An Evolutionary View on Reversible Shift-invariant Transformations

Luca Mariot, Stjepan Picek, Domagoj Jakobovic, Alberto Leporati

l.mariot@tudelft.nl

EuroGP 2020, 15–17 April 2020

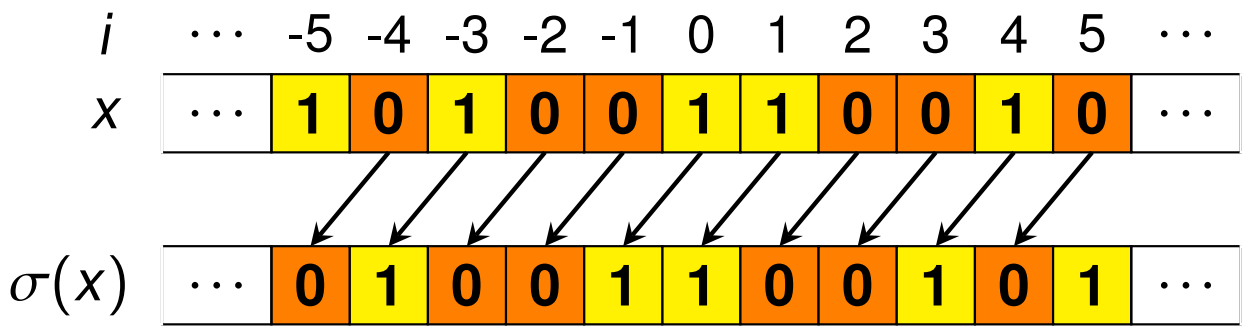
Outline

Outline

Shift-invariant Transformations

- Let $x \in \{0,1\}^{\mathbb{Z}}$ be a *bi-infinite binary string*
- The *shift operator* $\sigma : \{0,1\}^{\mathbb{Z}} \to \{0,1\}^{\mathbb{Z}}$ is defined as:

$$\sigma(x)_i = x_{i+1} \ , \ \text{for all } x \in \{0,1\}^{\mathbb{Z}},\ i \in \mathbb{Z}$$



- A mapping $F : \{0,1\}^{\mathbb{Z}} \to \{0,1\}^{\mathbb{Z}}$ is *shift-invariant* if it commutes with the shift operator, that is

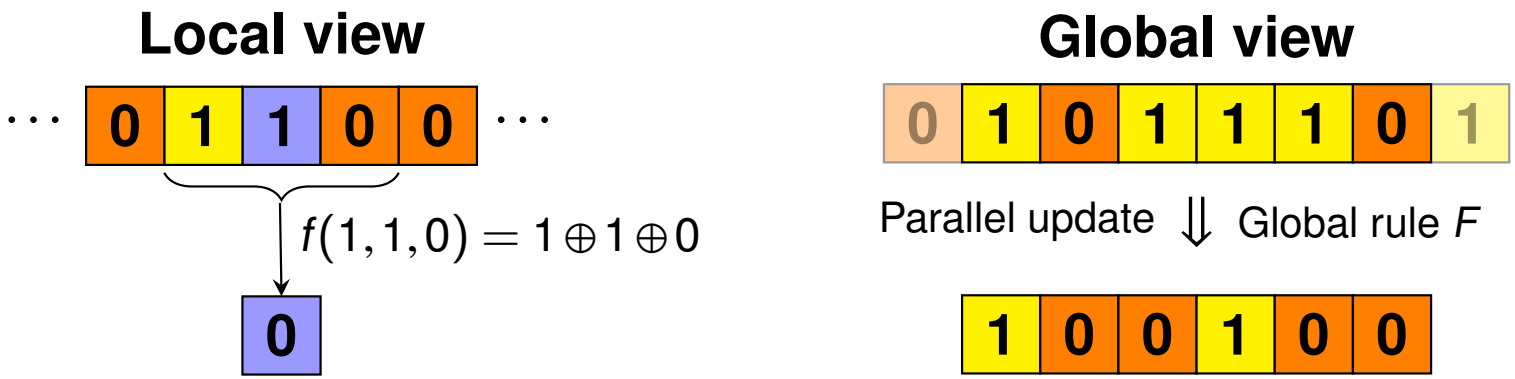
$$F(\sigma(x)) = \sigma(F(x)) \ , \ \text{for all } x \in \{0,1\}^{\mathbb{Z}}$$

