Combinatorial approach to counting equivalence classes in 1D and 2D cellular automata

IZTOK JERAS[1]*

Faculty of Computer and Information Science, University of Ljubljana, Slovenia

Received 17 December 2004; In final form 1 April 2005

There is much more theoretical research on 1D cellular automata compared to 2D. While not the only reason one of the arguments given is that the number of available rules for a common 2D neighborhood is too large for a comprehensive analysis. This issue can be mitigated by focusing on the smallest 2D neighborhoods (trid and quad) and by grouping rules into equivalence classes. This article gives an algebraic combinatorial solution to counting equivalence classes.

Key words: cellular automata, combinatorics, equivalence classes, congruence operations

1 INTRODUCTION

The focus of theorethical research is usually the simplest possible version of a system. For 1D CA elementary rules are the simplest which exsibit not trivial behavior. Rule equivalence has been used by many authors to organize 256 elementary CA rules into ??? equivalence classes. Representative rules from each class were then observed and described. This approach found the most interesting rules which were then further analyzed. Rule 110 is a good example.

The same approach was never applied to 2D CA. For example most studied 2D rule is Game of Life. It is based on a Moore neighborhood is binary cells.

* email: `iztok.jeras@rattus.info`

There are $2^{2^9} = 2^512$ possible rules, too many for a comprehensive ananlysis of the entire rule space. The von Neumann neighborhood with $2^{2^5} = 2^32$ rules is still impractical, and there is no rule with this neighborhood on a binary lattice, whish would receive as much attention as Game of Life. While this two neighborhoods were the focus of 2D CA research till recently, there are two simpler neighborhoods named quad and trid which were popularized by Tommaso Toffoli only in 2008 using the QUAD prize competition. The aim of the QUAD prize was to find a computation-universal 2-state cellular automaton on a 2-dimensional, 22-neighborhood. Edward Powley proved computational universality on an even smaller neighborhood, the trid.

The focus of my CA research are preimages. On 1D CA de Bruijn diagrams are the main tool used for this reaseadch. The size of this diagrams depends on the size of the rue space. While once the algorithms for calculating preimages are known, the ability to compute them is only limited by computer memory, the research of such algorithms usually requires the ability of the researcher to model such problems with mathematical notations like graphs and matrices. Only with 2D neighborhoods as small as quad and trid such representations become small enough for theoretical analysis with the help of visual representations.

The purpose of this article is to calculate the number of equivalence classes for a few small neighborhoods on a binary lattice (2D and 3D), which I hope will help further theoretical research into 2D CA.

??? Most research on CA was recently done by Hidenosuke Nishio ???.

2 DEFINITION OF EQUIVALENCE

Two rules are equivalent, if a transformation for the CA state exists, such that each CA current-next state pair for the current rule gets transformed into a current-next state pair for the other rule.

State transition graphs for equivalend rools are of the same shape, and a trasformation exists which maps each state in one graph to a state at the same position in the graph for the other rule.

There are generalizations to this definition. For example a rule and it second order rule can be thought as equivalent. (rule performing the same global transition in one step as the original rule achieves in two steps.

2.1 1D problem

Equivalence groups for elementary CA are long known. What is missing is a general algebraic solution.

Symetries
There is a set of known symetries or congruence operations on a given CA state. For 1D CA this are reflection and permutation.

Counting neighborhoods
The set of all distinct neighborhood values for a given neighborhood size and number of cell states is:

$$\mathbb{M} = \{n :\}$$
$$|\mathbb{M}| = k^m$$

Neighborhoods can be organized into subsets depending on their invariance to congruence operations.

There are neighborhoods which are invariant to reflection:

$$\mathbb{M}_{\mathrm{ref}} = \{\forall n \in \mathbb{M} : \phi_{\mathrm{ref}}(n) = n\}$$
$$|\mathbb{M}_{\mathrm{ref}}| = \begin{cases} k^{m/2+1} & \text{if } m \text{ is odd} \\ k^{m/2} & \text{if } m \text{ is even} \end{cases}$$

There are no neighborhoods which are invariant to permutation:

$$\mathbb{M}_{\mathrm{per}} = \{\forall n \in \mathbb{M} : \phi_{\mathrm{per}}(n) = n\} = \varnothing$$
$$|\mathbb{M}_{\mathrm{per}}| = 0$$

There are neighborhoods which are invariant after both reflection and permutation have been applied:

$$\mathbb{M}_{\mathrm{ref,per}} = \{\forall n \in \mathbb{M} : \phi_{\mathrm{per}}(\phi_{\mathrm{ref}}(n)) = n\}$$
$$|\mathbb{M}_{\mathrm{ref,per}}| = \begin{cases} 0 & \text{if } m \text{ is odd} \\ k^{m/2} & \text{if } m \text{ is even} \end{cases}$$

2.2 1D problem

Counting rules
Symetries
There is a set of known symetries or congruence operations on a given CA state. For 1D CA this are reflection and permutation.

REFERENCES

[1] Hidenosuke Nishio. (2012). A generalization of automorphism classification of cellular automata. *J. Cellular Automata*, 7(2):167–177.

[2] Hidenosuke Nishio and Thomas Worsch. (2010). Changing the neighborhood of cellular automata: Local structure, equivalence and isomorphism. *Journal of Cellular Automata*, (Vol. 5):pp. 227240.