

DELFT UNIVERSITY OF TECHNOLOGY

Bi-threshold Gates for Mechanical Logic in Intelligent Metamaterials

Authors:

Eoinlee Bley (5216737)

Supervisors:

Dr. Davood Farhadi Machekposhti
Malte ten Wolde

in partial fulfillment of the requirements for the degree of

Master of Science
in Mechanical Engineering

September 20, 2023



Abstract

Contents

0.1	Elementary Cellular Automata Formalism	2
0.2	Wolfram Numbering Scheme for ECA	2
1	Introduction	3
2	Results & Discussion	3
A	Supplementary Material	7

0.1 Elementary Cellular Automata Formalism

1. State Space: $S = \{0, 1\}$
2. Neighborhood Configuration: N
 $N = (N_{-1}, N_0, N_1)$ where $N_{-1}, N_0, N_1 \in S$
3. Rule Function: $f : S^3 \rightarrow S$
4. Rule Set: R
5. Cube Domain: $D \subset \mathbb{R}^3$
Each vertex directly corresponds to a neighborhood configuration N , and its state is
6. Separating Planes: P
Defined by a single normal vector \mathbf{n} and different offsets $\{d_1, d_2, \dots, d_n\}$.
7. Domain Classification Function: $\Delta : D \rightarrow \{0, 1, 2, 3\}$
 $\Delta(x) = \sum_{i=1}^n H(n_x \cdot x_x + n_y \cdot x_y + n_z \cdot x_z - d_i)$
 $H(z) = \begin{cases} 0 & \text{if } z < 0 \\ 1 & \text{if } z \geq 0 \end{cases}$

0.2 Wolfram Numbering Scheme for ECA

In the Wolfram numbering scheme for Elementary Cellular Automata (ECA), the rule set R can be uniquely identified by a single integer, which is the binary representation of the output states for all possible neighborhood configurations. For Rule 110, the binary representation is formed by considering all 8 possible 3-cell neighborhood configurations, starting from 111 down to 000.

For example, in Rule 110, the corresponding output states for these configurations are 01101110. Here's how it maps:

Neighborhood Configuration	Output State	Binary Position (b)
111	0	b_7
110	1	b_6
101	1	b_5
100	0	b_4
011	1	b_3
010	1	b_2
001	1	b_1
000	0	b_0

So, the Wolfram number for Rule 110 is obtained by reading the output states from b_7 to b_0 as a binary number: $01101110_2 = 110_{10}$.

1 Introduction

2 Results & Discussion

Elementary Cellular Automata Formalism

Cellular automata (CA) are grid-based computational models where each cell evolves over time according to a rule set R . In Elementary Cellular Automata (ECA), the domain is one-dimensional and the state space is binary, $S = \{0, 1\}$. Each cell's future state is determined by its current state and those of its immediate neighbours.

Mathematically, for cell i at time t , the next state u_i^{t+1} is governed by a rule function $f : S^3 \rightarrow S$:

$$u_i^{t+1} = f(u_{i-1}^t, u_i^t, u_{i+1}^t)$$

With a binary state and 3-cell neighbourhood, there are $2^8 = 256$ unique ECA rules. These are indexed from 0 to 255, following Wolfram's convention, detailed in subsection 0.2.

For example, Rule 110 is defined explicitly as:

$$f_{110} : \begin{array}{l} (0, 0, 0) \rightarrow 0, (0, 0, 1) \rightarrow 1, (0, 1, 0) \rightarrow 1, (0, 1, 1) \rightarrow 1, \\ (1, 0, 0) \rightarrow 0, (1, 0, 1) \rightarrow 1, (1, 1, 0) \rightarrow 1, (1, 1, 1) \rightarrow 0 \end{array}$$

Consult Figure 1A for a graphical depiction of Rule 110's eight possible neighbourhoods and their respective output states. Also shown is the time evolution of the rule, starting from a single 'on' cell at the left edge of the domain.

Hypercube Representation of Cellular Automata Rules

Consider a cube in \mathbb{R}^3 as the domain D , with each vertex representing a unique neighbourhood configuration $N = (N_{-1}, N_0, N_1)$, where $N_{-1}, N_0, N_1 \in \{0, 1\}$. The cube's vertices are colored based on a rule function $f : \{0, 1\}^3 \rightarrow \{0, 1\}$, thereby geometrically realizing the Boolean truth table of an Elementary Cellular Automaton (ECA).