Cellular Automata (CA)

Definition (Periodic Boolean Cellular Automata – CA)

A finite binary array of n cells, where each cell x_i updates its state by applying a *local rule* $f : \{0,1\}^d \to \{0,1\}$ to the *neighborhood* $\{x_{i-\omega}, \cdots, x_i, \cdots, x_{i-\omega+d-1}\}$ with *periodic boundary conditions*

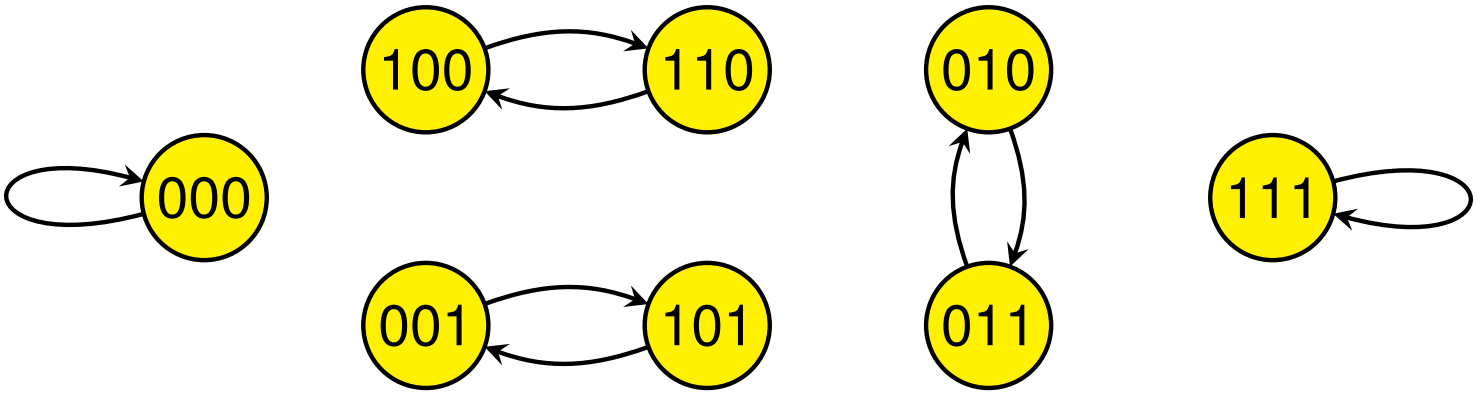
Example: $n = 6$, $d = 3$, $\omega = 1$, $f(x_{i-1}, x_i, x_{i+1}) = x_{i-1} \oplus x_i \oplus x_{i+1}$



Reversible CA

- A CA is *reversible* (RCA) if its global rule $F : \{0,1\}^n \to \{0,1\}^n$ is *bijective* and the inverse map F^{-1} is also a CA [Hedlund69]
- Interesting for applications in *reversible computing* and *cryptography* [Mariot19]

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* [Daemen95]

Marker CA

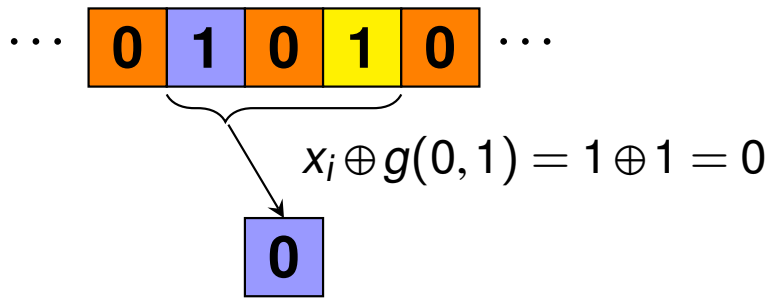
- The local rule f of marker CA is defined as follows:

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

- Equivalently: the *support* of g defines the *markers* for which the central cell *flips* its state

