OPTIMAL CNN TEMPLATES FOR LINEARLY-SEPARABLE ONE-DIMENSIONAL CELLULAR AUTOMATA

P.J. Chang and Bharathwaj Muthuswamy
University of California, Berkeley
Department of Electrical Engineering and Computer Sciences
Berkeley, California

Abstract

In this tutorial, we present optimal Cellular Nonlinear Network (CNN) templates for implementing linearly-separable one-dimensional (1-D) Cellular Automata (CA). From the gallery of CNN templates presented in this paper, one can calculate any of the 256 1-D CA Rules studied by Wolfram using a CNN Universal Machine chip that is several orders of magnitude faster than conventional programming on a digital computer.

1 Introduction

The main purpose of this paper is to derive the optimal Cellular Nonlinear Network (CNN) templates for linearly-separable one-dimensional (1-D) Cellular Automata (CA). A gallery of optimal CNN templates for linearly-separable 1-D CA is presented, and an appendix is included to illustrate the template derivation algorithm. These optimal templates may be implemented on any CNN universal chip [Chua & Roska, 2002], thereby allowing faster calculation of any of the 256 1-D CA rules [Wolfram, 2002] by several orders of magnitude. Such great enhancement in speed will enable researchers on CA to conduct extensive simulations over much longer periods (e.g., over many *trillions* of iterations) of Wolfram's class 3 and 4 CA rules than currently feasible.

A. Truth Tables and Boolean Functions

Boolean Functions are described by truth tables. For the purpose of this paper, the inputs to the truth tables are binary 3-tuples, (x_{i-1}, x_i, x_{i+1}). That is, each of x_{i-1}, x_i, x_{i+1} can be either 1 or 0, leading to the generation of $2^3 = 8$ input combinations, listed in order as (0,0,0), (0,0,1), (0,1,0), (0,1,1), (1,0,0), (1,0,1), (1,1,0), (1,1,1). Each 3-tuple input is mapped to a binary output y that is also either 1 or 0. Thus, there are $2^8 = 256$ possible truth tables generated from 3 binary inputs. Each of these 256 truth tables corresponds to a unique Boolean Function, which is named as the decimal equivalent of the binary number formed by concatenating the output of each row in the truth table in reverse order. For example, when mapping (0,0,0) to 0, (0,0,1) to 1, (0,1,0) to 1, (0,1,1) to 1, (1,0,0) to 0, (1,0,1) to 1, (1,1,0) to 1, (1,1,1) to 0, the outputs may be written as the 8-tuple binary code (0,1,1,1,0,1,1,0). Concatenating these outputs in reverse order results in the binary number 01101110, whose decimal equivalent is 110. The corresponding Boolean Function is thus named as 110, which equals $0{\cdot}2^7 + 1{\cdot}2^6 + 1{\cdot}2^5 + 0{\cdot}2^4 + 1{\cdot}2^3 + 1{\cdot}2^2 + 1{\cdot}2^1 + 0{\cdot}2^0$. Figure 1 presents that truth table and the derivation of the name.

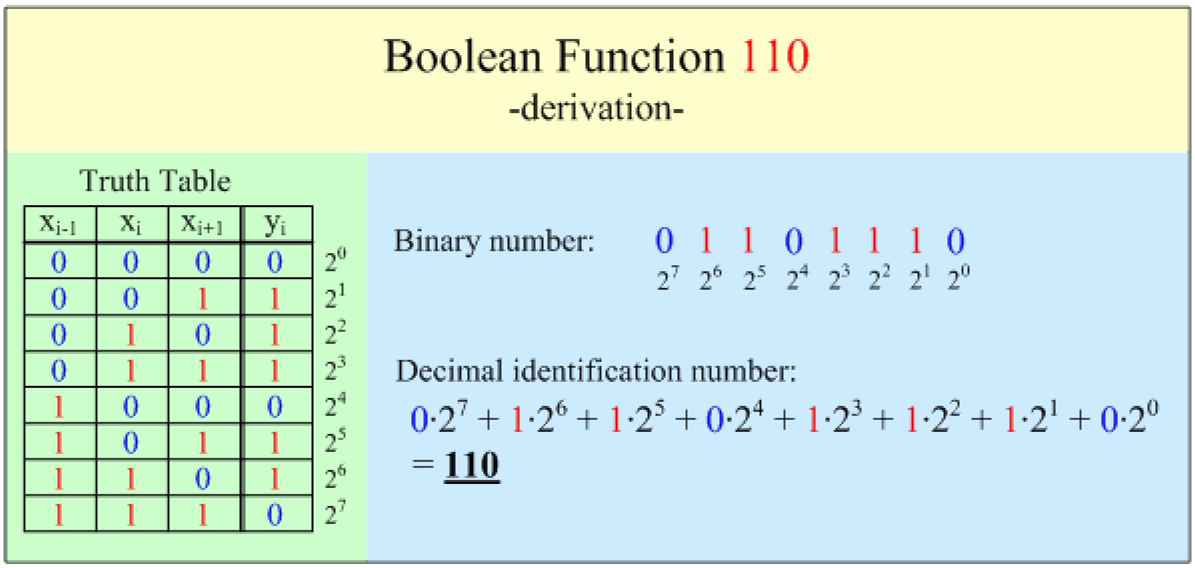


Figure 1. The derivation of Boolean Function 110 and its truth table

B. Cellular Automata

A cellular automaton is a collection of cells that iterates on a set of rules, creating a new generation of cells with each iteration. A 1-D CA is a string of cells. It is assumed that the boundary condition is *periodic*, so that the string of cells is effectively a *ring* of cells as illustrated in Figure 2 for the case of a 1-D CA made of 10 cells.

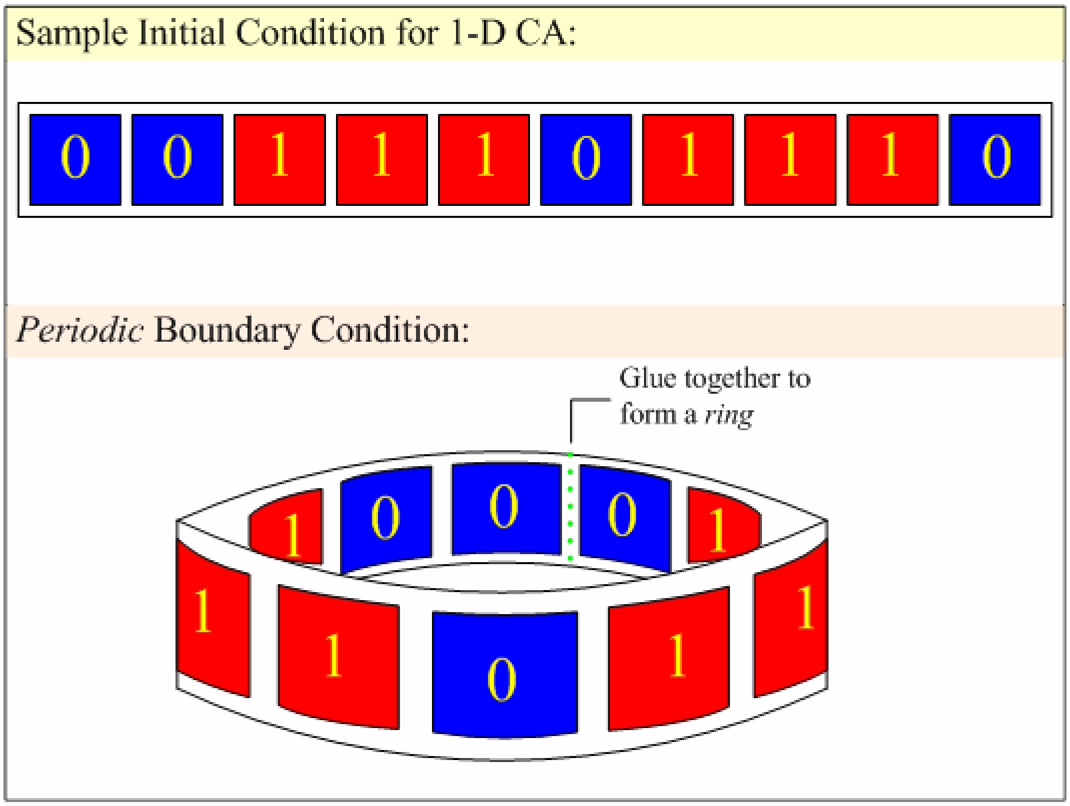


Figure 2. A sample 1-D 10-cell CA initial condition and its periodic boundary condition

The set of rules that govern the cells of the 1-D CA are summarized by 256 3-tuple truth tables. Each truth table is a Boolean Rule that dictates a distinct 1-D CA pattern from a given initial condition. For the purposes of studying 1-D CA, the initial condition would be a string of cells of an arbitrary length greater than or equal to 3 cells. Each cell would indicate a value of "0" or "1". To find the output of each cell of a particular 1-D cellular automaton according to a fixed Boolean Rule, the cell in question is seen as x_i, its left neighbor as x_{i-1}, and its right neighbor as x_{i+1}. The corresponding output y_i mapped from the resulting (x_{i-1}, x_i, x_{i+1}) in the truth table would be the output of the cell x_i. For example, let {1,1,0,1} be a sample initial condition bit string of length 4 for a 1-D 4-cell CA, given the truth table of Boolean Function 110, usually dubbed *local Rule 110* in literature. Consider the third cell that contains "0." The left and right neighbor cells are the second and fourth cells, respectively, both containing "1." The output of the third cell would be the output y mapped from the 3-tuple (1,0,1) from the truth table of local Rule 110. According to the truth table, when $x_{i-1} = 1$, $x_i = 0$, and $x_{i+1} = 1$, the output y_i is 1, and hence the output of the third cell is "1." The same process is repeated for each cell listed in the initial condition, resulting in the new *evolution* {0,1,1,1}, and completes one iteration of the 1-D CA. This new evolution, {0,1,1,1} would be the new initial condition for the next iteration. If all the cells containing “1” are colored red, and all the cells containing “0” are colored blue, then a pattern begins to emerge from the evolutions after several iterations. The first iteration for local Rule 110 is derived in Figure 3.

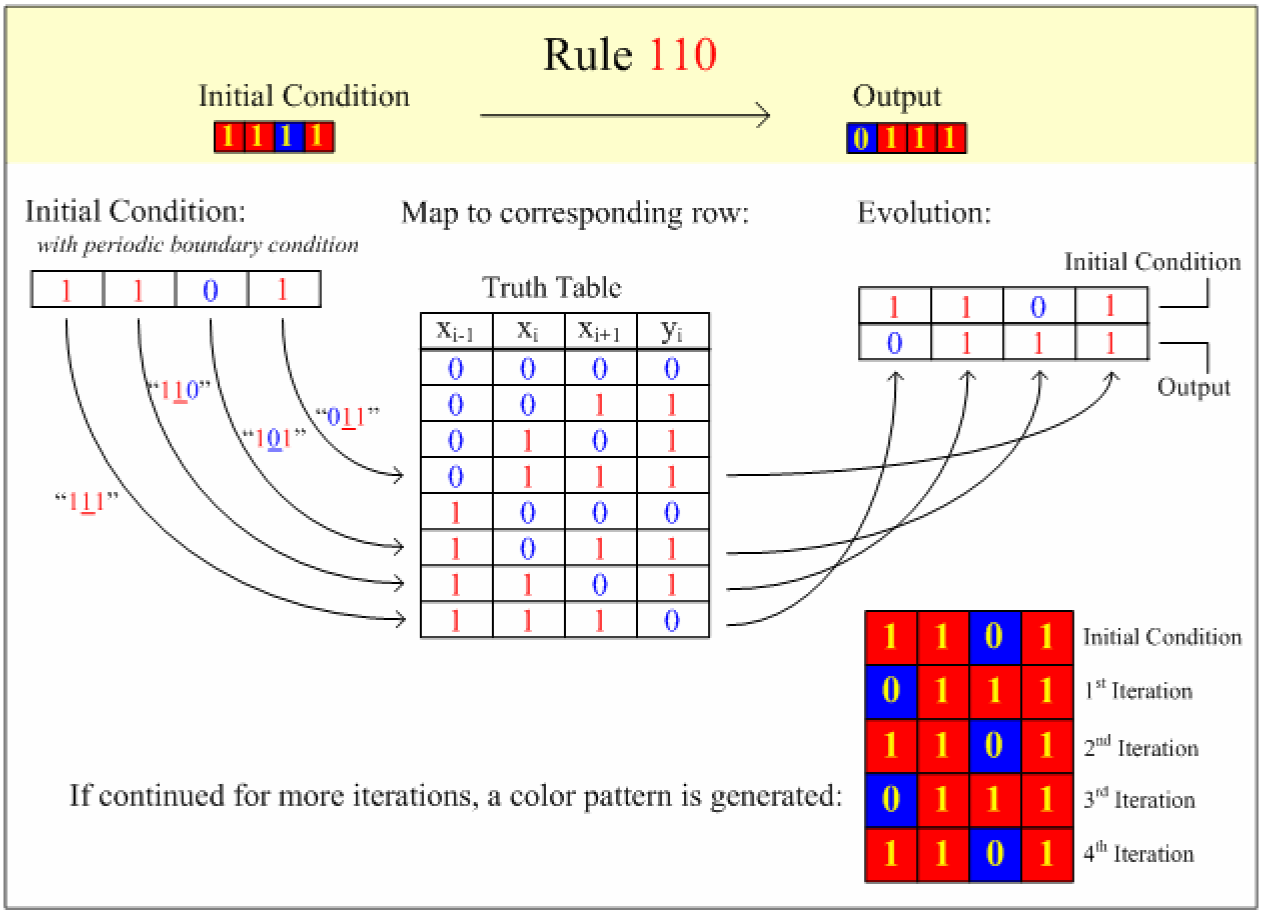


Figure 3. Starting with a sample initial condition, the next generation is found for Boolean Function 110.

In order to discover the distinct 1-D CA pattern generated by a local Rule and given initial condition, one would have to run the CA through several iterations. Some local Rules even have never-ending patterns[1] formed from almost all possible initial conditions, such as Rules 30 (binary code 00011110) and 110 (binary code 01101110), as illustrated by Figure 4, when assuming an infinite number of cells.

[1] Even for just 100 cells, there are 2^{100} distinct bit-string patterns.

