Enumerative combinatorics problems for cryptographic primitives based on CA

Luca Mariot

L.Mariot@tudelft.nl

Séminaire ECO/Escape – May 17, 2021
One-dimensional **Cellular Automaton** (CA): a discrete parallel computation model composed of a finite array of $n$ cells

Example: $n = 6$, $d = 3$, $\omega = 0$, $f(s_i, s_{i+1}, s_{i+2}) = s_i \oplus s_{i+1} \oplus s_{i+2}$ (rule 150)

No Boundary CA – NBCA

Periodic Boundary CA – PBCA

Each cell updates its state $s \in \{0, 1\}$ by applying a local rule $f : \{0, 1\}^d \rightarrow \{0, 1\}$ to itself, the $\omega$ cells on its left and the $d - 1 - \omega$ cells on its right
CA-based Pseudorandom Generator (PRG) [W86]: central cell of rule 30 CA used as a stream cipher keystream

This CA-based PRNG was later shown to be vulnerable [MS91]
CA-Based Crypto History: \texttt{Keccak} $\chi$ S-box

- Local rule: $\chi(x_1, x_2, x_3) = x_1 \oplus (1 \oplus (x_2 \cdot x_3))$ (rule 210)
- Invertible for every odd size $n$ of the CA [DGV94]

- Used as a PBCA with $n = 5$ in the \texttt{Keccak} specification of SHA-3 standard [BDPV11]
Motivations

**General Research Goal**: Investigate cryptographic primitives defined by Cellular Automata

Why CA, anyway?

1. **Security from Complexity**: CA can yield very complex dynamical behaviors, depending on the local rule

2. **Efficient implementation**: Leverage CA parallelism and locality for lightweight cryptography
CA-based Primitives Considered

For each primitive, we will survey the main results and point out open problems of enumerative combinatorics.
Part 1: Orthogonal Arrays & Orthogonal Latin Squares
Orthogonal Arrays (OA)

- $(N, k, s, t)$ **Orthogonal Array**: $N \times k$ matrix over $s$ symbols s.t.
each $t$-uple occurs $\lambda = N/s^t$ times in each $N \times t$ submatrix

Example: OA $(8, 4, 2, 3)$
Each 3-bit vector $(x_1, x_2, x_3) \in \{0, 1\}^3$
appears once in the submatrix with columns 1, 3, 4

- **Crypto Applications**: threshold secret sharing schemes,
masking for side-channel attacks
Orthogonal Latin Squares (OLS)

Definition

A *Latin square* is a $n \times n$ matrix where all rows and columns are permutations of $[n] = \{1, \cdots, n\}$. Two Latin squares are *orthogonal* if their superposition yields all the pairs $(x, y) \in [n] \times [n]$.

$k$ pairwise OLS are denoted as $k$-MOLS (*Mutually Orthogonal Latin Squares*)

$k$-MOLS are equivalent $OA(n^2, k, n, 2)$
Latin Squares through Bipermutive CA (1/2)

- Bipermutive CA: denoting $\mathbb{F}_2 = \{0, 1\}$, local rule $f$ is defined as
  \[ f(x_1, \cdots, x_d) = x_1 \oplus \varphi(x_2, \cdots, x_{d-1}) \oplus x_d \]
- $\varphi : \mathbb{F}_2^{d-2} \rightarrow \mathbb{F}_2$: generating function of $f$

Lemma ([MGFL20])

A CA $F : \mathbb{F}_2^{2(d-1)} \rightarrow \mathbb{F}_2^d$ with bipermutive rule $f : \mathbb{F}_2^d \rightarrow \mathbb{F}_2$ generates a Latin square of order $N = 2^{d-1}$
Example: $CA \ F : \mathbb{F}_2^4 \rightarrow \mathbb{F}_2^2, \ f(x_1, x_2, x_3) = x_1 \oplus x_2 \oplus x_3$ (Rule 150)

Encoding: $00 \mapsto 1, 10 \mapsto 2, 01 \mapsto 3, 11 \mapsto 4$
MOCA by Linear CA

- **Bipermutive Linear rule:** \( f(x) = x_1 \oplus a_2 x_2 \oplus \cdots \oplus a_{d-1} x_{d-1} \oplus x_d \)
- **Polynomial rule:** \( P_f(X) = 1 + a_2 X + \cdots + a_{d-1} X^{d-2} + X^{d-1} \)

**Theorem ([MGFL20])**

A set of \( k \) bipermutive linear CA are \( k \)-MOCA if and only if their associated polynomials are pairwise coprime.