To introduce the concept of linear separability, consider separating parallel hyperplanes P defined by a normal vector \mathbf{n} and offsets $\{d_1, d_2, \dots, d_n\}$. These hyperplanes partition D into regions where the vertices share the same output state as determined by f .

We define a domain classification function $\Delta : D \rightarrow \{0, 1, \dots, n\}$ as $\Delta(x) = \sum_{i=1}^n H(\mathbf{n} \cdot \mathbf{x} - d_i)$, where $H(z)$ is the Heaviside step function. Each value of $\Delta(x)$ corresponds to one of the $n + 1$ regions formed by P .

Finally, a mapping function $M : \{0, 1, \dots, n\} \rightarrow \{0, 1\}$ translates the region identifier $\Delta(x)$ into the 'on' or 'off' state for each neighbourhood configuration. Refer to Figure 1B for a graphical representation of the cube and separating planes for Rule 110.

As shown in Appendix A, most ECA rules are bi-planarly separable, i.e. they can be represented by two parallel planes.

Bi-threshold Gates

The result of this formulation of 3-input Boolean functions as linearly separable regions in a cube is the observation that most 3-input Boolean functions can be represented by a pair of parallel planes. This means we can translate a complicated Boolean algebraic expression composed of several AND, OR, XNOR, etc., gates into a single bi-threshold perceptron gate.

The bi-threshold perceptron for this ECA context can be formally represented as:

$$f(N) = \begin{cases} 0 & \text{if } \sum_{i=-1}^1 w_i \cdot N_i > T_1 \\ 1 & \text{if } T_2 \leq \sum_{i=-1}^1 w_i \cdot N_i \leq T_1 \\ 0 & \text{if } \sum_{i=-1}^1 w_i \cdot N_i < T_2 \end{cases}$$

Concept Mechanism

In light of the mathematical formalism presented, we introduce a conceptual mechanical metamaterial designed to embody the logic and behavior of Elementary Cellular Automata (ECAs). This metamaterial is constructed from an array of interconnected unit cells, each serving as a mechanical analog to the bi-threshold perceptron.