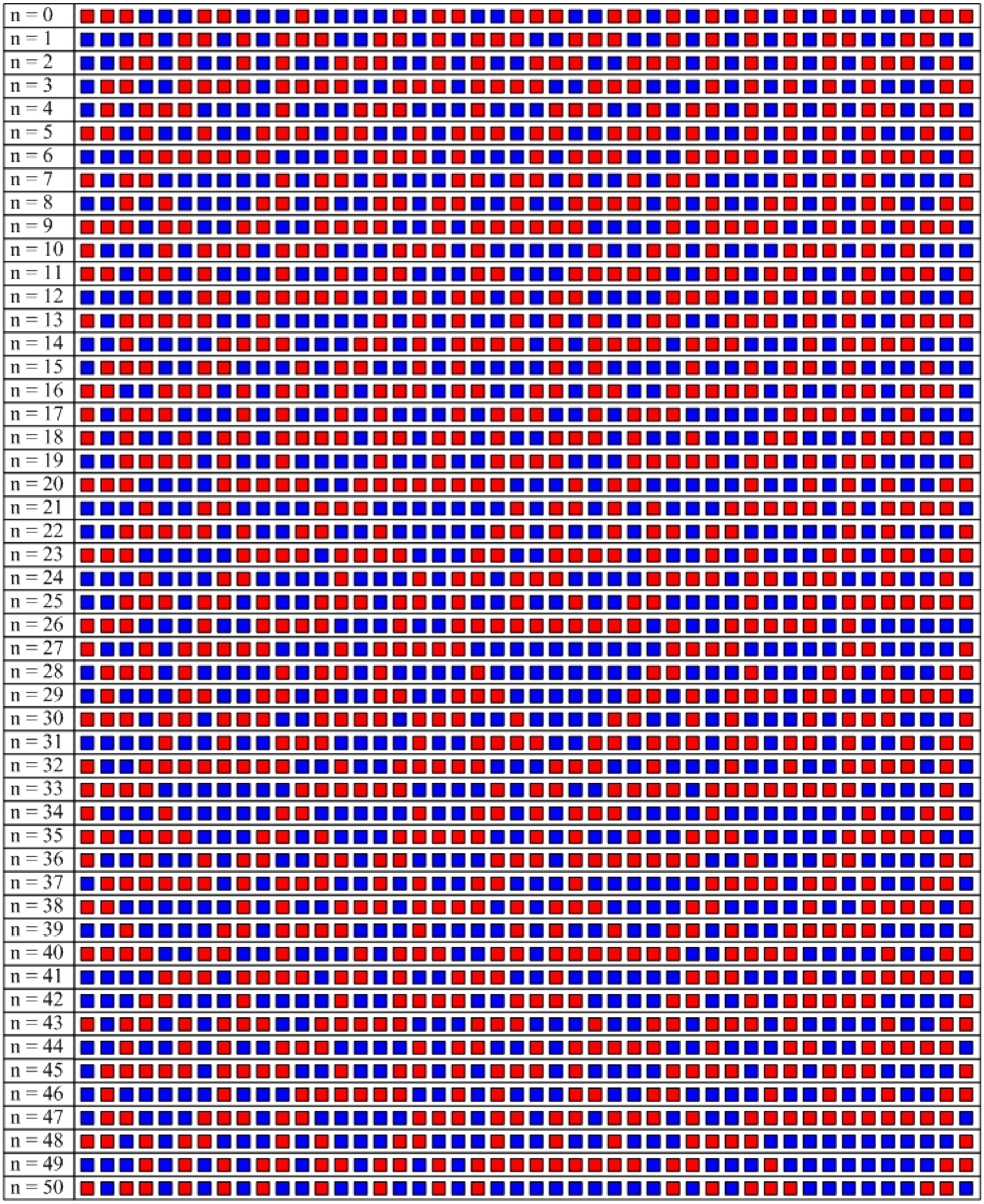


Figure 4(a). The pattern generated from rule 30 for a random initial condition. Notice how there is no repeated iteration.

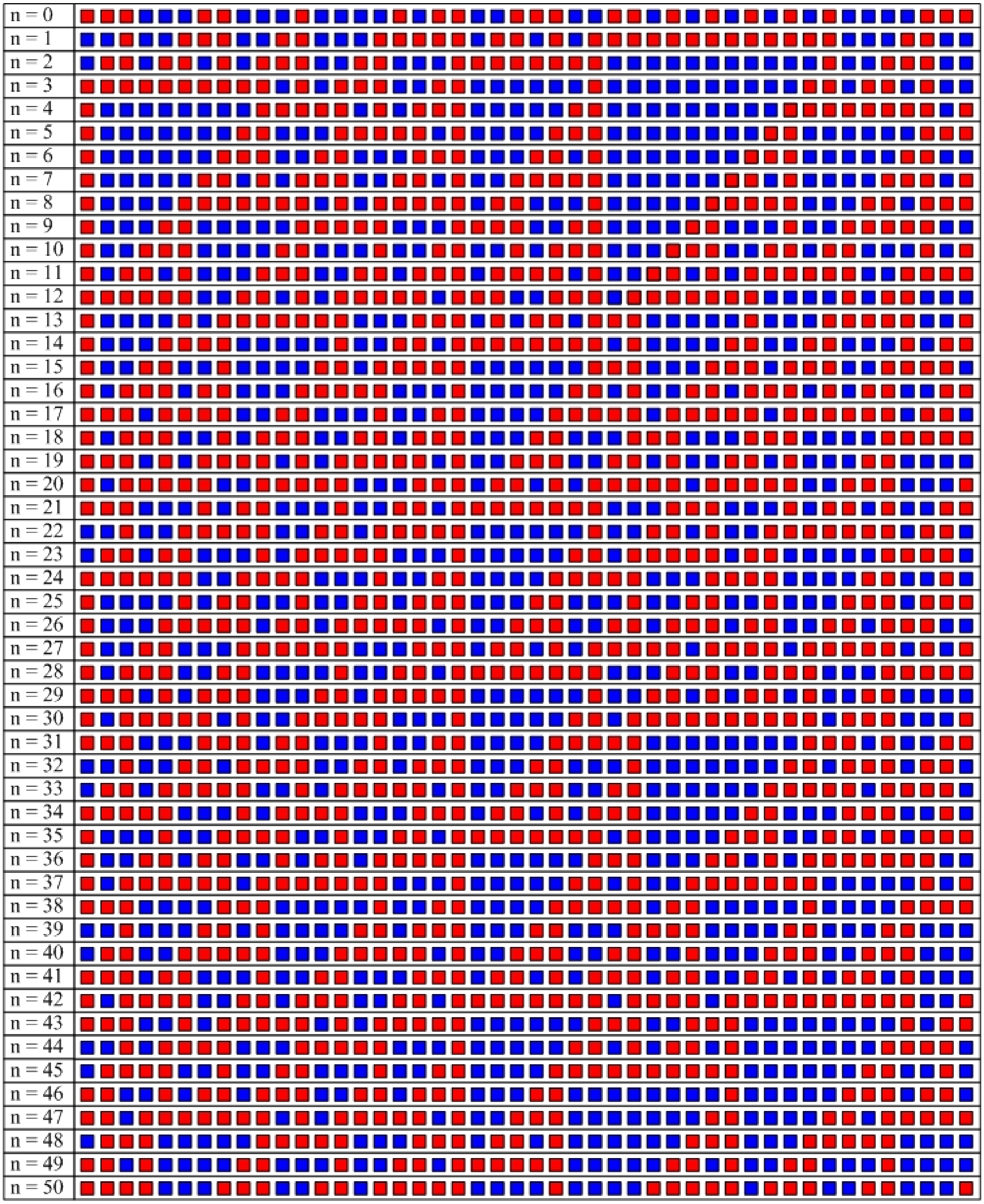


Figure 4(b). The pattern generated from rule 110 for a random initial condition. Notice how there is no repeated iteration.

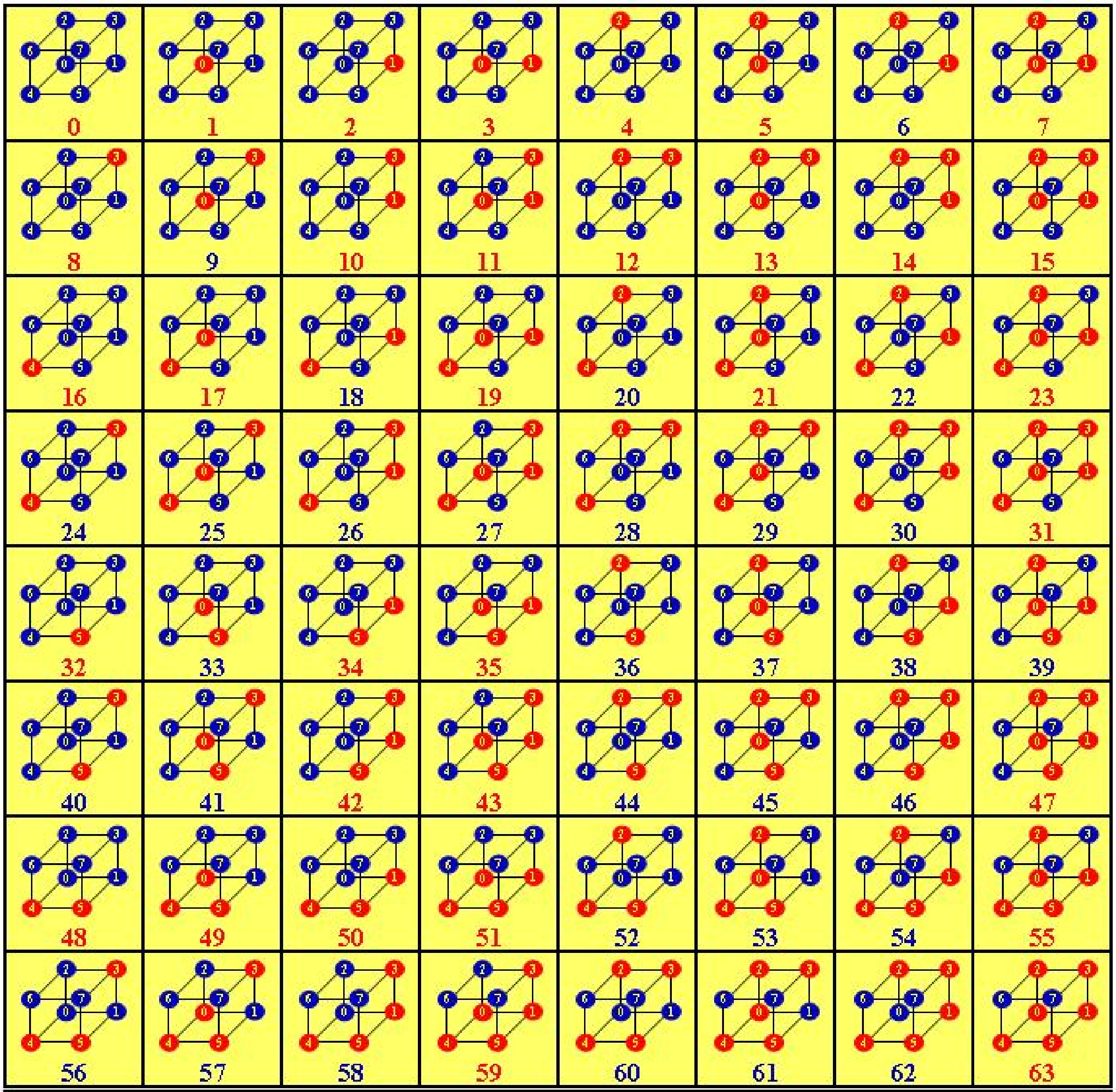
A Personal Computer (PC), which only has limited processing speed, certainly will not return these patterns instantaneously, especially if the initial condition bit string increases in length. Investigating higher dimensions of CA also poses future problems in pattern finding with a PC. A CNN universal chip is ideal for automation of n-cell 1-D CA not exceeding the array size of the chip.[2] Thus, a CNN provides a dynamical system that completely predicts the CA evolution for any initial condition. The CNN chip has high processing speed (it can process under 1 nanosecond per iteration!), low power dissipation, and parallel processing. [Chua et al, 2002] All that is needed to program a CNN universal chip to automate the pattern-generation is templates that describe the Boolean Rules. These templates are found from corresponding Boolean Cubes.

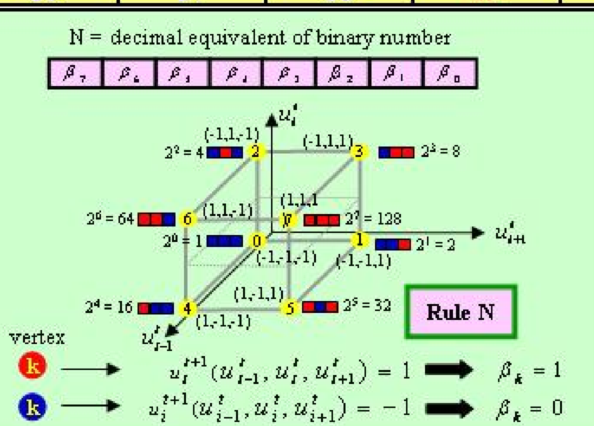
C. Boolean Cubes

Since the 8 possible outputs in each truth table is either 0 or 1, there are $2^8 = 256$ possible Boolean Functions/Rules. Each of these 256 Boolean Rules can be represented as a three-dimensional Boolean Cube for better visualization and analysis. For the reader's convenience, Table 1 from [Chua et al, 2003] is reprinted herein as Table 1. Note that the Cubes that have numbers labeled in red are *linearly-separable*, while the Cubes that have numbers labeled in blue are *linearly non-separable*; these terms will be clarified in the following Section D. The lower box describes the coloring of the vertices of the Cubes, and how the vertices' coordinates are derived from substituting "-1" for "0" for use in determining an analytical formula. Please refer to Figure 5 in deriving the analytical truth table and hence the coordinates of the vertices.

[2] The current commercially available CNN chip has an array size of 176 x 144 [Anafocus, 2007].

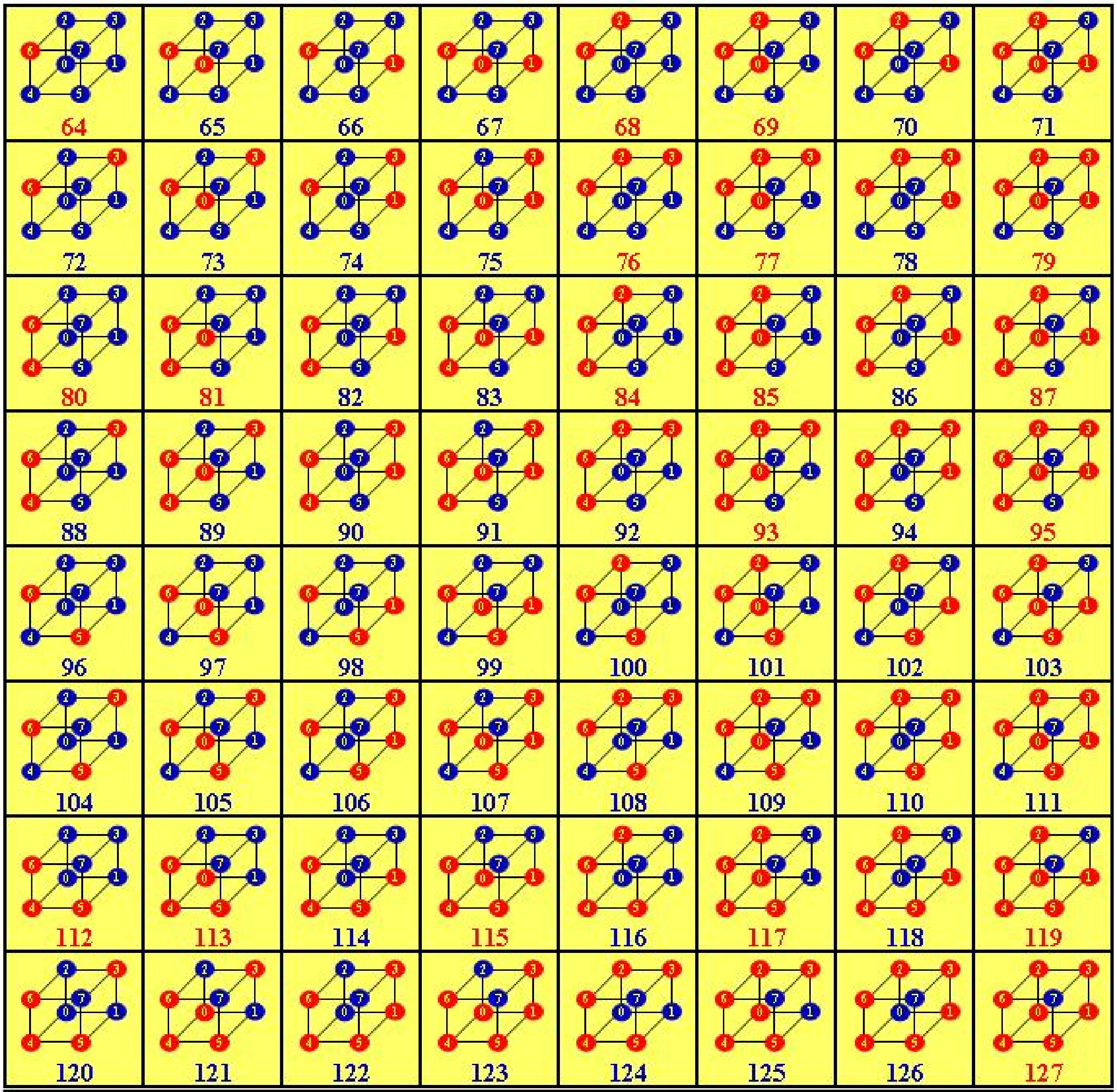
Table 1. Encoding 256 local rules defining a binary 1D CA onto 256 corresponding “Boolean Cubes” [Chua et. al, 2003].