(a) Rule 150  
(b) Rule 90  
(c) Superposition

**Figure:** \( P_{150}(X) = 1 + X + X^2, \ P_{90}(X) = 1 + X^2 \) (coprime)
Counting linear CA-based (M)OLS

- Number of CA-based OLS pairs of diameter \( d = n + 1 \) over \( \mathbb{F}_q \):
  \[
an = q(q - 1)^3 \frac{q^{2n-2} - 1}{q^2 - 1} + (q - 1)(q - 2)
\]

- For \( \mathbb{F}_2 \), \( a_n \) coincides with OEIS sequence A002450
- Size of the biggest MOCA family of diameter \( d = n + 1 \):
  \[
  N_{n,q} = I_{n,q} + \sum_{k=1}^{\left\lfloor \frac{n}{2} \right\rfloor} I_{k,q}, \quad \text{where } I_{n,q} = \frac{1}{n} \sum_{e|n} \mu(e) \cdot q^n e
  \]

- Results published in [MGFL20]:

Generalization to nonlinear CA

- The previous results depend on the *linearity* of the local rules.
- **Open Problem**: count and enumerate OLS and MOCA families generated by *nonlinear* bipermutive rules.
- **Direction**: investigate the paths on the *de Bruijn graph*.

Example: \( f(x_1, x_2, x_3) = x_1 \oplus x_2 \oplus x_3 \) (Rule 150)

- CA input vector ⇔ path on the (overlapped) vertices.
- CA output vector ⇔ path on the edges.
Orthogonal labelings

Definition
Two bipermutative labelings $l_1, l_2$ are orthogonal for $G_{m,n}$ over $S$ if, for each pair $(x, y) \in S^n \times S^n$, there is exactly one path in $G_{m,n}$ of length $n$ labelled by $(x, y)$ under the superposed labeling $l_1 \cdot l_2$.

Example: $S = \{0, 1\}$, $m = n = 2$, $l_1 = v_1 \oplus u_2$, $l_2 = v_1 \oplus u_1 \oplus u_2$

Luca Mariot
Enumerative combinatorics problems for cryptographic primitives based on CA
Problem (Counting)

Given $m, n \in \mathbb{N}$, what is the number $N(m, n)$ of orthogonal pairs of bipermutative labelings for $G_{m,n}$?

Problem (Enumeration)

Find an algorithm that enumerates only $N(m, n)$ of orthogonal pairs of bipermutative labelings for $G_{m,n}$. 
Part 2: Hadamard matrices & Bent functions
Hadamard Matrices & Bent functions

- Hadamard Matrix: a $n \times n$ matrix with $\pm 1$ entries s.t. $H \cdot H^\top = I_n$. Example for order $n = 4$:

\[
H = \begin{pmatrix} + & + & + & + \\ + & - & + & - \\ + & + & - & - \\ + & - & - & + \end{pmatrix}
\]

- Walsh Transform of $f : \mathbb{F}_2^n \rightarrow \mathbb{F}_2$: $W(f, a) = \sum_{x \in \mathbb{F}_2^n} (-1)^{f(x) \oplus a \cdot x}$

- Bent function: "flattest" Walsh spectrum, $W_f(a) = \pm 2^{\frac{n}{2}}$

Example: $f(x_1, x_2, x_3, x_4) = x_1 x_3 + x_1 x_4 + x_2 x_4$
Constructions of Bent Functions

- **Relevance in crypto**: bent functions reach the highest possible nonlinearity
- Number of Boolean functions of $n$ variables: $2^{2n}$
- $\Rightarrow$ too huge for exhaustive search when $n > 5$

In practice, one can resort to *algebraic constructions*

- **Primary constructions**: new functions are built from scratch (e.g., Maiorana-McFarland construction)
- **Secondary constructions**: new functions are obtained from existing ones (e.g., Rothaus’s construction)
Theorem (Dillon, 1974)

Given $f : \{0, 1\}^n \rightarrow \{0, 1\}$ and $\hat{f}(x) = (-1)^{f(x)}$. Define the $2^n \times 2^n$ matrix $H$ for all $x, y \in \{0, 1\}^n$ as:

$$H(x, y) = \hat{f}(x \oplus y)$$

Then, $f$ is a bent function if and only if $H$ is a Hadamard matrix.