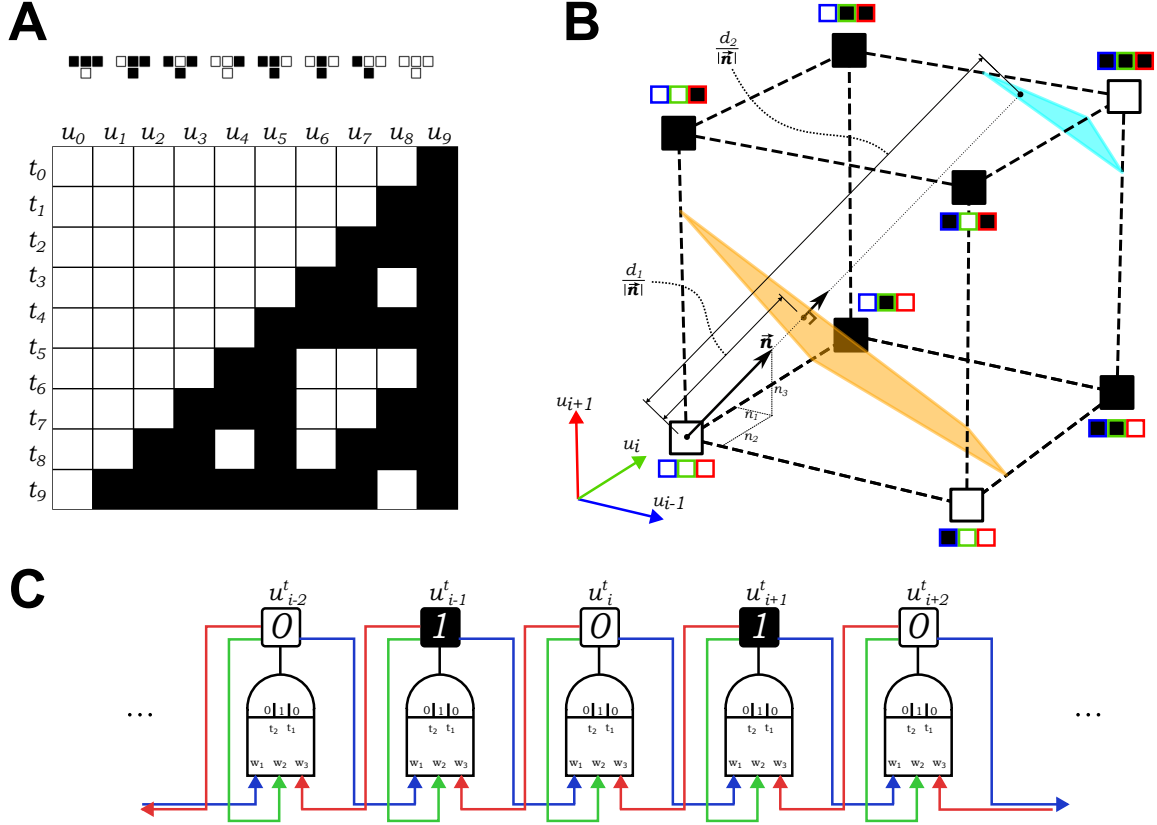


Figure 1: A. The transition rule and time evolution of the Rule 110 cellular automata. B. Cube representation Rule 110 with separating planes defined by normal vector \vec{n} and offset constants d_1 and d_2 . The red, green and blue colouring corresponds to left neighbour, middle, and right neighbour cells of the neighbourhood respectively. C. Bi-threshold gate representation of an ECA architecture.

The core of each unit cell is a tristable element, functioning as the decision-making component. The tristable element has three stable states, akin to the three regions separated by the two parallel planes in the cube of our geometric representation. This element is responsible for holding the output state of the cell, dictated by the weighted sum of its inputs.

Each unit cell is interconnected via coupling springs, which transmit mechanical signals between adjacent cells. The stiffness values of the coupling springs k_i act as the weights w_i in the bi-threshold perceptron equation. These values determine the force interactions and state transitions between adjacent unit cells. The tristable elements have multiple stable states, analogous to the regions separated by planes in the cube of our geometric ECA representation.

An input clock signal introduces a temporal dimension to the mechanical system, enabling dynamic state evolution similar to time-stepping in ECAs. This clock signal sets the computational cycle and synchronizes the unit cells.

Thus, the mechanical properties of the springs and tristable elements correspond directly to the mathematical constructs of the bi-threshold perceptron, providing a means to implement ECA rules in a mechanical system. The specific embodiment of this concept is detailed in the following section.

Unit cell design

The unit cell is designed to be planar and monolithic for scalability, operate under a single shared clock signal for synchronization, transmit forces between adjacent cells, hold state in the absence of input, and transition states according to neighboring conditions. Figure 2 depicts the unit cell and its key components: the tristable element in teal, the bistable element in purple, the signal transmission element in orange, and the input bifurcation

element in dark blue. Coupling springs, coloured in red, green, and blue, link the tristable element out-of-plane to the bifurcation elements of adjacent unit cells. The configuration of each unit cell is fully defined by two displacements: d^t in the \hat{e}_1 direction for the tristable element and d^b in the \hat{e}_1 direction for the bifurcation element. The input bifurcation element is so named because

The bistable element acts as a mechanical binary memory element for the unit cell, while the tristable element's two snapthrough force thresholds act as decision boundaries corresponding to the thresholds of the threshold gate, or the separating planes of the boolean function. The signal transmission and bifurcation elements facilitate the temporal clocking and informational interconnection of the unit cells.