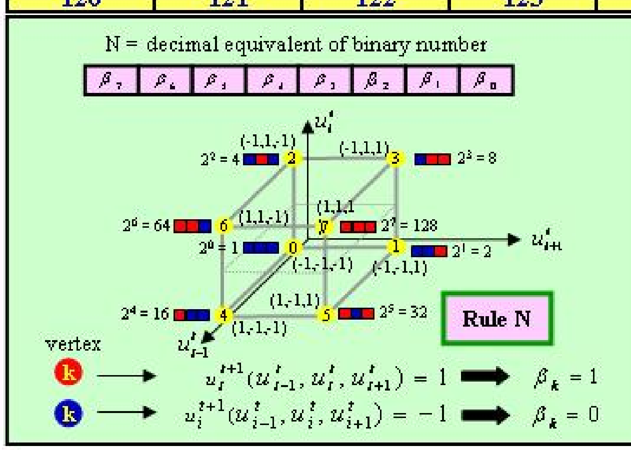




vertex k	u_{i-1}^t	u_i^t	u_{i+1}^t	u_i^{t+1}
0	-1	-1	-1	γ_0
1	-1	-1	1	γ_1
2	-1	1	-1	γ_2
3	-1	1	1	γ_3
4	1	-1	-1	γ_4
5	1	-1	1	γ_5
6	1	1	-1	γ_6
7	1	1	1	γ_7

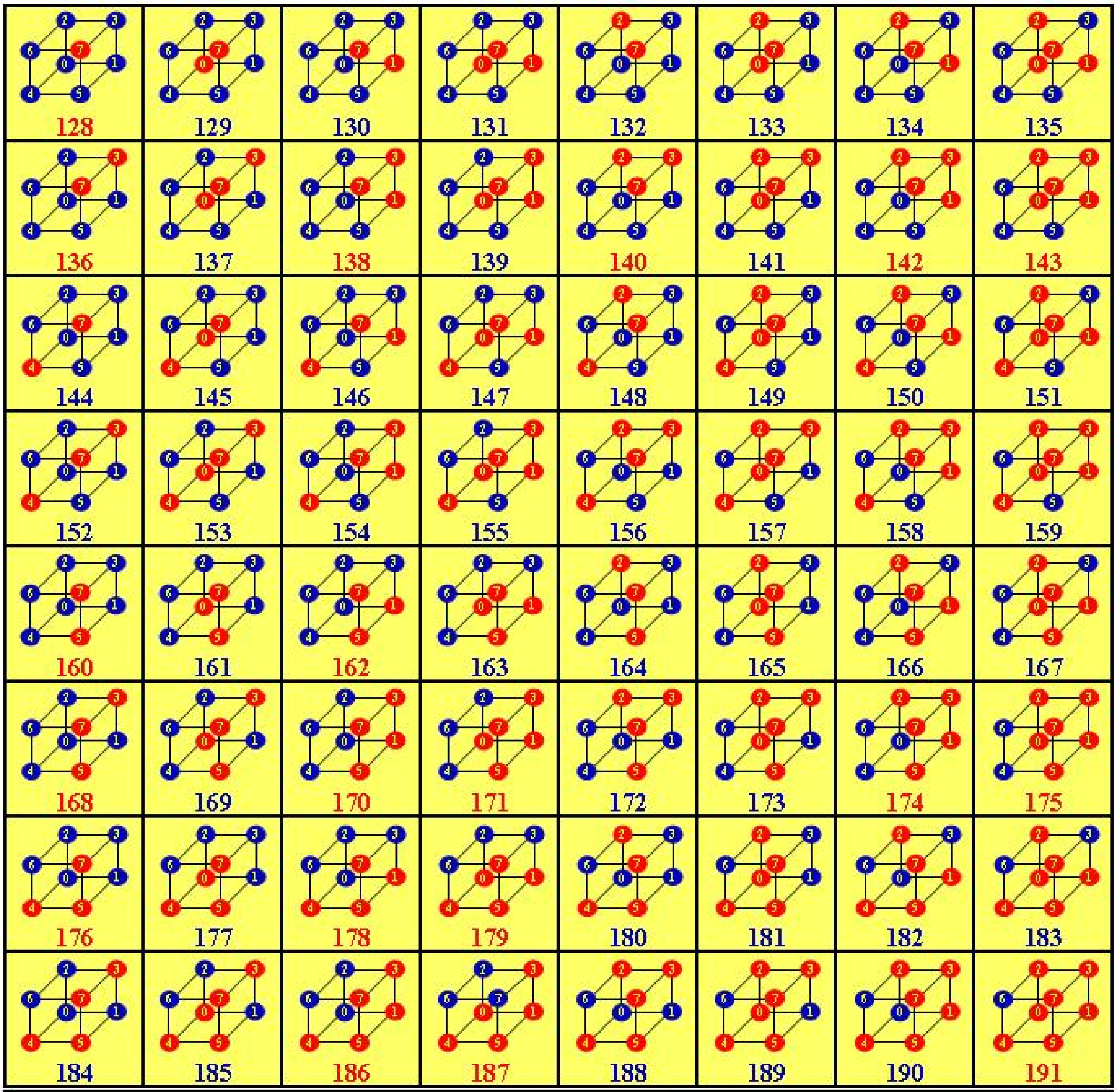
Table 1 (*continued*)

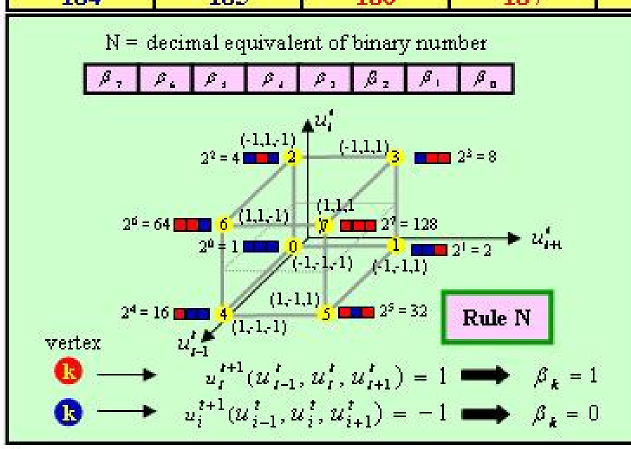




vertex (k)	u_{i-1}^t	u_i^t	u_{i+1}^t	u_i^{t+1}
0	-1	-1	-1	γ_0
1	-1	-1	1	γ_1
2	-1	1	-1	γ_2
3	-1	1	1	γ_3
4	1	-1	-1	γ_4
5	1	-1	1	γ_5
6	1	1	-1	γ_6
7	1	1	1	γ_7

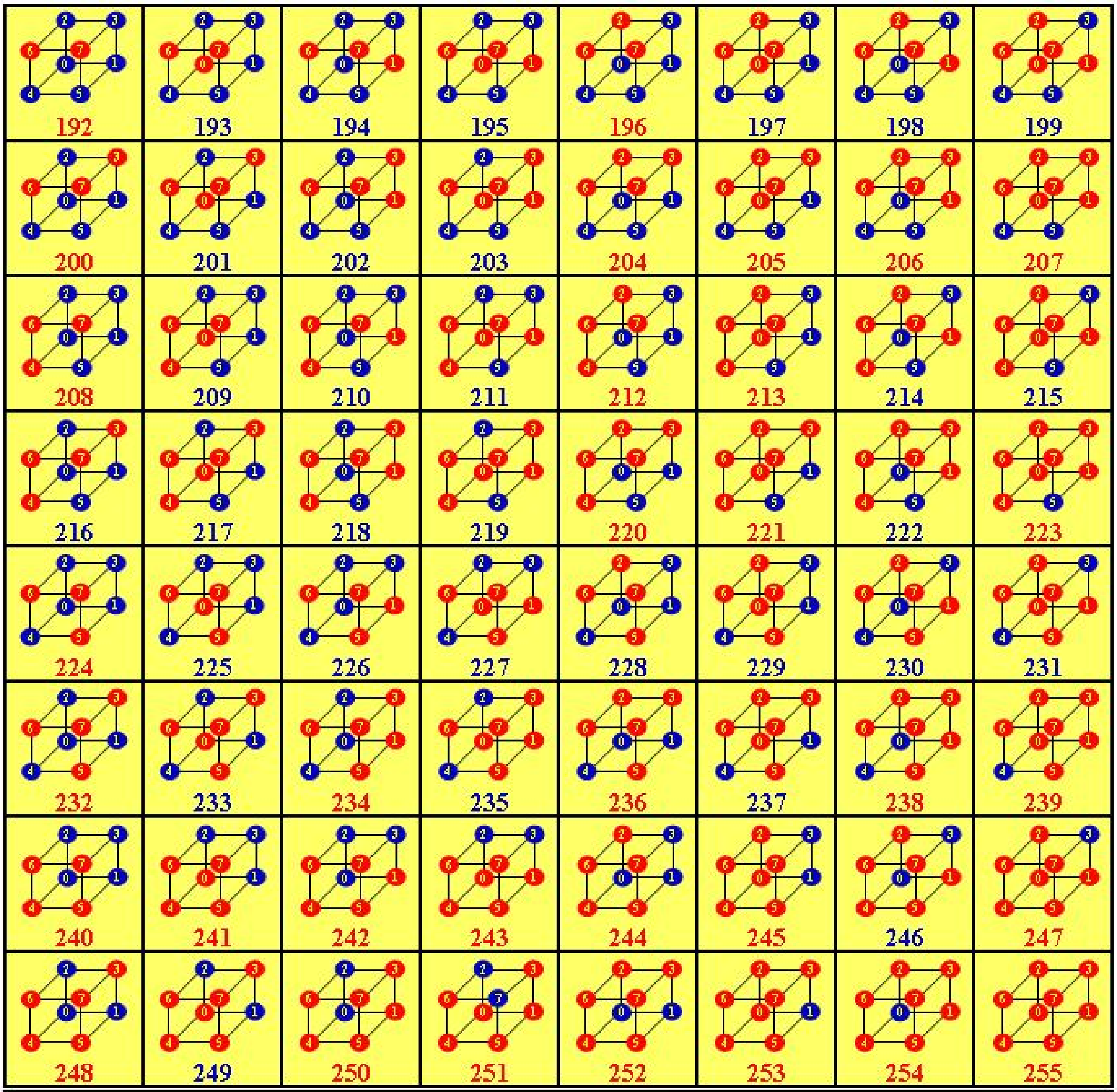
Table 1 (*continued*)

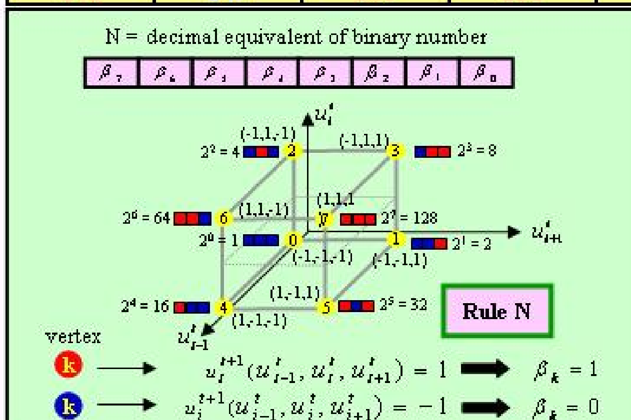




vertex ⓚ	u_{i-1}^t	u_i^t	u_{i+1}^t	u_i^{t+1}
⓪	-1	-1	-1	γ_0
①	-1	-1	1	γ_1
②	-1	1	-1	γ_2
③	-1	1	1	γ_3
④	1	-1	-1	γ_4
⑤	1	-1	1	γ_5
⑥	1	1	-1	γ_6
⑦	1	1	1	γ_7

Table 1 (*continued*)





vertex ⓚ	u_{i-1}^t	u_i^t	u_{i+1}^t	u_i^{t+1}
⓪	-1	-1	-1	γ_0
①	-1	-1	1	γ_1
②	-1	1	-1	γ_2
③	-1	1	1	γ_3
④	1	-1	-1	γ_4
⑤	1	-1	1	γ_5
⑥	1	1	-1	γ_6
⑦	1	1	1	γ_7

To recapitulate, in determining the Cube, if all the symbolic 0 values in the truth tables are replaced as -1 for numerical analytical purposes, then the new input values u_{i-1}, u_i, u_{i+1} would be the coordinates of the Cube's vertices, with u = x - 1. The vertices would be colored according to the associated output y_i such that each Rule is represented as a unique Cube; vertices would be colored blue if the associated output is -1, and vertices would be colored red if the associated output is 1. The corresponding analytic truth table and Boolean Cube for Rule 110 is presented in Figure 5.

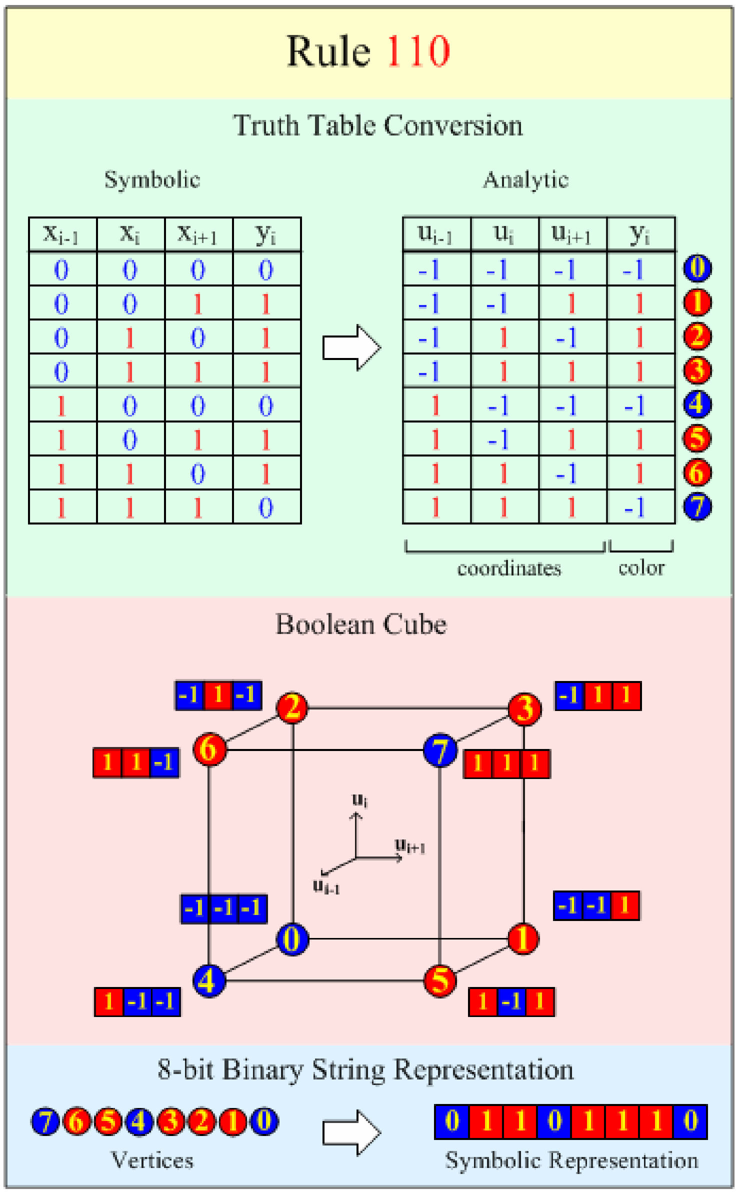


Figure 5. Converting symbolic truth table into analytic truth table to derive Boolean Cube. The 8-bit binary string is the binary number whose decimal equivalent is the name of the local Rule.