Example: $f(x_1, x_2) = x_1 x_2$

<table>
<thead>
<tr>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_1 x_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

$$H = \begin{pmatrix} + & + & + & - \\ + & + & - & + \\ + & - & + & + \\ - & + & + & + \end{pmatrix}$$
Theorem (Bush, 1973)

Given a set of $t$ MOLS of order $N = 2t$, and $A$ the associated $OA(t, 2t)$, define the $4t^2 \times 4t^2$ matrix $H$ as follows:

$$H(i, j) = \begin{cases} 
+1, & \text{if } i = j \\
-1, & \text{if } i \neq j \text{ and } \exists k \in \{1, \cdots, t\} \text{ s.t. the column } \\
& \text{k of } A \text{ has the same symbol in rows } i \text{ and } j \\
+1, & \text{otherwise}
\end{cases}$$

for $i, j \in \{1, \cdots, 4t^2\}$. Then, $H$ is a symmetric Hadamard matrix.

\begin{itemize}
  \item \textbf{Remark:} the Hadamard matrix constructed from a MOLS family in general does not correspond to a bent function
\end{itemize}
Question: Are the MOLS arising from linear CA suitable for constructing bent functions?

We consider only CA over $\mathbb{F}_q$ with $q = 2^l$, $l \in \mathbb{N}$

The order of the Hadamard matrix must be $4t^2 = 2^n$

We need $t$ coprime polynomials of degree $b$:

$$2^{lb} = 2t \iff lb = 1 + \log_2 t$$

Since both $l$ and $b$ are integers, $t = 2^w$ for $w \in \mathbb{N}$
Theorem

Let $H$ be the Hadamard matrix of order $2^{2(w+1)}$ defined by the $t$ LBCA $F_1, \cdots, F_t : \mathbb{F}_q^{2b} \to \mathbb{F}_q^b$, and define $f : \mathbb{F}_2^n \to \mathbb{F}_2$, $n = 2(w + 1)$ as:

$$f(x) = \begin{cases} 
0, & \text{if } x = 0 \\
1, & \text{if } x \neq 0 \text{ and } \exists k \in \{1, \ldots, t\} \text{ s.t. } F_k(x) = 0 \\
0, & \text{otherwise}
\end{cases}$$

Then, it holds that:

$$H(x, y) = \hat{f}(x \oplus y)$$

and thus $f$ is a bent function

Remark: The linearity of the CA is crucial to grant this result.
Figure 3: Example of bent function of $n = 4$ variables generated by the $t = 2$ MOLS of order $2t = 4$ defined by the LBCA with rule 90 and 150, respectively. The two Latin squares are represented on the left in the OA form. The first row and the first column of the Hadamard matrix $H$ coincide with the polarity truth table of the function.
Combinatorial questions addressed in [GMP20]:

- **Existence:** for even $n$, a large enough family of coprime polynomials exists iff the degree is either 1 or 2

- **Counting:** how many families of this kind exist (= number of CA-based bent functions)

Currently submitted to Designs, Codes and Cryptography
Other Remarkable findings [GMP20]:

- Bent functions from this construction belong to the *Partial Spread class* \( \mathcal{PS}^- \)
- For degree 1, the resulting class of bent functions coincides with the *Desarguesian spread*

Open problems:

- Investigate the case of degree 2, to see if our functions are equivalent to other known classes
- Is our construction generalizable to *nonlinear CA*?
Part 3: S-Boxes & Vectorial Boolean functions
Block Ciphers: Substitution-Permutation Network

- $S_i : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ are S-boxes, or vectorial Boolean functions, providing confusion
- The S-boxes must:
  - be bijective
  - have high nonlinearity
  - have low differential uniformity

- **Research line**: use periodic CA to design S-boxes
- In [MPLD19], *Genetic Programming* is used to design CA-based S-boxes, with $n = d$ and $4 \leq n \leq 8$
An (infinite) CA is reversible (RCA) if its global rule $F : \{0, 1\}^n \rightarrow \{0, 1\}^n$ is bijective and $F^{-1}$ is also a CA [H69].