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

x_{i+1}	x_{i+2}	$g(x_{i+1}, x_{i+2})$
0	0	0
1	0	0
0	**1**	**1**
1	1	0

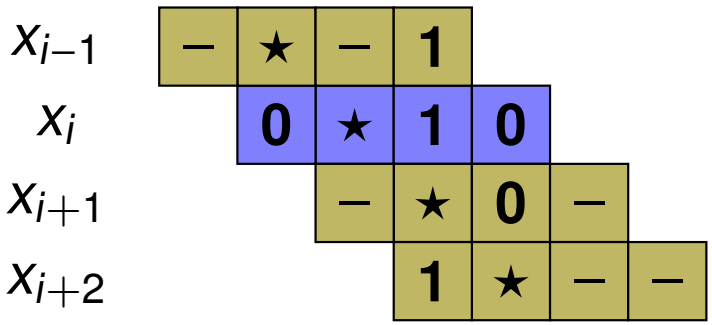


Marker: $01 \Rightarrow \star 01$ **Flipping landscape**

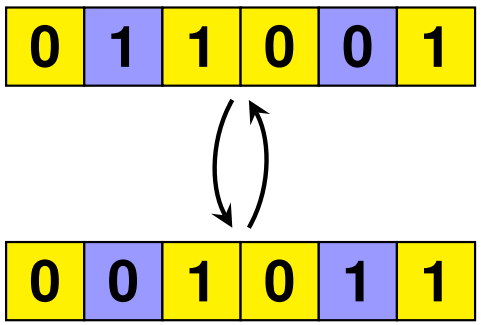
Conserved Landscape Marker CA

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

Example: $d = 4$, $\omega = 1$, Landscape: $0 \star 10$



Landscape tabulation



Example of orbit of period 2

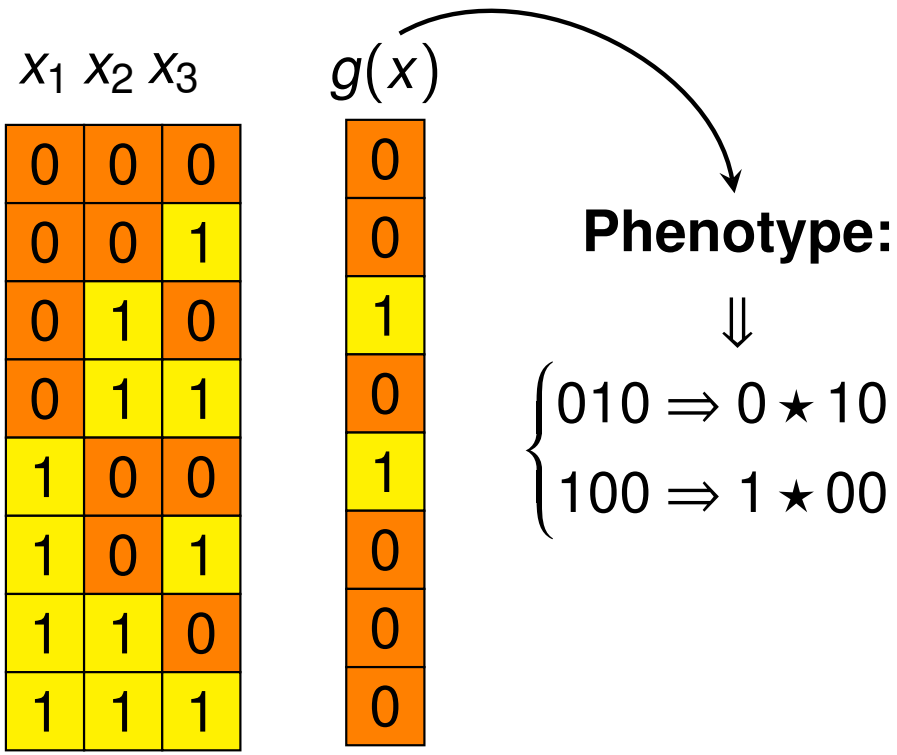
- A landscape is conserved if it is *incompatible* with all its *neighborhood landscapes* [Toffoli90]
- **Question:** How to turn the search of conserved landscape marker CA into an optimization problem?