D. Complexity Index

These 256 Boolean Cubes may be categorized according to *complexity*. Each Cube is assigned a *complexity index* κ, which is the minimum number of planes needed to separate red vertices to one side of the plane, and blue vertices to the other. This complexity index is determined by inspection of the color of the 8 vertices in the Boolean Cube. Cubes that have κ= 1 require only 1 plane to separate the red vertices from the blue, and are *linearly-separable*, e.g. the Cube for Rule 16. Cubes that have κ= 2 require 2 planes, such as the Cube for Rule 18, and Cubes that have κ= 3 require 3 planes, such as the Cube for Rule 150, as shown in Figure 6. These Cubes that require more than 1 plane to separate the red and blue vertices are *linearly non-separable*. The highest complexity index possible for the 256 1-D CA Boolean Cubes is 3.

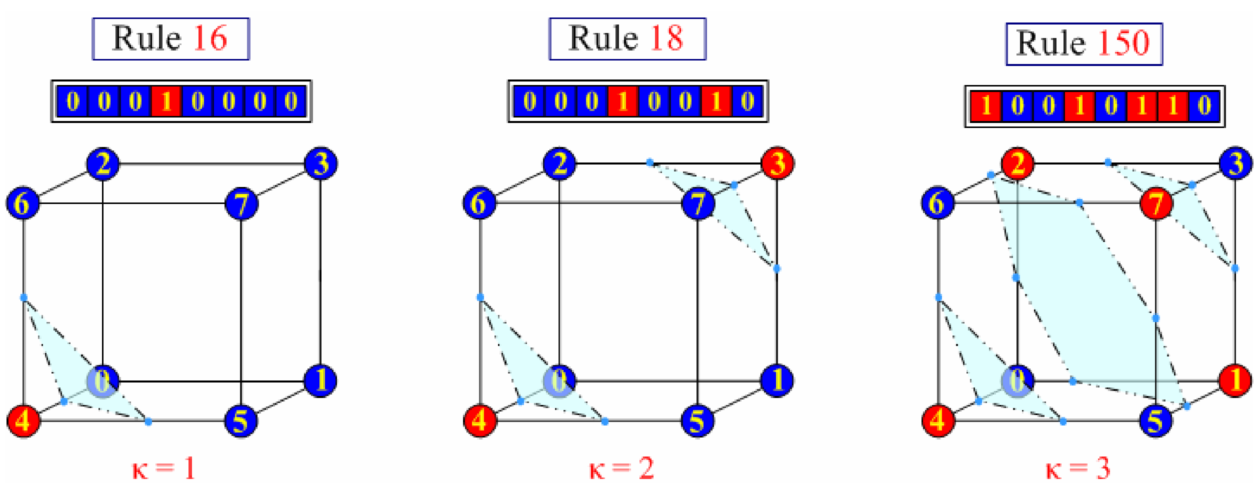


Figure 6. Rule 16 has κ=1, Rule 18 has κ=2, Rule 150 has κ= 3. The 8-bit binary string above each Boolean Cube denotes the binary number whose decimal equivalent is the name of each local Rule.

There are 104 Boolean Cubes that are linearly-separable; the corresponding 104 Boolean Rules are listed in Table 2. Tables 3 and 4 list κ= 2 and κ= 3 Rules respectively. To program a CNN universal chip to generate the pattern associated with a given Rule, templates describing the Rule must be specified as data input of the program automatically by the CNN operating system that comes with the chip.

Table 2. All 104 Linearly-Separable Rules, κ= 1

0	1	2	3	4	5	7	8	10	11	12	13	14
15	16	17	19	21	23	31	32	34	35	42	43	47
48	49	50	51	55	59	63	64	68	69	76	77	79
80	81	84	85	87	93	95	112	113	115	117	119	127
128	136	138	140	142	143	160	162	168	170	171	174	175
176	178	179	186	187	191	192	196	200	204	205	206	207
208	212	213	220	221	223	224	232	234	236	238	239	240
241	242	243	244	245	247	248	250	251	252	253	254	255

Table 3. All κ= 2 Rules

6	9	18	20	22	24	25	26	28	30	33	36	37
38	40	41	44	45	52	54	56	57	60	61	62	65
66	67	70	72	73	74	75	82	86	88	89	90	91
94	96	97	98	99	100	101	102	103	104	106	107	108
109	110	111	118	120	121	122	123	124	125	126	129	130
131	132	133	134	135	137	144	145	146	147	148	149	151
152	153	154	155	156	157	158	159	161	164	165	166	167
169	173	180	181	182	183	185	188	189	190	193	194	195
198	199	201	203	210	211	214	215	217	218	219	222	225
227	229	230	231	233	235	237	246	249				

Table 4. All κ= 3 Rules

27	29	39	46	53	58	71	78	83
92	105	114	116	139	141	150	163	172
177	184	197	202	209	216	226	228	

E. Templates

Three templates are needed to program a CNN universal chip: the A template, the B template, and the z template (Figure 8). The A template is the feedback term of the CNN nonlinear differential equation. For simplicity and robustness, a_{00} is chosen to be 1, while the other cells of the templates are 0. In the linearly-separable cases, the B and z templates are obtained from the *orientation vector* [Chua et al, 2002] and [Dogaru & Chua, 1998], or normal vector, and the offset of the single separating plane respectively. These templates also convey the equation that describes the outputs of the corresponding truth table. Usually, the output equation is a *signum* function, *sgn*, involving analytical variables u_{i-1}, u_i, u_{i+1}, since the CNN chip operates analytically. This signum function can be converted into a *sign* function, s, involving symbolic variables x_{i-1}, x_i, x_{i+1} in accordance to symbolic truth tables for better understanding. The signum and sign functions are presented in Figure 7. The separating plane and resulting B and z templates are shown in Figure 8 for Rule 204, which is linearly separable.

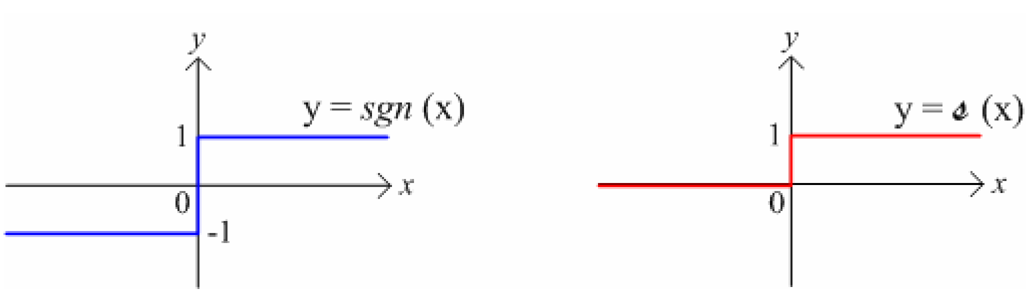


Figure 7. y = sgn (x) and y = s(x), respectively.

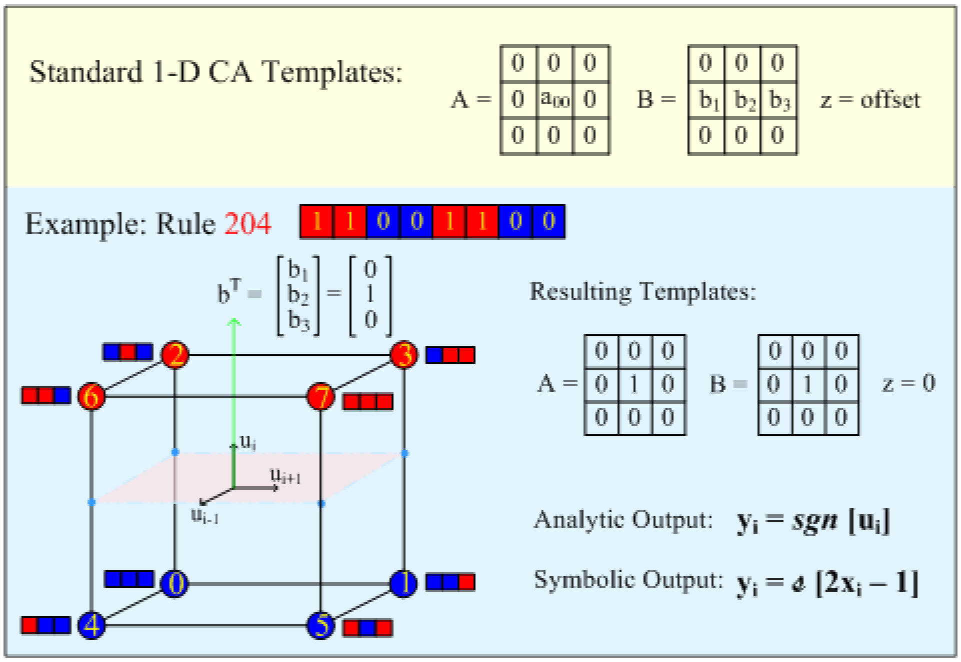


Figure 8. The Standard 1-D CA Templates and the derivation of the templates for Rule 204.

Rules that have higher complexity indices are not linearly-separable, but they always can be decomposed into two or three linearly-separable Rules by Boolean Operations (AND , OR). These Rules are also known as *linearly non-separable* Rules. An example is Rule 184, which is equivalent to Rule 186 AND Rule 248 as presented by Figure 9. Table 5 provides a synthesis of all the higher complexity Rules.

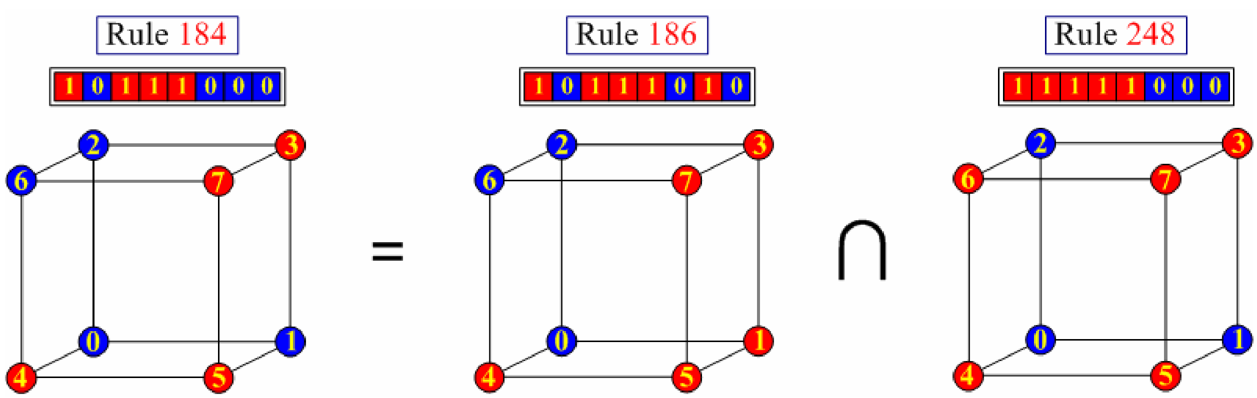


Figure 9. Rule 184 is Rule 186 AND Rule 248.

Table 5. Synthesis of Linearly Non-Separable Boolean Rules via Boolean Operation (AND, OR) using Linearly-Separable Rules

Rule N	Synthesis using Rules from Table 1		
Rule 6	Rule 7	AND	Rule 14
Rule 9	Rule 11	AND	Rule 13
Rule 18	Rule 19	AND	Rule 50
Rule 20	Rule 21	AND	Rule 84
Rule 22	Rule 23	AND	Rule 254
Rule 24	Rule 31	AND	Rule 248
Rule 25	Rule 59	AND	Rule 221
Rule 26	Rule 31	AND	Rule 250
Rule 27	Rule 31	AND	Rule 251
Rule 28	Rule 31	AND	Rule 220
Rule 29	Rule 31	AND	Rule 93
Rule 30	Rule 31	AND	Rule 254
Rule 33	Rule 35	AND	Rule 49
Rule 36	Rule 47	AND	Rule 244
Rule 37	Rule 47	AND	Rule 117
Rule 38	Rule 55	AND	Rule 238
Rule 39	Rule 47	AND	Rule 55
Rule 40	Rule 42	AND	Rule 168
Rule 41	Rule 43	AND	Rule 253
Rule 44	Rule 47	AND	Rule 236
Rule 45	Rule 47	AND	Rule 253
Rule 46	Rule 47	AND	Rule 254
Rule 52	Rule 55	AND	Rule 244
Rule 53	Rule 55	AND	Rule 245
Rule 54	Rule 55	AND	Rule 254
Rule 56	Rule 59	AND	Rule 248

Rule N	Synthesis using Rules from Table 1		
Rule 57	Rule 59	AND	Rule 253
Rule 58	Rule 59	AND	Rule 254
Rule 60	Rule 63	AND	Rule 252
Rule 61	Rule 63	AND	Rule 253
Rule 62	Rule 63	AND	Rule 254
Rule 65	Rule 69	AND	Rule 81
Rule 66	Rule 79	AND	Rule 242
Rule 67	Rule 79	AND	Rule 243
Rule 70	Rule 87	AND	Rule 206
Rule 71	Rule 79	AND	Rule 87
Rule 72	Rule 76	AND	Rule 232
Rule 73	Rule 77	AND	Rule 251
Rule 74	Rule 79	AND	Rule 234
Rule 75	Rule 79	AND	Rule 251
Rule 78	Rule 79	AND	Rule 254
Rule 82	Rule 87	AND	Rule 250
Rule 83	Rule 87	AND	Rule 243
Rule 86	Rule 87	AND	Rule 254
Rule 88	Rule 93	AND	Rule 248
Rule 89	Rule 93	AND	Rule 251
Rule 90	Rule 95	AND	Rule 250
Rule 91	Rule 95	AND	Rule 251
Rule 92	Rule 95	AND	Rule 220
Rule 94	Rule 95	AND	Rule 254
Rule 96	Rule 112	AND	Rule 224
Rule 97	Rule 113	AND	Rule 239

Table 5 (*continued*)

Rule N	Synthesis using Rules from Table 1
Rule 98	Rule 115 AND Rule 234
Rule 99	Rule 115 AND Rule 239
Rule 100	Rule 117 AND Rule 238
Rule 101	Rule 117 AND Rule 239
Rule 102	Rule 119 AND Rule 238
Rule 103	Rule 119 AND Rule 239
Rule 104	Rule 127 AND Rule 232
Rule 105	(Rule 43 OR Rule 64) AND Rule 253
Rule 106	Rule 127 AND Rule 234
Rule 107	Rule 43 OR Rule 64
Rule 108	Rule 127 AND Rule 236
Rule 109	Rule 32 OR Rule 77
Rule 110	Rule 127 AND Rule 238
Rule 111	Rule 127 AND Rule 239
Rule 114	Rule 115 AND Rule 242
Rule 116	Rule 117 AND Rule 252
Rule 118	Rule 119 AND Rule 254
Rule 120	Rule 127 AND Rule 248
Rule 121	Rule 8 OR Rule 113
Rule 122	Rule 127 AND Rule 250
Rule 123	Rule 127 AND Rule 251
Rule 124	Rule 127 AND Rule 252
Rule 125	Rule 127 AND Rule 253
Rule 126	Rule 127 AND Rule 254
Rule 129	Rule 143 AND Rule 241
Rule 130	Rule 138 AND Rule 162