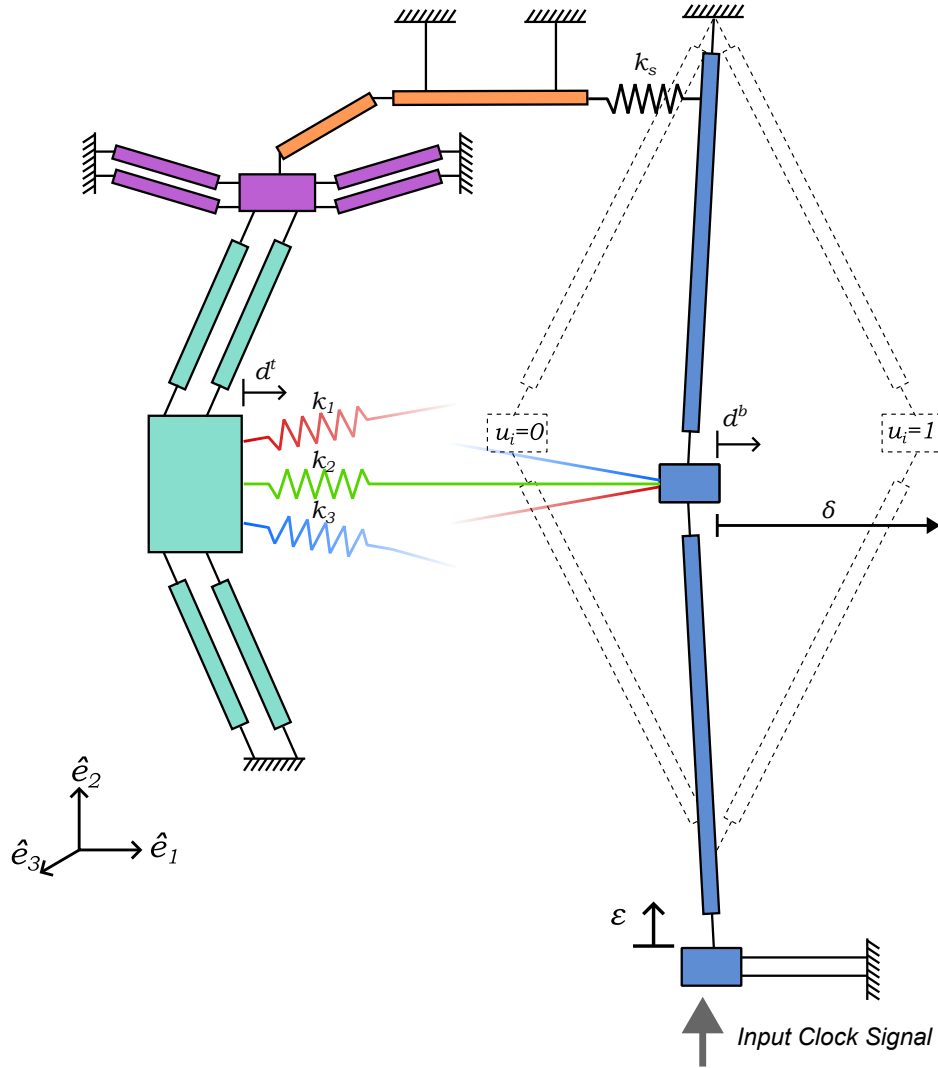


Figure 2: This is a figure.