Outline

Genotype Encoding – GA

- *Phenotype*: the set of markers in the generating function g
- *GA Genotype*: Bitstring $g(x)$ corresponding to the output column of the *truth table* of g

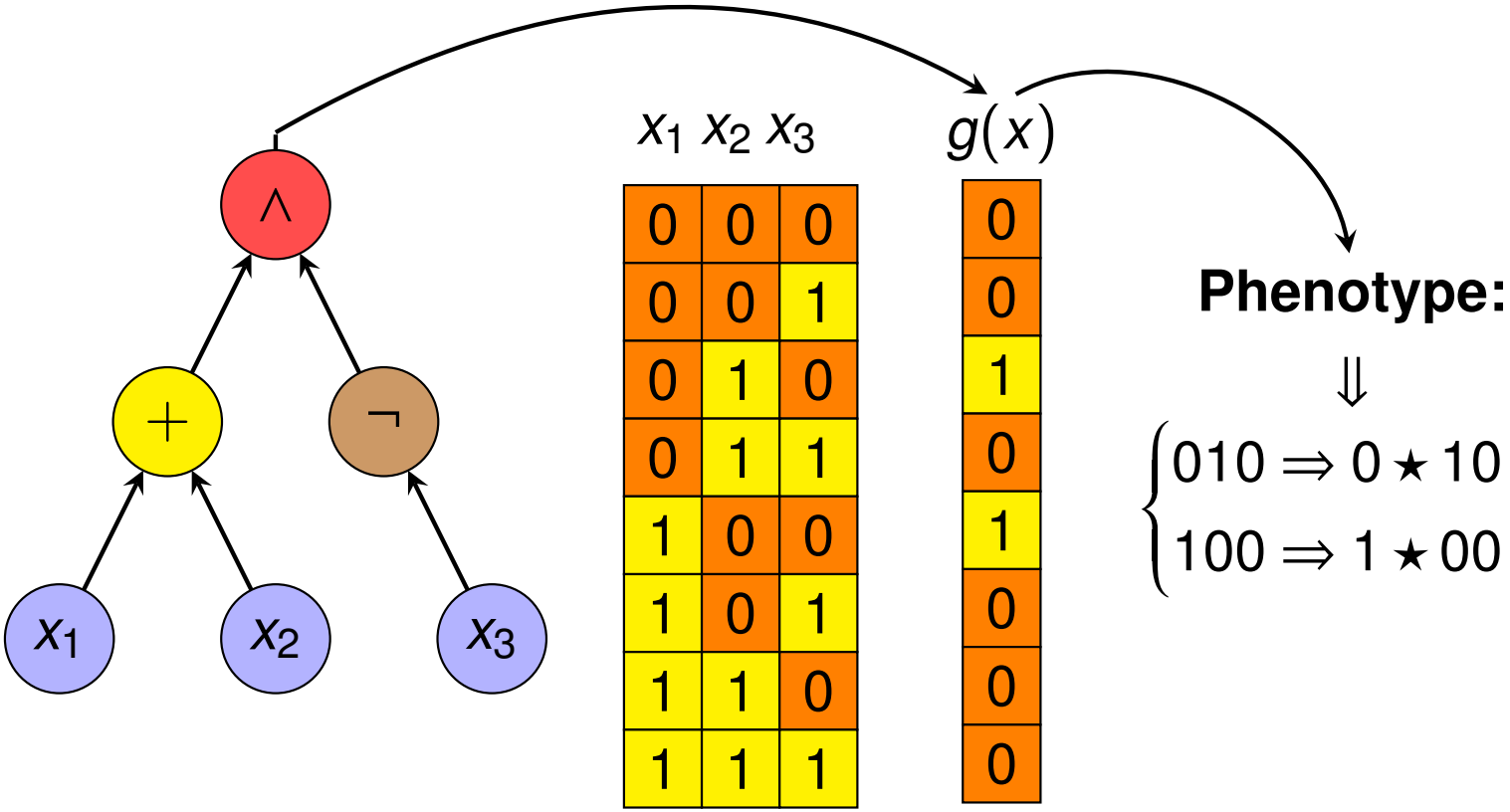
Example: $d = 4$, $\omega = 1$, $g : \{0,1\}^3 \to \{0,1\}$



Genotype Encoding – GP

- *GP Genotype*: Boolean tree
- The truth table $g(x)$ is synthesized from the tree [Mariot18]