Rule N	Synthesis using Rules from Table 1
Rule 131	Rule 143 AND Rule 243
Rule 132	Rule 140 AND Rule 196
Rule 133	Rule 143 AND Rule 245
Rule 134	Rule 142 AND Rule 247
Rule 135	Rule 143 AND Rule 247
Rule 137	Rule 171 AND Rule 221
Rule 139	Rule 143 AND Rule 251
Rule 141	Rule 143 AND Rule 205
Rule 144	Rule 176 AND Rule 213
Rule 145	Rule 179 AND Rule 213
Rule 146	Rule 178 AND Rule 223
Rule 147	Rule 179 AND Rule 223
Rule 148	Rule 191 AND Rule 212
Rule 149	Rule 191 AND Rule 213
Rule 150	(Rule 23 OR Rule 128) AND Rule 254
Rule 151	Rule 23 OR Rule 128
Rule 152	Rule 186 AND Rule 220
Rule 153	Rule 187 AND Rule 221
Rule 154	Rule 186 AND Rule 223
Rule 155	Rule 187 AND Rule 223
Rule 156	Rule 191 AND Rule 220
Rule 157	Rule 191 AND Rule 221
Rule 158	Rule 16 OR Rule 142
Rule 159	Rule 191 AND Rule 223
Rule 161	Rule 171 AND Rule 241
Rule 163	Rule 171 AND Rule 179

Table 5 (*continued*)

Rule N	Synthesis using Rules from Table 1
Rule 164	Rule 174 AND Rule 244
Rule 165	Rule 175 AND Rule 245
Rule 166	Rule 174 AND Rule 247
Rule 167	Rule 175 AND Rule 247
Rule 169	Rule 171 AND Rule 253
Rule 172	Rule 174 AND Rule 253
Rule 173	Rule 175 AND Rule 253
Rule 177	Rule 179 AND Rule 241
Rule 180	Rule 191 AND Rule 244
Rule 181	Rule 191 AND Rule 245
Rule 182	Rule 4 OR Rule 178
Rule 183	Rule 191 AND Rule 247
Rule 184	Rule 186 AND Rule 248
Rule 185	Rule 187 AND Rule 253
Rule 188	Rule 191 AND Rule 252
Rule 189	Rule 191 AND Rule 253
Rule 190	Rule 191 AND Rule 254
Rule 193	Rule 205 AND Rule 241
Rule 194	Rule 206 AND Rule 242
Rule 195	Rule 207 AND Rule 243
Rule 197	Rule 205 AND Rule 213
Rule 198	Rule 206 AND Rule 247
Rule 199	Rule 207 AND Rule 247
Rule 201	Rule 205 AND Rule 251
Rule 202	Rule 206 AND Rule 251
Rule 203	Rule 207 AND Rule 251

Rule N	Synthesis using Rules from Table 1
Rule 209	Rule 213 AND Rule 241
Rule 210	Rule 223 AND Rule 242
Rule 211	Rule 223 AND Rule 243
Rule 214	Rule 2 OR Rule 212
Rule 215	Rule 223 AND Rule 247
Rule 216	Rule 220 AND Rule 250
Rule 217	Rule 221 AND Rule 251
Rule 218	Rule 223 AND Rule 250
Rule 219	Rule 223 AND Rule 251
Rule 222	Rule 223 AND Rule 254
Rule 225	Rule 239 AND Rule 241
Rule 226	Rule 239 AND Rule 242
Rule 227	Rule 239 AND Rule 243
Rule 228	Rule 239 AND Rule 244
Rule 229	Rule 239 AND Rule 245
Rule 230	Rule 238 AND Rule 247
Rule 231	Rule 239 AND Rule 247
Rule 233	Rule 1 OR Rule 232
Rule 235	Rule 239 AND Rule 251
Rule 237	Rule 239 AND Rule 253
Rule 246	Rule 247 AND Rule 254
Rule 249	Rule 251 AND Rule 253

Therefore, instead of finding the templates for all 256 1-D CA Rules, it is sufficient to implement the templates of the 104 linearly-separable Rules to program the CNN universal chip.

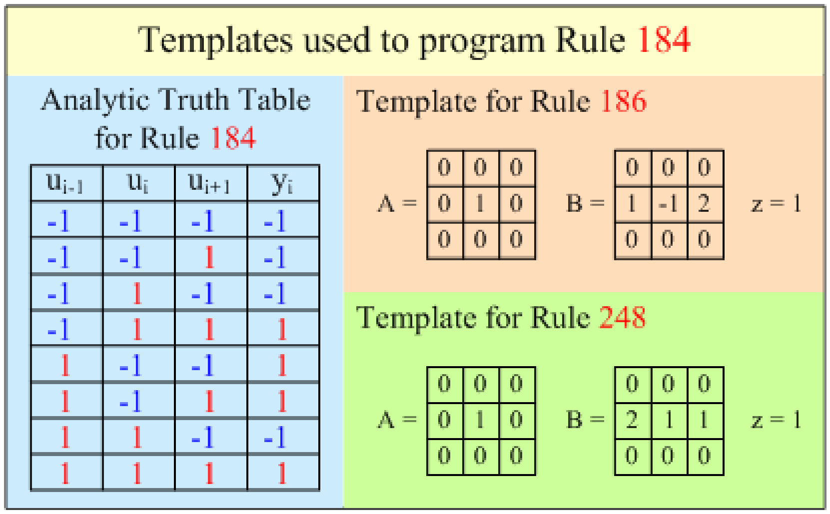


Figure 10. In order to program the CNN chip for Rule 184, the templates for Rules 186 and 248 are used.

F. Optimal Templates

Mathematically speaking, there are an infinite number of possible separating planes. Since the B and z templates are based on the normal vector and the offset of the separating plane, respectively, there are also an infinite number of possible templates per Boolean Rule. This is easily seen in the example of Rule 16, shown in Figure 11.

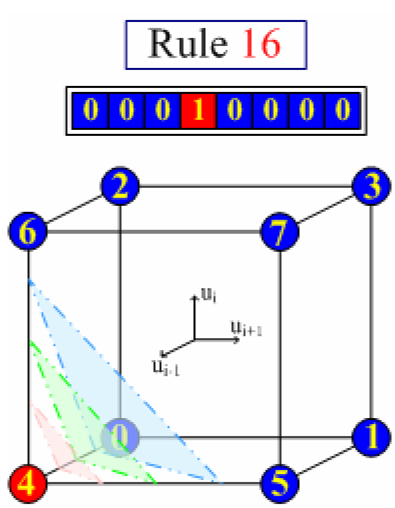


Figure 11. Here are 3 possible separating planes for Rule 16.

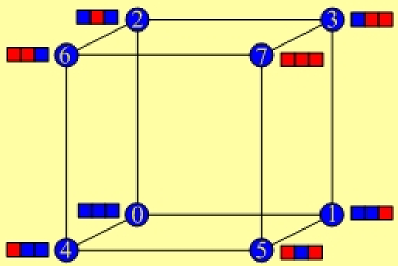
To provide a basis for studies concerning CNN universal chips and Boolean Rules, it is best to adhere to a standard set of templates. Among the infinite template possibilities, there is only one set of *optimal CNN templates*. In this paper, the *optimal separating plane* is defined as the plane that is at the maximum projected distance possible from each of the cube's 8 vertices, but which still separates the red vertices from the blue ones. The corresponding optimal CNN templates are derived from the normal vectors and offsets of these optimal separating planes. An algorithm to find this plane is provided in the Appendix. In short, a plane is selected that satisfactorily separates the red vertices from the blue. The projected distance from each vertex to the plane is found, and the minimum distance is maximized by adjusting the plane until an optimal plane is achieved.

2 Gallery of Templates

The optimal CNN templates of the 104 linearly-separable 1-D CA are implemented and presented in table 6. The truth table, Cube, templates, and output formula are presented for each of the 104 Boolean Rules. The truth tables and output formula use symbolic variables x_{i-1}, x_i, x_{i+1} for better understanding, but the templates are found based on analytic variables u_{i-1}, u_i, u_{i+1} since the CNN universal chip operates on a numerical analytical basis.

Table 6. A Gallery of Templates of all 104 Linearly Separable CA

Rule 0 — κ = 1



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

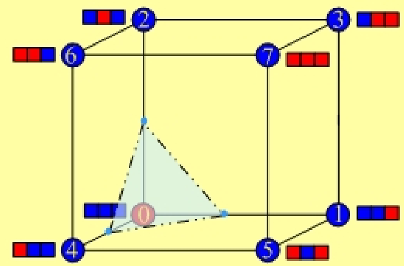
Template

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad z = -1$$

Formula

$y = -1$

Rule 1 — κ = 1



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

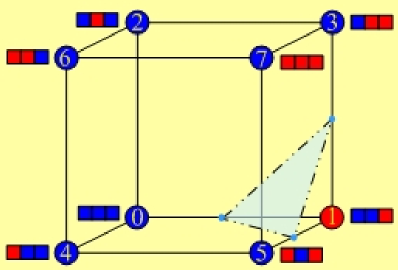
Template

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ -1 & -1 & -1 \\ 0 & 0 & 0 \end{bmatrix} \quad z = -2$$

Formula

$y = ȼ[\,-2x_{i-1} - 2x_i - 2x_{i+1} + 1\,]$

Rule 2 — κ = 1



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

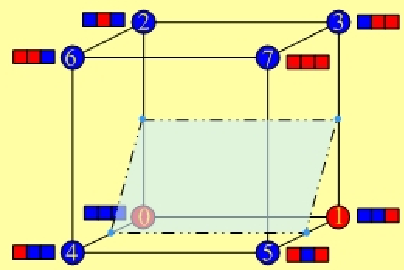
Template

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ -1 & -1 & 1 \\ 0 & 0 & 0 \end{bmatrix} \quad z = -2$$

Formula

$y = ȼ[\,-2x_{i-1} - 2x_i + 2x_{i+1} - 1\,]$

Rule 3 — κ = 1



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

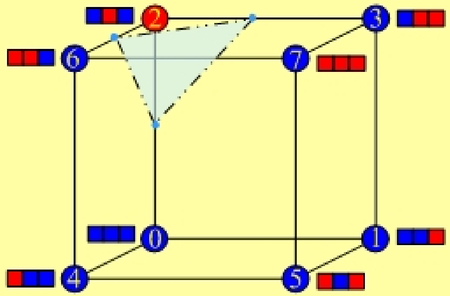
Template

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ -1 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad z = -1$$

Formula

$y = ȼ[\,-2x_{i-1} - 2x_i + 1\,]$

Rule 4 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

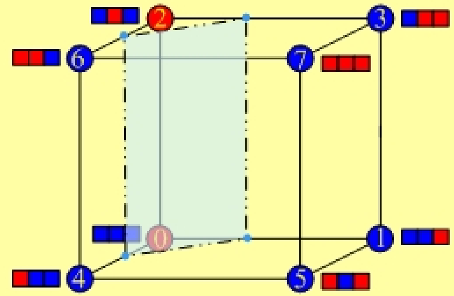
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ -1 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = -2$

Formula

$y = \phi\,[\,-2x_{i-1} + 2x_i - 2x_{i+1} - 1\,]$

Rule 5 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

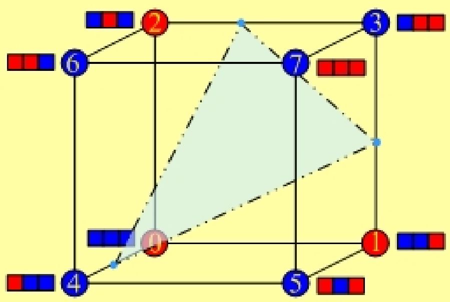
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ -1 & 0 & -1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = -1$

Formula

$y = \phi\,[\,-2x_{i-1} - 2x_{i+1} + 1\,]$

Rule 7 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

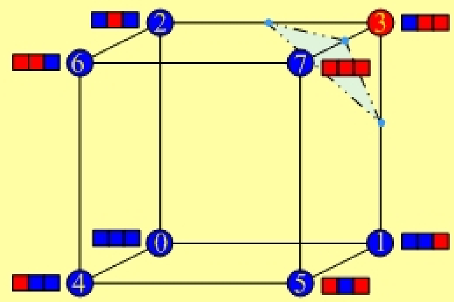
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ -2 & -1 & -1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = -1$

Formula

$y = \phi\,[\,-4x_{i-1} - 2x_i - 2x_{i+1} + 3\,]$

Rule 8 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

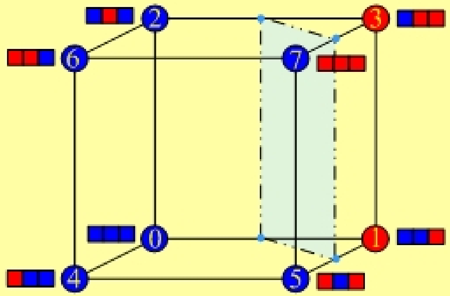
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ -1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = -2$

Formula

$y = \phi\,[\,-2x_{i-1} + 2x_i + 2x_{i+1} - 3\,]$

Rule 10   κ = 1



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

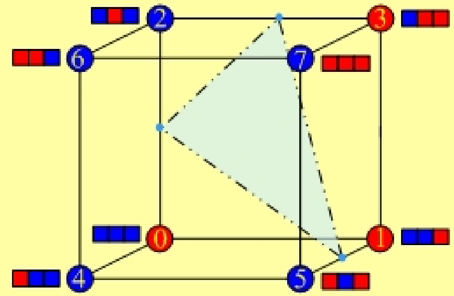
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ -1 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = -1$

Formula

$y = \text{ȼ}[\,-2x_{i-1} + 2x_{i+1} - 1]$

Rule 11   κ = 1



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

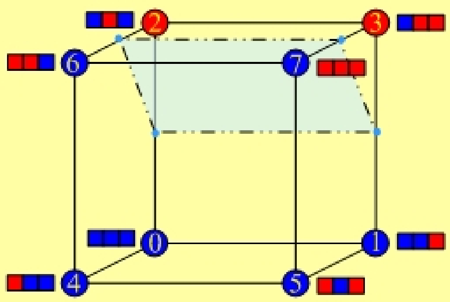
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ -2 & -1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = -1$

Formula

$y = \text{ȼ}[\,-4x_i - 2x_i + 2x_{i+1} + 1]$

Rule 12   κ = 1



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

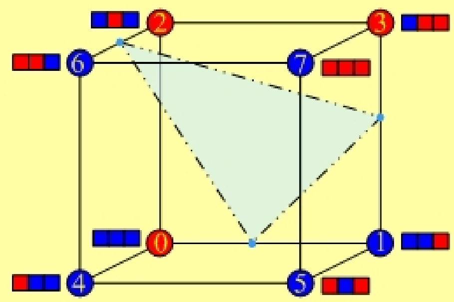
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $z = -1$

Formula

$y = \text{ȼ}[\,-2x_{i-1} + 2x_i - 1]$

Rule 13   κ = 1



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

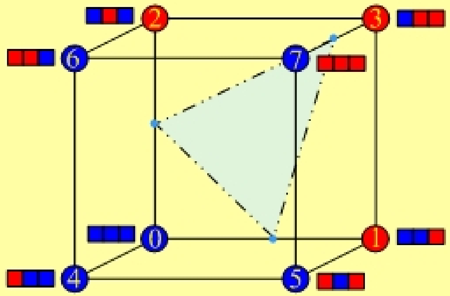
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ -2 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = -1$