For finite PBCA, there exist globally invertible rules that give RCA only for certain lengths $n$. Example: $n = 3$, $d = 3$, $\omega = 0$, $f(x_i, x_{i+1}, x_{i+2}) = x_i \oplus x_{i+1} \cdot x_{i+2} \oplus x_{i+2}$.

Local rules resulting in RCA for every size $n$ of the array are also called locally invertible [DGV94].
The local rule $f$ of marker CA is defined as follows:

$$f(x_{i-\omega} \cdots x_{i-1}x_i x_{i+1} \cdots x_{i-\omega+d-1}) = x_i \oplus g(x_{i-\omega} \cdots x_{i-1}x_{i+1} \cdots x_{i-\omega+d-1})$$

Equivalently: the support of $g$ defines the markers for which the central cell flips its state.

Example: $d = 3$, $\omega = 0$, $f(x_i, x_{i+1}, x_{i+2}) = x_i \oplus x_{i+1} \cdot x_{i+2} \oplus x_{i+2}$

<table>
<thead>
<tr>
<th>$x_{i+1}$</th>
<th>$x_{i+2}$</th>
<th>$g(x_{i+1}, x_{i+2})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Marker: 01 $\Rightarrow$ $\star$01 Flipping landscape
Conserved Landscape Marker CA

- **Conserved Landscape**: each cell in a flipping landscape must be in the same landscape after applying the CA global rule.

Example: \( d = 4, \omega = 1, \text{Landscape: } 0 \star 10 \)

Landscape tabulation

Example of orbit of period 2

A landscape is conserved if it is *incompatible* with all its *neighborhood landscapes*.
Idea: Use *Evolutionary Algorithms* to investigate the class of conserved landscape CA [MPJL20]

**First Objective:** minimize the number of neighborhood landscapes that are compatible with each flipping landscape

\[
\begin{align*}
x_{i-1} & \quad \quad - \quad \quad - \quad \quad 0 \\
x_i & \quad \quad 1 \quad \quad \star \quad \quad 0 \quad \quad 0 \\
x_{i+1} & \quad \quad - \quad \quad \star \quad \quad 0 \quad \quad - \\
x_{i+2} & \quad \quad 0 \quad \quad \star \quad \quad - \quad \quad -
\end{align*}
\]

COMPATIBLE!

COMPATIBLE!

**Second objective:** maximize the Hamming weight of the generating function (related to S-box nonlinearity)

\[
g(x) = 0 \quad 0 \quad 1 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad \Rightarrow \quad \text{weight: 2}
\]
Experimental Findings

- **Main finding**: The more a marker CA rule is reversible, the lower its Hamming weight must be.

- **Open problem 1**: determine an *upper bound* on the Hamming weight for conserved landscape rules.
Open problem 2: count the number of such rules

Compatible landscapes induce a partial order $\leq_C$:

$$L \leq_C M \iff l_i = m_i \text{ or } l_i \in \{0, 1\} \text{ and } m_i = -, \ 0 \leq i \leq d - 1$$

A conserved landscape rule is an antichain on this poset

Counting antichains in general finite posets is $\#\mathcal{P}$-complete!

Is there an efficient way to count them on this poset?
Wrap-up & Conclusions
Conclusions

We surveyed three CA-based cryptographic primitives:

- Orthogonal Arrays and Mutually Orthogonal Latin Squares,
- Hadamard Matrices and Bent Functions,
- S-boxes defined by reversible CA,

showing several open combinatorial problems related to them.

Several **other directions** exist on this research line, such as:

- *Latin Hypercubes* defined by (linear) CA [GM20]
- *Asynchrony Immunity* as a relevant cryptographic property for CA-based S-boxes [MMD20]
Thank you!
References