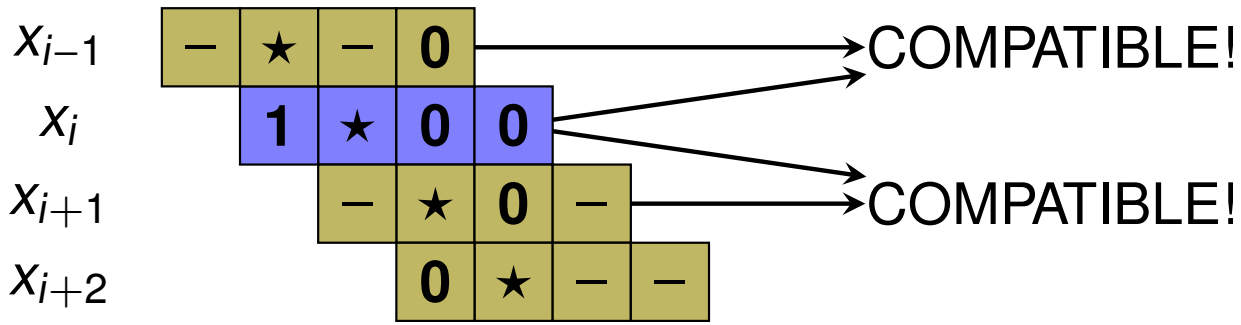
Example: $d = 4$, $\omega = 1$, $g : \{0,1\}^3 \to \{0,1\}$



First Fitness Function

- **Objective:** *minimize* the number of neighborhood landscapes that are compatible with each landscape in g

Example: $d = 4$, $\omega = 1$, Landscape: $1 \star 00$



- **Fitness function:** Loop over all landscapes in the support of g and count the compatible neighborhood landscapes

$$fit_1(g) = \sum_{i,t \in [k], j \in [d-1]_\omega} comp(M_{i,j}, L_t)$$

Second Fitness Function

- **Objective:** *maximize* the *Hamming weight* of g
- This criterion is relevant in cryptography: the higher the Hamming weight of g, the higher the nonlinearity of the CA

Example: $d = 4$, $\omega = 1$, $g : \{0,1\}^3 \rightarrow \{0,1\}$

$g(x) =$ | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 |

$\Downarrow$

Hamming weight: 2

- **Fitness function:** Count the number of 1s in $g(x)$

$$fit_2(g) = |supp(g(x))|$$

Exhaustive Search up to $d = 6$

- No. of generating functions of $d-1$ variables: $\#\mathcal{P}(d) = 2^{2^{d-1}}$
- We performed an exhaustive search of all conserved landscape rules up to $d = 6$, with $\omega = \left\lfloor \frac{d-1}{2} \right\rfloor$

d	2^{d-1}	$\#\mathcal{P}(d)$	#REV	Weights
3	4	16	0	–
4	8	256	1	1
5	16	65536	10	1,2
6	32	$4.3 \cdot 10^9$	46	1,2,3

- The number of conserved landscape rules is *really small* wrt the number of generating functions
- The possible Hamming weights are *really low* wrt to the length of the truth table of g

Research Questions

- **RQ1:** Given the limited number conserved landscape rules, is it difficult for GA and GP to find them?
- **RQ2:** Do there exist conserved landscapes rules of a larger diameter and with higher Hamming weight?
- **RQ3:** Is there a trade-off between the reversibility of a marker CA rule and its Hamming weight?

Outline

Experimental settings

Common Parameters:

- Problem instances: diameters $7 \leq d \leq 13$
- Termination condition: 500 000 fitness evaluations
- Each experiment is repeated over 30 independent runs
- Selection operator: steady-state with 3-tournament operator

GA Parameters:

- Population size: 30 individuals
- Mutation probability: $p_m = 0.2$

GP Parameters:

- Boolean operators: AND, OR, XOR, XNOR, NOT, IF
- Population size: 500 individuals
- Mutation probability: $p_m = 0.5$