Formula

$y = \text{ȼ}[\,-4x_{i-1} + 2x_i - 2x_{i+1} + 1]$

Rule 14 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

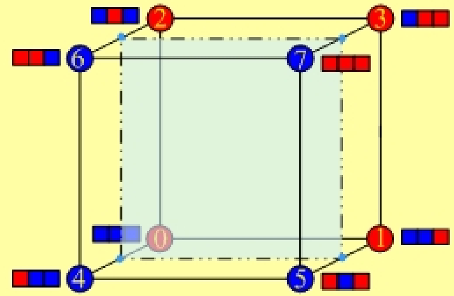
Template

$A = \begin{array}{|c|c|c|}\hline 0&0&0\\\hline 0&1&0\\\hline 0&0&0\\\hline\end{array}$ $B = \begin{array}{|c|c|c|}\hline 0&0&0\\\hline -2&1&1\\\hline 0&0&0\\\hline\end{array}$ $z = -1$

Formula

$y = \phi\,[\,-4x_{i-1} + 2x_i + 2x_{i+1} - 1]$

Rule 15 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

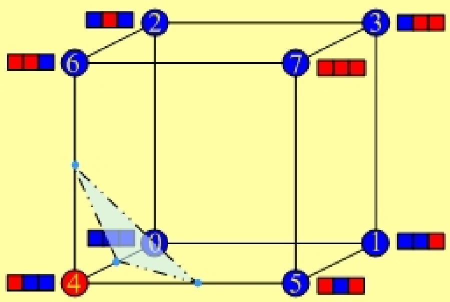
Template

$A = \begin{array}{|c|c|c|}\hline 0&0&0\\\hline 0&1&0\\\hline 0&0&0\\\hline\end{array}$ $B = \begin{array}{|c|c|c|}\hline 0&0&0\\\hline -1&0&0\\\hline 0&0&0\\\hline\end{array}$ $z = 0$

Formula

$y = \phi\,[\,-2x_i + 1]$

Rule 16 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

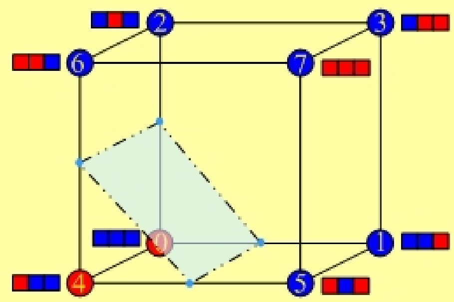
Template

$A = \begin{array}{|c|c|c|}\hline 0&0&0\\\hline 0&1&0\\\hline 0&0&0\\\hline\end{array}$ $B = \begin{array}{|c|c|c|}\hline 0&0&0\\\hline 1&-1&-1\\\hline 0&0&0\\\hline\end{array}$ $z = -2$

Formula

$y = \phi\,[\,2x_{i-1} - 2x_i - 2x_{i+1} - 1]$

Rule 17 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

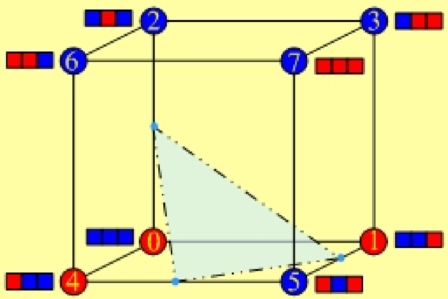
Template

$A = \begin{array}{|c|c|c|}\hline 0&0&0\\\hline 0&1&0\\\hline 0&0&0\\\hline\end{array}$ $B = \begin{array}{|c|c|c|}\hline 0&0&0\\\hline 0&-1&-1\\\hline 0&0&0\\\hline\end{array}$ $z = -1$

Formula

$y = \phi\,[\,-2x_i - 2x_{i+1} + 1]$

Rule 19 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

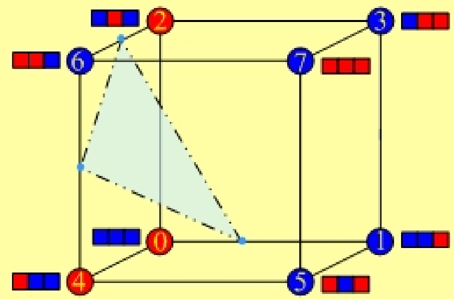
Template

$A = \begin{array}{|c|c|c|}\hline 0&0&0\\\hline 0&1&0\\\hline 0&0&0\\\hline\end{array}$ $B = \begin{array}{|c|c|c|}\hline 0&0&0\\\hline -1&-2&-1\\\hline 0&0&0\\\hline\end{array}$ $z = -1$

Formula

$y = \sigma[-2x_{i-1} - 4x_i - 2x_{i+1} + 3]$

Rule 21 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

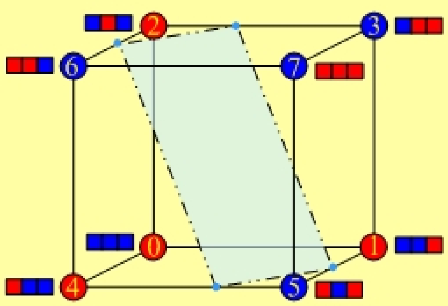
Template

$A = \begin{array}{|c|c|c|}\hline 0&0&0\\\hline 0&1&0\\\hline 0&0&0\\\hline\end{array}$ $B = \begin{array}{|c|c|c|}\hline 0&0&0\\\hline -1&-1&-2\\\hline 0&0&0\\\hline\end{array}$ $z = -1$

Formula

$y = \sigma[-2x_i - 2x_i - 4x_{i+1} + 3]$

Rule 23 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

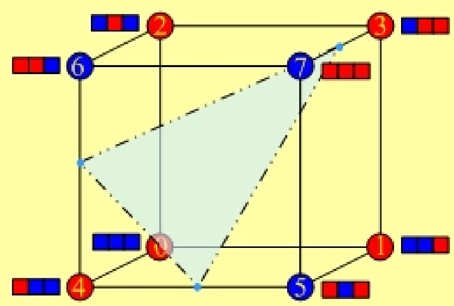
Template

$A = \begin{array}{|c|c|c|}\hline 0&0&0\\\hline 0&1&0\\\hline 0&0&0\\\hline\end{array}$ $B = \begin{array}{|c|c|c|}\hline 0&0&0\\\hline -1&-1&-1\\\hline 0&0&0\\\hline\end{array}$ $z = 0$

Formula

$y = \sigma[-2x_{i-1} - 2x_i - 2x_{i+1} + 3]$

Rule 31 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

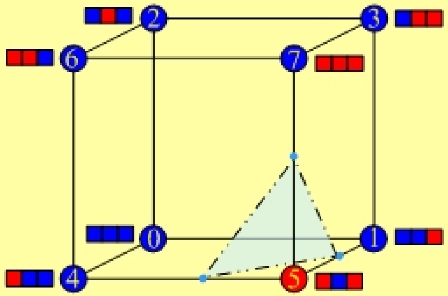
Template

$A = \begin{array}{|c|c|c|}\hline 0&0&0\\\hline 0&1&0\\\hline 0&0&0\\\hline\end{array}$ $B = \begin{array}{|c|c|c|}\hline 0&0&0\\\hline -2&-1&-1\\\hline 0&0&0\\\hline\end{array}$ $z = 1$

Formula

$y = \sigma[-4x_{i-1} - 2x_i - 2x_{i+1} + 5]$

Rule 32 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0

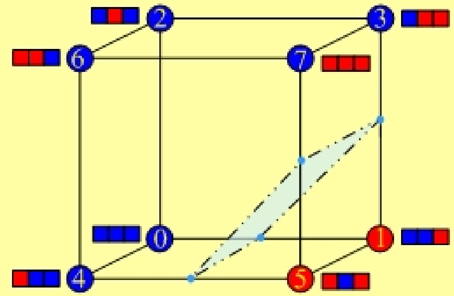
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & -1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = -2$

Formula

$y = \sigma[\,2x_{i-1} - 2x_i + 2x_{i+1} - 3\,]$

Rule 34 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0

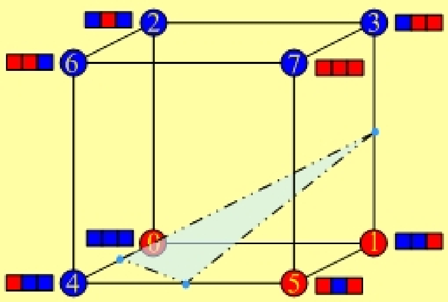
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = -1$

Formula

$y = \sigma[\,-2x_i + 2x_{i+1} - 1\,]$

Rule 35 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0

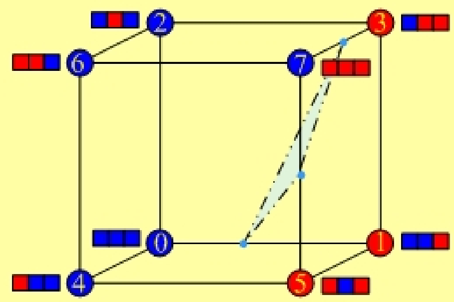
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ -1 & -2 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = -1$

Formula

$y = \sigma[\,-2x_{i-1} - 4x_i + 2x_{i+1} + 1\,]$

Rule 42 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0

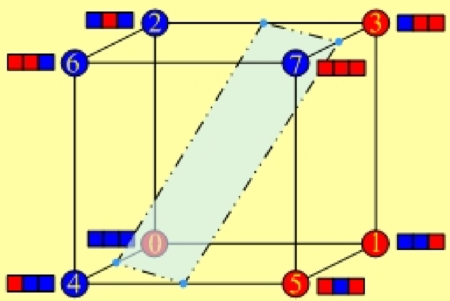
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ -1 & -1 & 2 \\ 0 & 0 & 0 \end{bmatrix}$ $z = -1$

Formula

$y = \sigma[\,-2x_{i-1} - 2x_i + 4x_{i+1} - 1\,]$

Rule 43 — κ = 1



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0

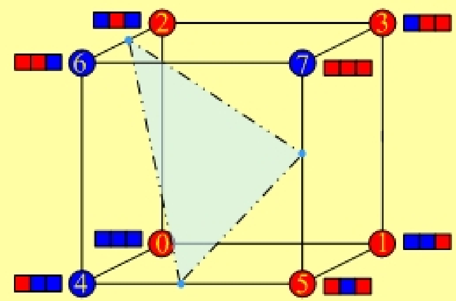
Template

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ -1 & -1 & 1 \\ 0 & 0 & 0 \end{bmatrix} \quad z = 0$$

Formula

$$y = \sigma[\,-2x_{i-1} - 2x_i + 2x_{i+1} + 1]$$

Rule 47 — κ = 1



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0

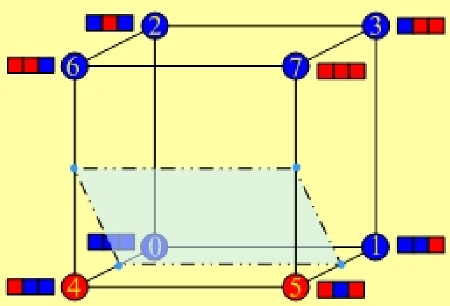
Template

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ -2 & -1 & 1 \\ 0 & 0 & 0 \end{bmatrix} \quad z = 1$$

Formula

$$y = \sigma[\,-4x_{i-1} - 2x_i + 2x_{i+1} + 3]$$

Rule 48 — κ = 1



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

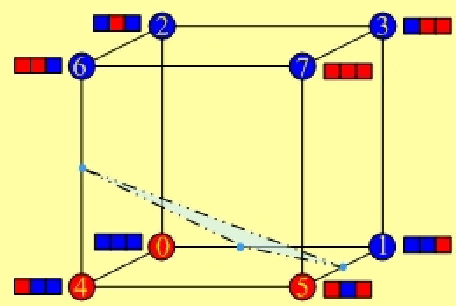
Template

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad z = -1$$

Formula

$$y = \sigma[\,2x_{i-1} - 2x_i - 1]$$

Rule 49 — κ = 1



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

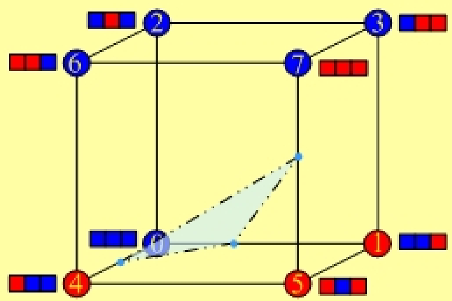
Template

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & -2 & -1 \\ 0 & 0 & 0 \end{bmatrix} \quad z = -1$$

Formula

$$y = \sigma[\,2x_{i-1} - 4x_i - 2x_{i+1} + 1]$$

Rule 50 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

Template

A =

0	0	0
0	1	0
0	0	0

B =

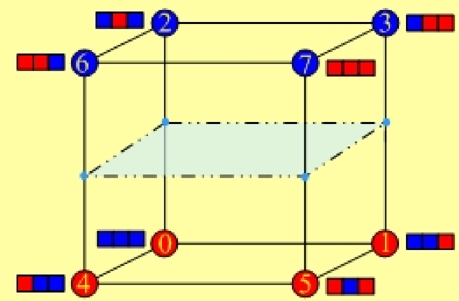
0	0	0
1	-2	1
0	0	0

z = -1

Formula

$y = \sigma\,[\,2x_{i-1} - 4x_i + 2x_{i+1} - 1]$

Rule 51 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

Template

A =

0	0	0
0	1	0
0	0	0

B =

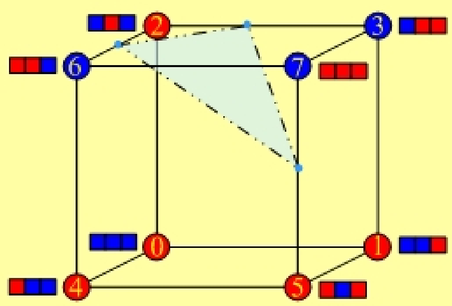
0	0	0
0	-1	0
0	0	0

z = 0

Formula

$y = \sigma\,[\,-2x_i + 1]$

Rule 55 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

Template

A =

0	0	0
0	1	0
0	0	0

B =

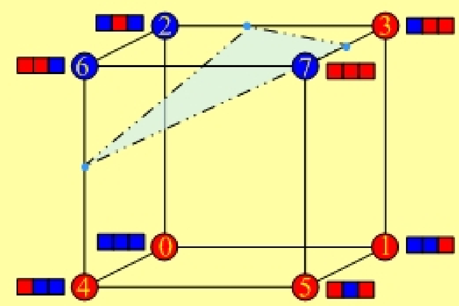
0	0	0
-1	-2	-1
0	0	0

z = 1

Formula

$y = \sigma\,[\,-2x_{i-1} - 4x_i - 2x_{i+1} + 5]$

Rule 59 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

Template

A =

0	0	0
0	1	0
0	0	0

B =

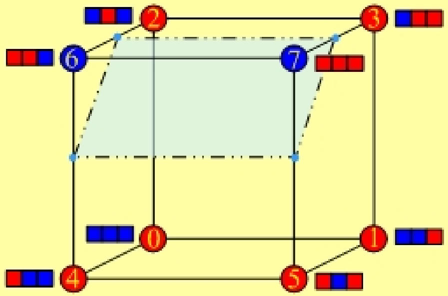
0	0	0
-1	-2	1
0	0	0

z = 1

Formula

$y = \sigma\,[\,-2x_{i-1} - 4x_i + 2x_{i+1} + 3]$

Rule 63 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

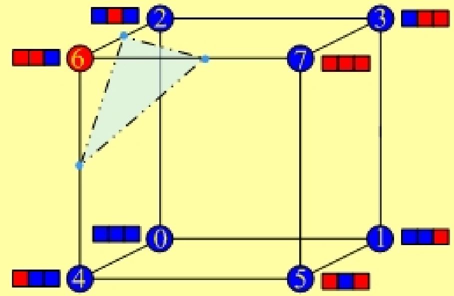
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ -1 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 1$