Working Principle

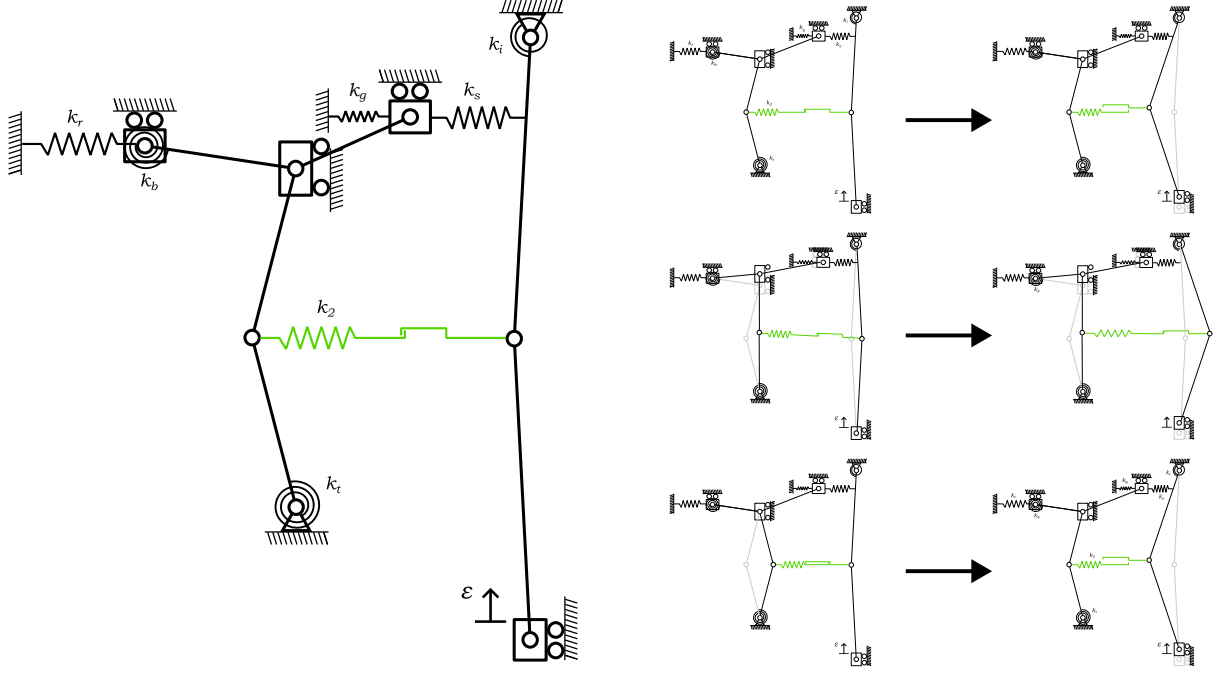


Figure 3: This is a figure.

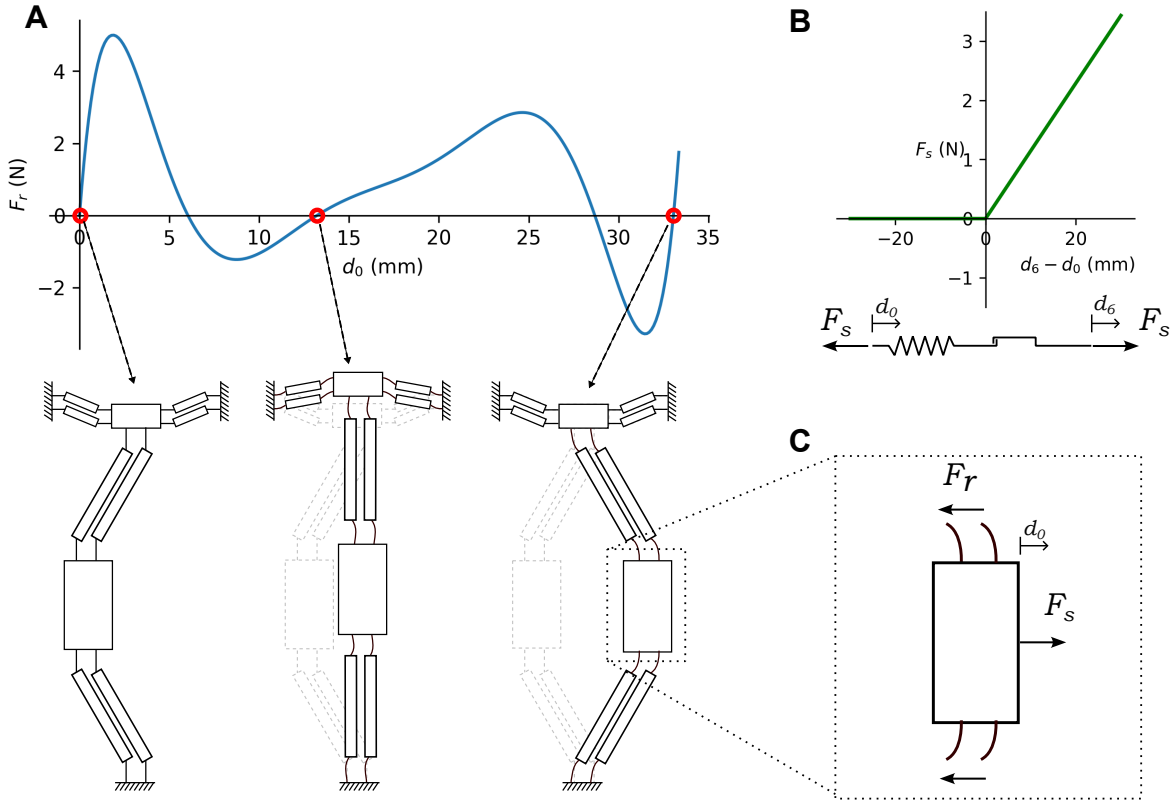


Figure 4: A. Force-displacement response of the three stable equilibria and corresponding configurations of the state element. B. Force-displacement response of the tension only spring

A Supplementary Material

Bi-planar separability of ECA rules