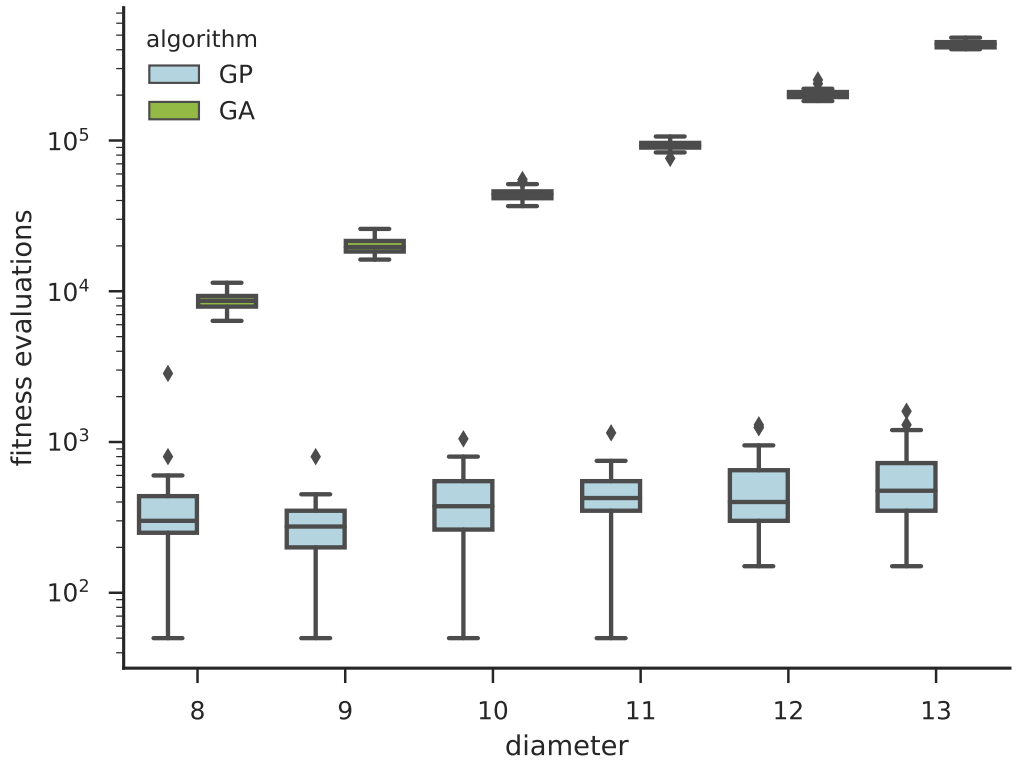
Optimization Approaches

We employed three different optimization approaches to investigate the research questions:

- *Single-objective Optimization* only of the reversibility property with GA and GP, by minimizing fit_1
- *Multi-objective Optimization* with GP, by minimizing fit_1 and maximizing the Hamming weight fit_2
- *Lexicographic Optimization* with GP, by first minimizing fit_1 and then maximizing fit_2 while retaining reversibility

Single-Objective GA and GP

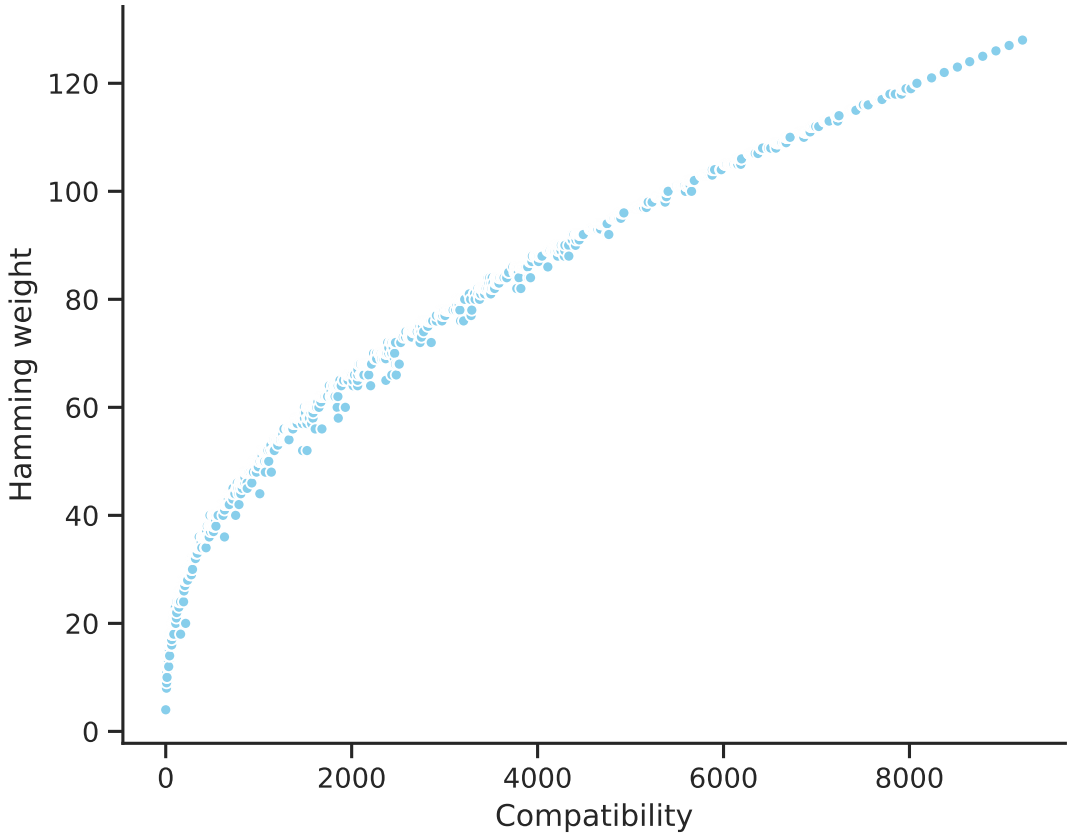
- **Main finding**: *both GA and GP converge to an optimal solution over all experimental runs*