Formula

$y = \phi\,[\,-2x_{i-1} - 2x_i + 3]$

Rule 64 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

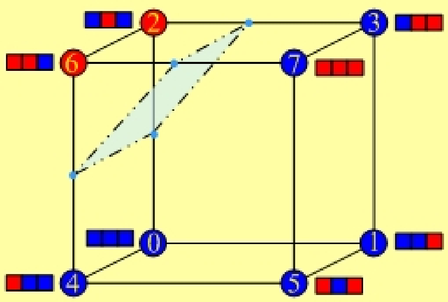
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = -2$

Formula

$y = \phi\,[\,2x_{i-1} + 2x_i - 2x_{i+1} - 3]$

Rule 68 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

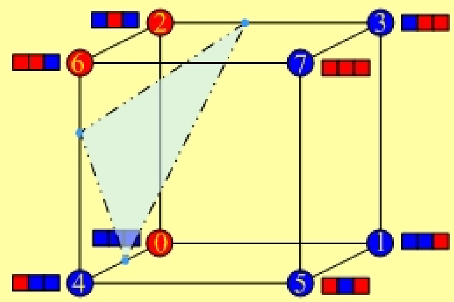
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = -1$

Formula

$y = \phi\,[\,2x_i - 2x_{i+1} - 1]$

Rule 69 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

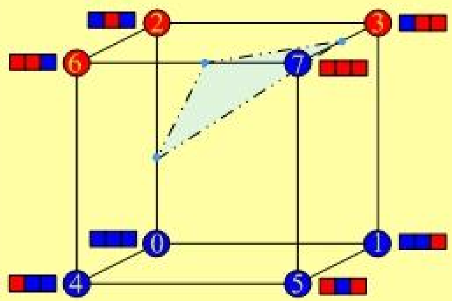
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ -1 & 1 & -2 \\ 0 & 0 & 0 \end{bmatrix}$ $z = -1$

Formula

$y = \phi\,[\,-2x_{i-1} + 2x_i - 4x_{i+1} + 1]$

Rule 76 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

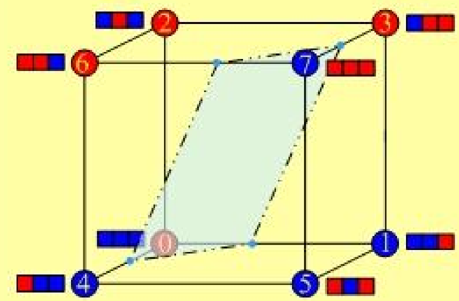
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ -1 & 2 & -1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = -1$

Formula

$y = \partial\,[\,-2x_{i-1} + 4x_i - 2x_{i+1} - 1\,]$

Rule 77 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

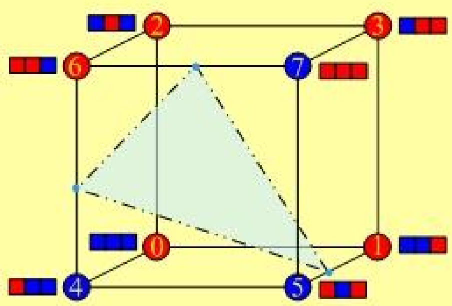
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ -1 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 0$

Formula

$y = \partial\,[\,-2x_{i-1} + 2x_i - 2x_{i+1} + 1\,]$

Rule 79 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

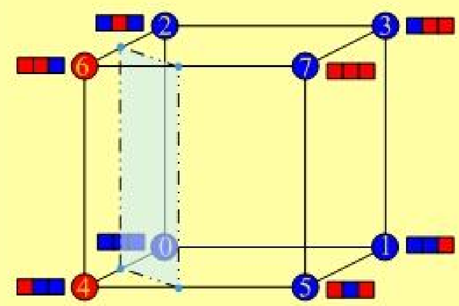
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ -2 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 1$

Formula

$y = \partial\,[\,-4x_{i-1} + 2x_i - 2x_{i+1} + 3\,]$

Rule 80 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

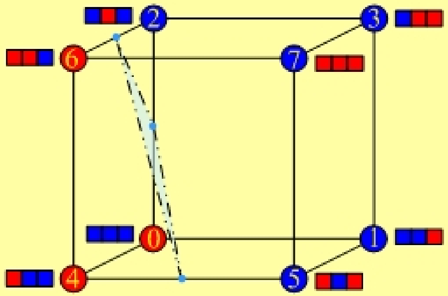
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & -1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = -1$

Formula

$y = \partial\,[\,2x_{i-1} - 2x_{i+1} - 1\,]$

Rule 81 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

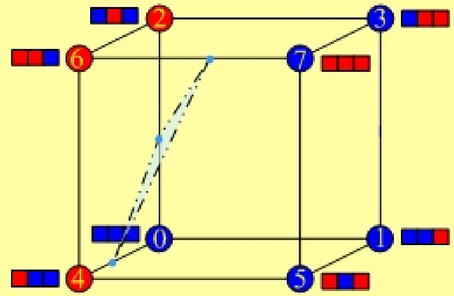
Template

$A = \begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 0 & 1 & 0 \\ \hline 0 & 0 & 0 \\ \hline\end{array}$ $B = \begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 1 & -1 & -2 \\ \hline 0 & 0 & 0 \\ \hline\end{array}$ $z = -1$

Formula

$y = \sigma\,[\,2x_{i-1} - 2x_i - 4x_{i+1} + 1\,]$

Rule 84 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

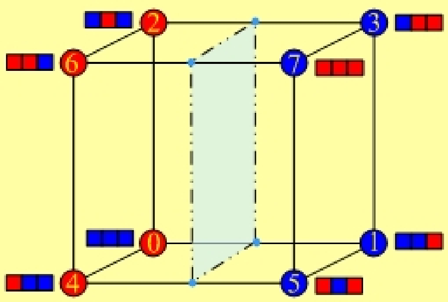
Template

$A = \begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 0 & 1 & 0 \\ \hline 0 & 0 & 0 \\ \hline\end{array}$ $B = \begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 1 & 1 & -2 \\ \hline 0 & 0 & 0 \\ \hline\end{array}$ $z = -1$

Formula

$y = \sigma\,[\,2x_{i-1} + 2x_i - 4x_{i+1} - 1\,]$

Rule 85 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

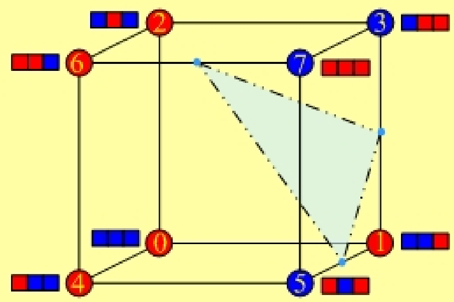
Template

$A = \begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 0 & 1 & 0 \\ \hline 0 & 0 & 0 \\ \hline\end{array}$ $B = \begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 0 & 0 & -1 \\ \hline 0 & 0 & 0 \\ \hline\end{array}$ $z = 0$

Formula

$y = \sigma\,[\,-2x_{i+1} + 1\,]$

Rule 87 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

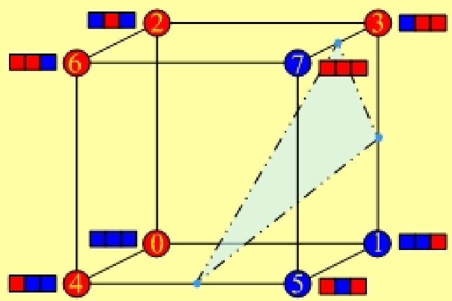
Template

$A = \begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 0 & 1 & 0 \\ \hline 0 & 0 & 0 \\ \hline\end{array}$ $B = \begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline -1 & -1 & -2 \\ \hline 0 & 0 & 0 \\ \hline\end{array}$ $z = 1$

Formula

$y = \sigma\,[\,-2x_{i-1} - 2x_i - 4x_{i+1} + 5\,]$

Rule 93 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

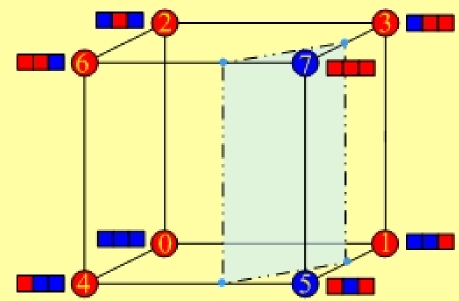
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ -1 & 1 & -2 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 1$

Formula

$y = \sigma[-2x_{i-1} + 2x_i - 4x_{i+1} + 3]$

Rule 95 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

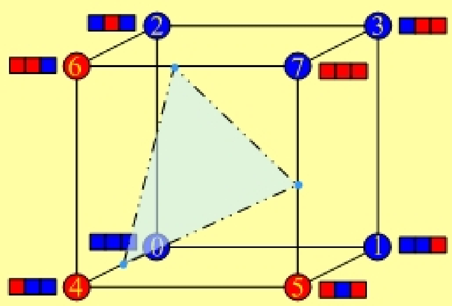
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ -1 & 0 & -1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 1$

Formula

$y = \sigma[-2x_{i-1} - 2x_{i+1} + 3]$

Rule 112 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

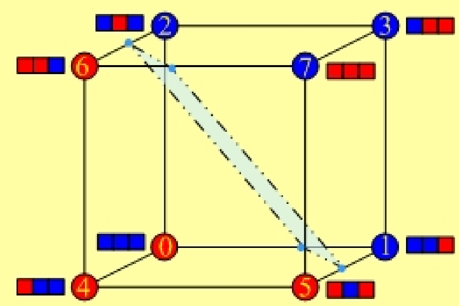
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 2 & -1 & -1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = -1$

Formula

$y = \sigma[4x_{i-1} - 2x_i - 2x_{i+1} - 1]$

Rule 113 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

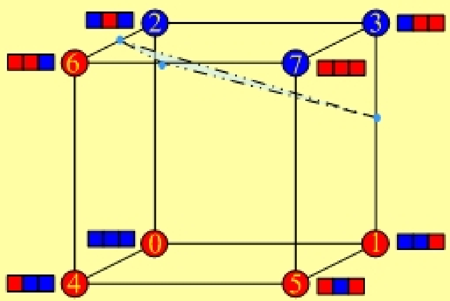
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & -1 & -1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 0$

Formula

$y = \sigma[2x_{i-1} - 2x_i - 2x_{i+1} + 1]$

Rule 115 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

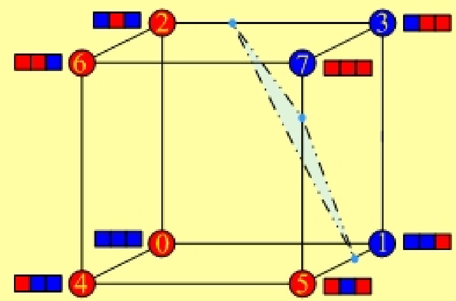
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & -2 & -1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 1$

Formula

$y = \sigma[\,2x_{i-1} - 4x_i - 2x_{i+1} + 3\,]$

Rule 117 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

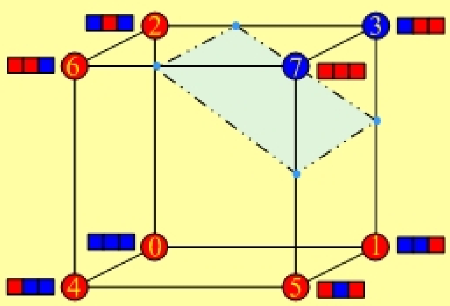
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & -1 & -2 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 1$

Formula

$y = \sigma[\,2x_{i-1} - 2x_i - 4x_{i+1} + 3\,]$

Rule 119 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

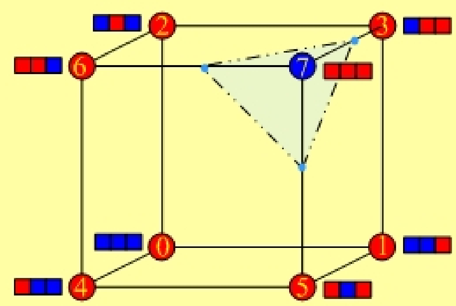
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -1 & -1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 1$

Formula

$y = \sigma[\,-2x_i - 2x_{i+1} + 3\,]$

Rule 127 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

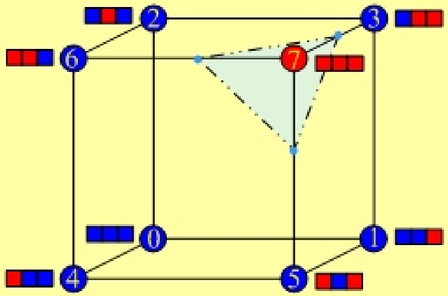
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ -1 & -1 & -1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 2$

Formula

$y = \sigma[\,-2x_{i-1} - 2x_i - 2x_{i+1} + 5\,]$

Rule 128 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

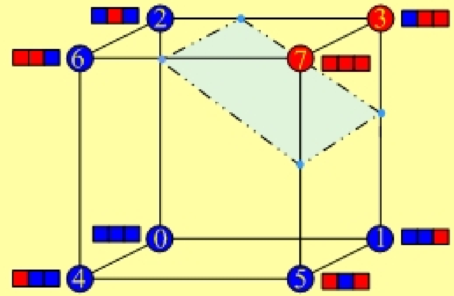
Template

$A = \begin{array}{|c|c|c|}\hline 0&0&0\\ \hline 0&1&0\\ \hline 0&0&0\\ \hline\end{array}$ $B = \begin{array}{|c|c|c|}\hline 0&0&0\\ \hline 1&1&1\\ \hline 0&0&0\\ \hline\end{array}$ $z = -2$

Formula

$y = \sigma\,[\,2x_{i-1} + 2x_i + 2x_{i+1} - 5]$

Rule 136 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

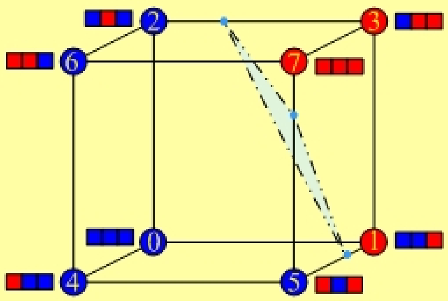
Template

$A = \begin{array}{|c|c|c|}\hline 0&0&0\\ \hline 0&1&0\\ \hline 0&0&0\\ \hline\end{array}$ $B = \begin{array}{|c|c|c|}\hline 0&0&0\\ \hline 0&1&1\\ \hline 0&0&0\\ \hline\end{array}$ $z = -1$

Formula

$y = \sigma\,[\,2x_i + 2x_{i+1} - 3]$

Rule 138 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

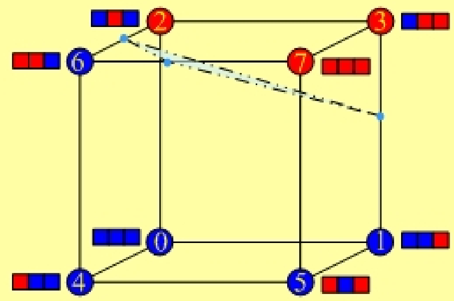
Template

$A = \begin{array}{|c|c|c|}\hline 0&0&0\\ \hline 0&1&0\\ \hline 0&0&0\\ \hline\end{array}$ $B = \begin{array}{|c|c|c|}\hline 0&0&0\\ \hline -1&1&2\\ \hline 0&0&0\\ \hline\end{array}$ $z = -1$

