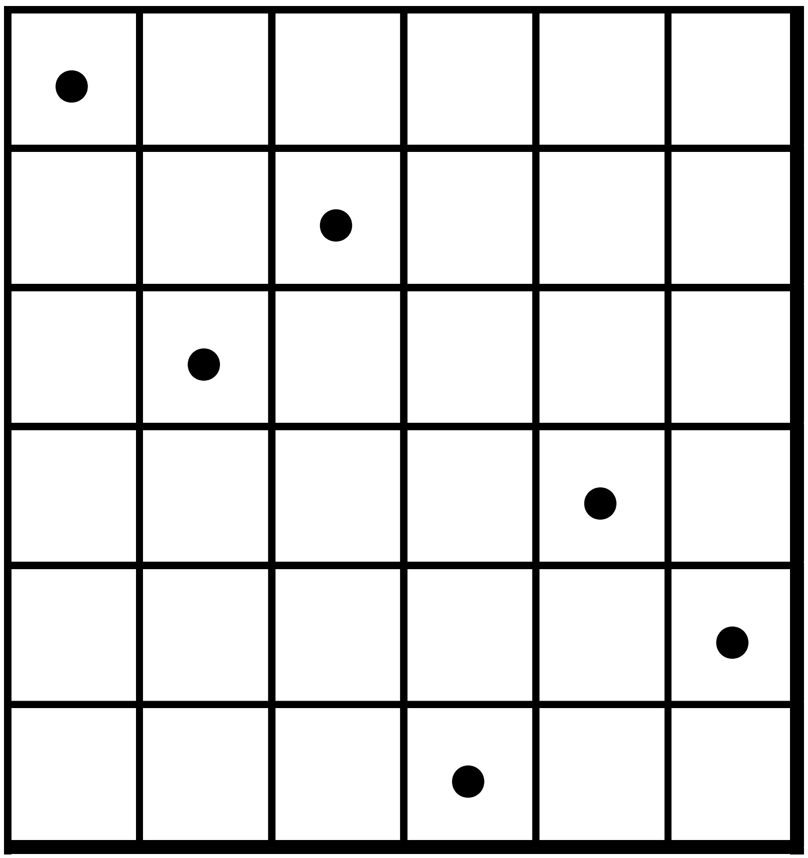
CISS 2010

- Multiple independent searches over allocated task space produces mountains of data
- Central bookkeeping methodology
  - Read all the data, every time
  - Produce counts, breakdowns as output
  - Begin with output of extended generator program
  - A change in the count indicates that a new Costas array has been found
  - Comparison of outputs with output from generator program reveals which ones are new
- High-powered CS engines for sort and other tasks allows processing “mountains of data” for data summaries as often as desired