- However, the number of fitness evaluations required by GA scales exponentially with the number of variables

Multi-Objective GP

- We used Multi-objective GP to approximate the Pareto fronts of reversibility vs. Hamming weight



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

Lexicographic GP Optimizer

- We compared the Hamming weights and distinct solutions achieved by lexicographic GP with MOGP, SOGP and SOGA

	SOGA			SOGP			MOGP			LEXGP		
d	UHW	MHW	USol	UHW	MHW	USol	UHW	MHW	USol	UHW	MHW	USol
8	5	6	30	4	8	27	4	10	24	**5**	**10**	**47**
9	6	7	30	4	16	29	2	20	22	**8**	**20**	**60**
10	**7**	11	30	3	16	30	4	**32**	48	6	28	**65**
11	**9**	15	30	3	32	29	6	56	40	6	**56**	**64**
12	**11**	23	30	4	64	30	4	72	29	7	**80**	**71**
13	**12**	29	30	2	64	29	4	128	50	7	**160**	**73**

- **Main finding**: *Lexicographic GP achieves the best trade-off among number of distinct optimal solutions, highest and distinct Hamming weights achieved*

Outline

Conclusions

Summing up our findings:

- **RQ1:** Despite the small size of the optimal solution set, GA and GP always converge to conserved landscape rules (although GP is far more efficient than GA)
- **RQ2:** Conserved landscape rules seem to be characterized by low Hamming weights with respect to their size (thus, they are not interesting for cryptographic purposes)
- **RQ3:** The Pareto fronts suggest that the closer a rule is of the conserved landscape type, the lower its Hamming weight is

Future Directions

Several directions open for further research:

- Investigate the performance gap between GA and GP, by performing fitness landscape analysis
- Consider marker CA rules with *partially overlapping* landscapes, which may be more interesting for cryptography
- Find a theoretical explanation for the trade-off between reversibility and Hamming weight observed on the Pareto fronts.

References

- [Daemen95] Daemen, J.: Cipher and hash function design strategies based on linear and differential cryptanalysis. PhD thesis, Doctoral Dissertation, March 1995, KU Leuven (1995)
- [Hedlund69] Hedlund, G.A.: Endomorphisms and Automorphisms of the Shift Dynamical Systems. Mathematical Systems Theory 3(4): 320–375 (1969)
- [Mariot19] Mariot, L., Picek, S., Leporati, A., Jakobovic, D.: Cellular automata based S-boxes. Cryptography and Communications 11(1): 41–62 (2019)
- [Mariot18] Mariot, L., Picek, S., Jakobovic, D., Leporati, A.: Evolutionary Search of Binary Orthogonal Arrays. In: Auger, A., Fonseca, C.M., Lourenço, N., Machado, P., Paquete, L., Whitley, D. (eds.): PPSN 2018 (I). LNCS vol. 11101, pp. 121–133. Springer (2018)
- [Toffoli90] Toffoli, T., Margolus, N.H.: Invertible cellular automata: a review. Physica D: Nonlinear Phenomena 45(1-3): 229–253 (1990)