Other Methods

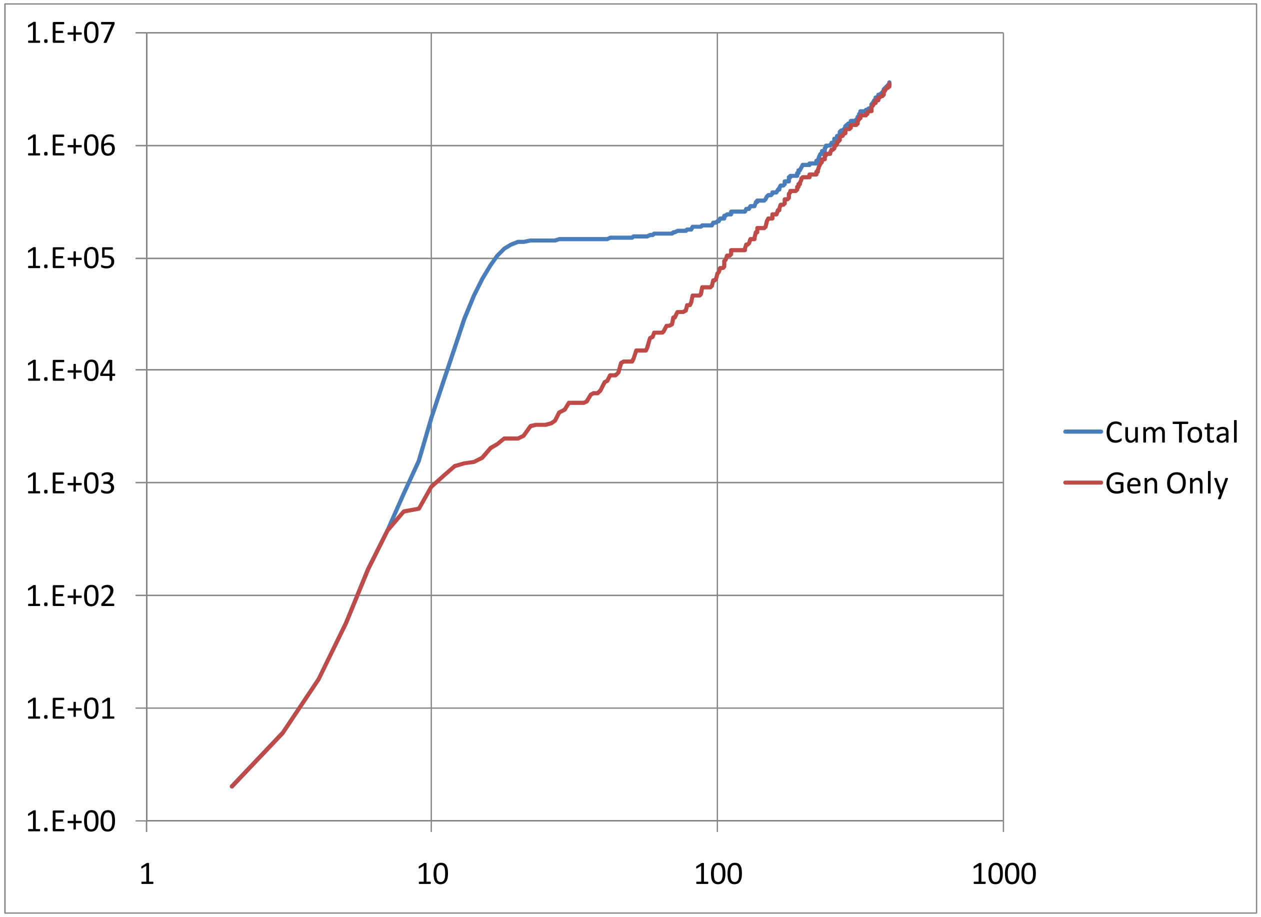
- Augmentation
  - Construct augmented matrix from two Costas arrays
  - Result must satisfy Costas condition
  - Interaction between matrices will almost always result in a violation of the Costas condition
- Interleaving
  - Two Costas arrays with orders differing by at most one
  - Construct checkerboard interleaved matrix

A Remarkable Example



Cumulative Totals versus Order

CISS 2010



Summary

- Search algorithm developed by this team
- Computational resources were unremarkable
  - Desktop computers owned and maintained by team
  - Occasional off-hours use of desktops by consenting organizations
  - Other temporary resources of opportunity as identified and exploited by team members
- Published first completed searches over orders 24, 25, 26
- Found and first reported last Costas array of order 27

Conjectures (1 of 2)

CISS 2010

- No Costas arrays above order six exist that have two empty quadrants.
- Orders 32 and 33 will be searched within the next 15 years. No Costas arrays of those orders will be found.
- The current generalizations and generators find all that exist above order 27.
- The number of Costas arrays of any given order $N>23$ does not exceed N^2. [FALSE]
- Cumulative count fits $0.19 \cdot N^{2.78}$ for large N

CISS 2010

Conjectures (2 of 2)

- The number of consecutive orders *K* for which no Costas arrays exist has no upper bound. But, for any order *L*, an order *N* exists for which there are Costas arrays, and max{|*L/N-1*|} has no lower bound as *L* increases.
- The value of Costas arrays in spectrum sharing will make them ubiquitous in communications and radar waveforms.
- The 2-D correlation properties of Costas arrays will make them fundamental to digital fingerprinting.

Acknowledgements

CISS 2010

- Wright and Monteleone participated through orders 26
- LMCO ATL Publications proofread and reviewed this work
- Keith G Erickson's contributions include the discovery of "the last Costas array"
  - Identified unique resources of opportunity
  - Defined revision of allocation to condition data to meet restrictions of this resource
  - Exploited this resource and produced results to Team Bookkeeping central processing

References (1 of 2)

CISS
2010

•John P. Costas, Project Medior – A medium-oriented approach to sonar signal processing, *HMED Technical Publication R66EMH12*, GE Syracuse NY (now Lockheed Martin Marine Systems and Sensors, Syracuse), January 1966.
•John P. Costas, "A study of a Class of Detection Waveforms Having Nearly Ideal Range-Doppler Ambiguity Properties," *Proceedings of the IEEE*, Vol. 72, No. 8, August 1984.
•S. W. Golomb and H. Taylor, "Constructions and Properties of Costas Arrays," *Proceedings of the IEEE* 72(9) pp 1143-2263, September 1984.
•Solomon Golomb and Herbert Taylor, "The T-4 and G-4 Constructions for Costas Arrays," *IEEE Transactions on Information Theory*, vol. IT-38, no. 4, July 1992, pp. 1404-1406.
•Oscar Moreno, "Survey on Costas Arrays and their Generalizations," in *Mathematical Properties of Sequences and Other Combinatorial Structures*, Jong-Seon No, Hong-Yeop Song, Tor Helleseth, and P. Vijay Kumar, Eds., Springer (Kluwer), 2003, ISBN 1-4020-7403-4.
•James K Beard, "Generating Costas Arrays to Order 200," *Conference on Information Sciences and Systems (CISS) 2006* (IEEE IT Society and Princeton University).
•Solomon Golomb and Leonard Baumert, "Backtrack Programming," *JACM*, October 1965, pp. 516-524.

References (2 of 2)

CISS 2010

•Beard, James K., Russo, Jon C., Keith G. Erickson., Michael Monteleone, and Michael Wright, "Costas Array Generation and Search Methodology," *IEEE Transactions on Aerospace and Electronic Systems*, 43, 2 (April 2007), 522-538.
•Konstantinos Drakakis, Scott Rickard, James K Beard, Rodrigo Caballero, Francesco Iorio, Gareth O'Brien and John Walsh, Results of the enumeration of Costas arrays of order 27, IEEE Transactions on Information Theory 54 10 (October 2008) pp 4684-4687.
• James K Beard, announcement of new Costas array of order 27 on personal web site, http://jameskbeard.com/jameskbeard/Costas_Arrays.html#NewCA27.
• *MacTech*, Programmer's Challenge, available on web page http://www.mactech.com/progchallenge/, click on "Costas Arrays (December 1999)."
• James K Beard, "Costas array generator polynomials in finite fields," *CISS 2008*, March 21 2008, Princeton University, Session TP 03, Paper 5; Database of Costas arrays from orders 2 to 200 on a CD-ROM given out at CISS 2006 was extended to order 400.
•Oscar Moreno, John Rameirez, Dorothy Bollman, and Edusmildo Orozco, "Faster backtracking algorithms for the generation of symmetry-invariant permutations," *Journal of Applied Mathematics* 2:6 (2002) pp. 277-287.
• James K Beard, Jon C Russo, Keith Erickson, Michael Monteleone, and Mike Wright, "Combinatoric collaboration on Costas arrays and radar applications," *Proceedings of the IEEE 2004 Radar Conference*, April 26-29 2004, ISBN 0-7803-8234-X, pp. 260-265.