Formula

$y = \sigma\,[\,-2x_{i-1} + 2x_i + 4x_{i+1} - 3]$

Rule 140 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

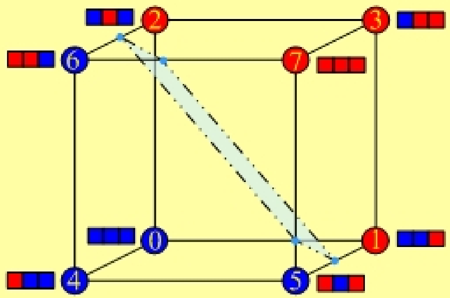
Template

$A = \begin{array}{|c|c|c|}\hline 0&0&0\\ \hline 0&1&0\\ \hline 0&0&0\\ \hline\end{array}$ $B = \begin{array}{|c|c|c|}\hline 0&0&0\\ \hline -1&2&1\\ \hline 0&0&0\\ \hline\end{array}$ $z = -1$

Formula

$y = \sigma\,[\,-2x_{i-1} + 4x_i + 2x_{i+1} - 3]$

Rule 142 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

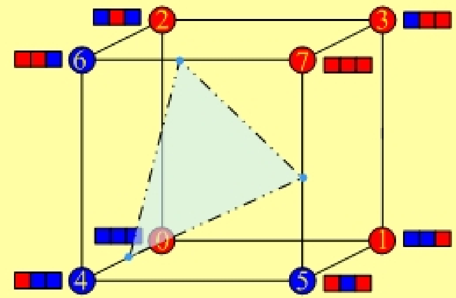
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ -1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 0$

Formula

$y = \sigma[\, -2x_{i-1} + 2x_i + 2x_{i+1} - 1]$

Rule 143 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

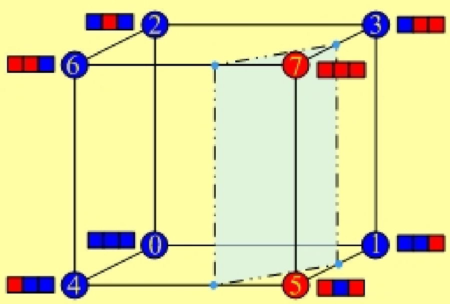
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ -2 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 1$

Formula

$y = \sigma[\, -4x_{i-1} + 2x_i + 2x_{i+1} + 1]$

Rule 160 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

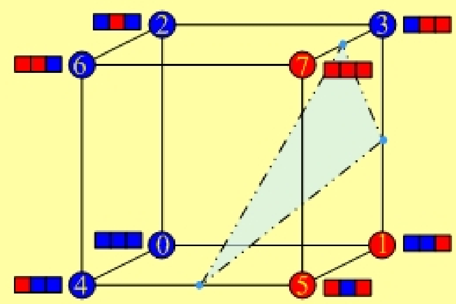
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = -1$

Formula

$y = \sigma[\, 2x_{i-1} + 2x_{i+1} - 3]$

Rule 162 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

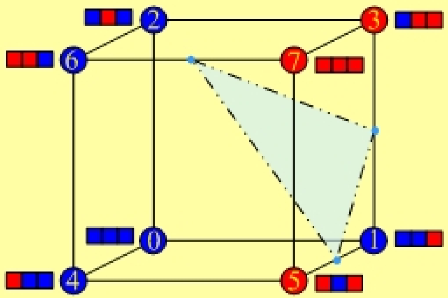
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & -1 & 2 \\ 0 & 0 & 0 \end{bmatrix}$ $z = -1$

Formula

$y = \sigma[\, 2x_{i-1} - 2x_i + 4x_{i+1} - 3]$

Rule 168 $\quad \kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

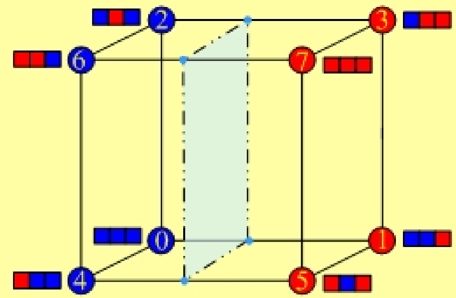
Template

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 2 \\ 0 & 0 & 0 \end{bmatrix} \quad z = -1$$

Formula

$$y = \phi\,[\,2x_{i-1} + 2x_i + 4x_{i+1} - 5\,]$$

Rule 170 $\quad \kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

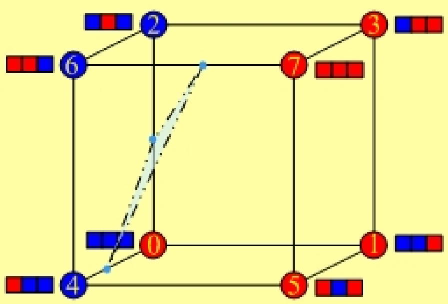
Template

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \quad z = 0$$

Formula

$$y = \phi\,[\,2x_{i+1} - 1\,]$$

Rule 171 $\quad \kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

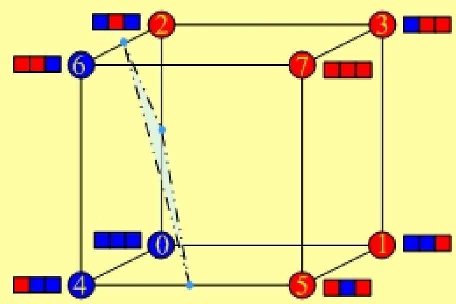
Template

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ -1 & -1 & 2 \\ 0 & 0 & 0 \end{bmatrix} \quad z = 1$$

Formula

$$y = \phi\,[\,-2x_{i-1} - 2x_i + 4x_{i+1} + 1\,]$$

Rule 174 $\quad \kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

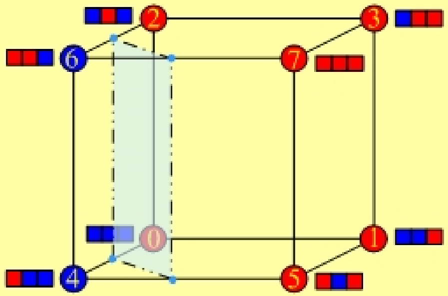
Template

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ -1 & 1 & 2 \\ 0 & 0 & 0 \end{bmatrix} \quad z = 1$$

Formula

$$y = \phi\,[\,-2x_{i-1} + 2x_i + 4x_{i+1} - 1\,]$$

Rule 175 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

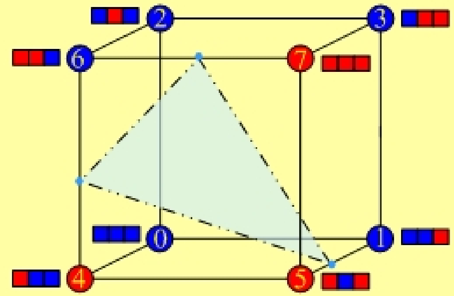
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ -1 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 1$

Formula

$y = \phi\,[\,-2x_{i-1} + 2x_{i+1} + 1]$

Rule 176 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

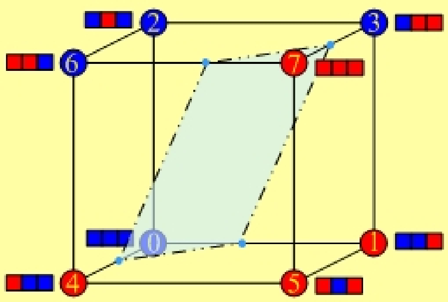
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 2 & -1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = -1$

Formula

$y = \phi\,[\,4x_{i-1} - 2x_i + 2x_{i+1} - 3]$

Rule 178 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

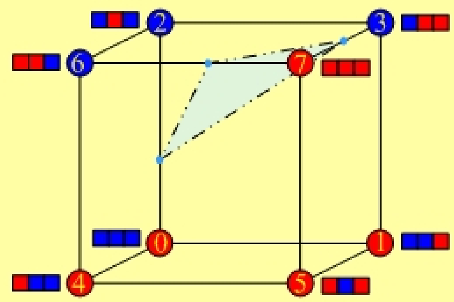
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & -1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 0$

Formula

$y = \phi\,[\,2x_{i-1} - 2x_i + 2x_{i+1} - 1]$

Rule 179 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

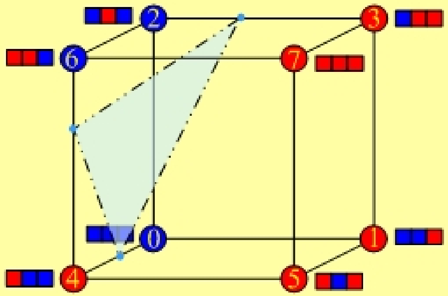
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & -2 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 1$

Formula

$y = \phi\,[\,2x_{i-1} - 4x_i + 2x_{i+1} + 1]$

Rule 186 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

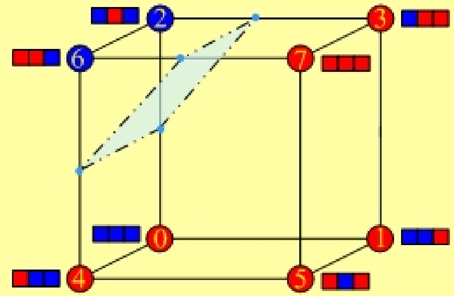
Template

$A = \begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 0 & 1 & 0 \\ \hline 0 & 0 & 0 \\ \hline\end{array}$ $B = \begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 1 & -1 & 2 \\ \hline 0 & 0 & 0 \\ \hline\end{array}$ $z = 1$

Formula

$y = \phi\,[\, 2x_{i-1} - 2x_i + 4x_{i+1} - 1]$

Rule 187 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

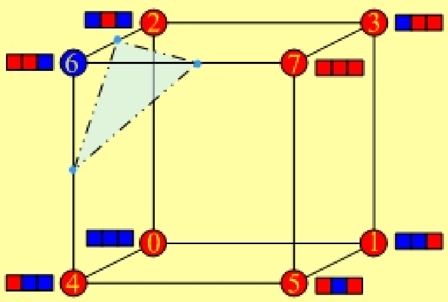
Template

$A = \begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 0 & 1 & 0 \\ \hline 0 & 0 & 0 \\ \hline\end{array}$ $B = \begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 0 & -1 & 1 \\ \hline 0 & 0 & 0 \\ \hline\end{array}$ $z = 1$

Formula

$y = \phi\,[\, -2x_i + 2x_{i+1} + 1]$

Rule 191 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

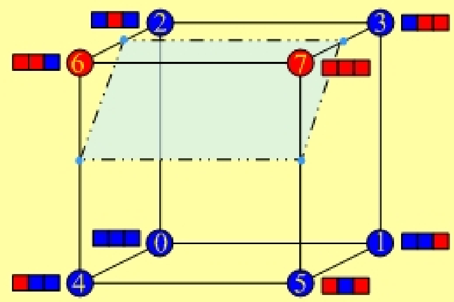
Template

$A = \begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 0 & 1 & 0 \\ \hline 0 & 0 & 0 \\ \hline\end{array}$ $B = \begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline -1 & -1 & 1 \\ \hline 0 & 0 & 0 \\ \hline\end{array}$ $z = 2$

Formula

$y = \phi\,[\, -2x_{i-1} - 2x_i + 2x_{i+1} + 3]$

Rule 192 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

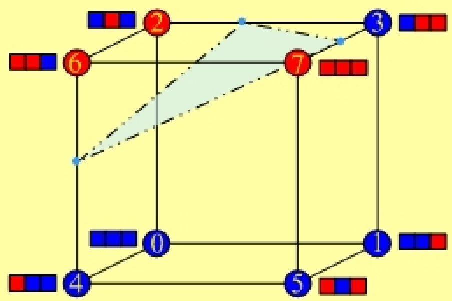
Template

$A = \begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 0 & 1 & 0 \\ \hline 0 & 0 & 0 \\ \hline\end{array}$ $B = \begin{array}{|c|c|c|}\hline 0 & 0 & 0 \\ \hline 1 & 1 & 0 \\ \hline 0 & 0 & 0 \\ \hline\end{array}$ $z = -1$

Formula

$y = \phi\,[\, 2x_{i-1} + 2x_i - 3]$

Rule 196 — $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

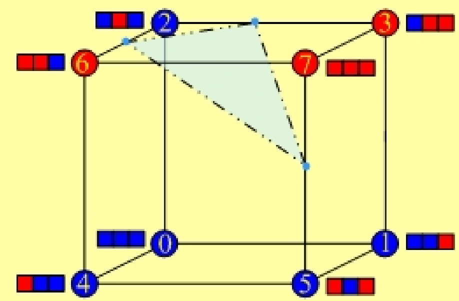
Template

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 2 & -1 \\ 0 & 0 & 0 \end{bmatrix} \quad z = -1$$

Formula

$$y = \sigma[\,2x_{i-1} + 4x_i - 2x_{i+1} - 3\,]$$

Rule 200 — $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

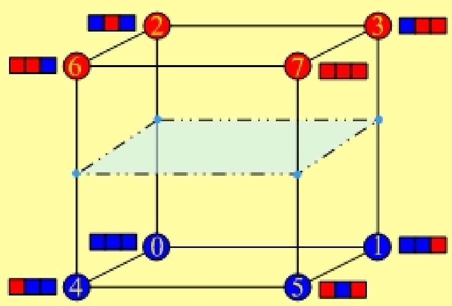
Template

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 2 & 1 \\ 0 & 0 & 0 \end{bmatrix} \quad z = -1$$

Formula

$$y = \sigma[\,2x_{i-1} + 4x_i + 2x_{i+1} - 5\,]$$

Rule 204 — $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

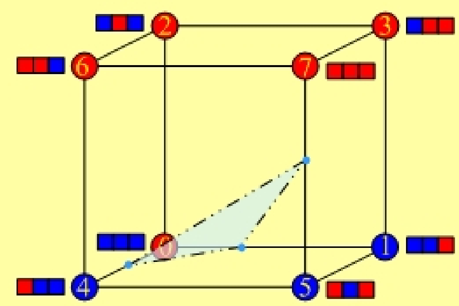
Template

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad z = 0$$

Formula

$$y = \sigma[\,2x_i - 1\,]$$

Rule 205 — $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

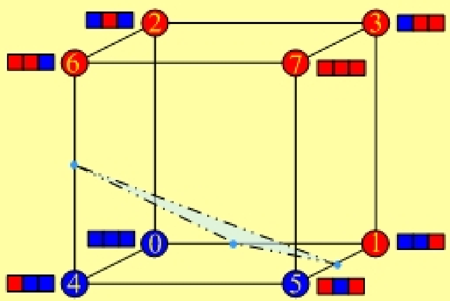
Template

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ -1 & 2 & -1 \\ 0 & 0 & 0 \end{bmatrix} \quad z = 1$$

Formula

$$y = \sigma[\,-2x_{i-1} + 4x_i - 2x_{i+1} + 1\,]$$

Rule 206 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

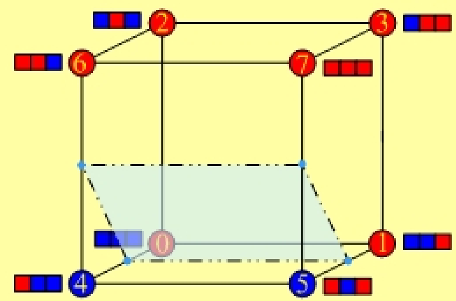
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ -1 & 2 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 1$

Formula

$y = \sigma[\,-2x_{i-1} + 4x_i + 2x_{i+1} - 1\,]$

Rule 207 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

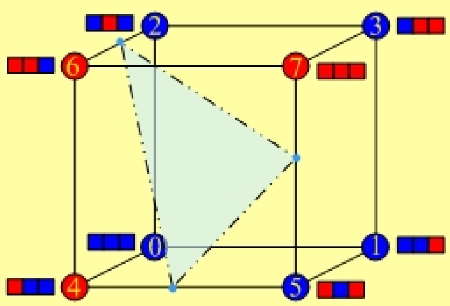
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 1$

Formula

$y = \sigma[\,-2x_{i-1} + 2x_i + 1\,]$

Rule 208 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

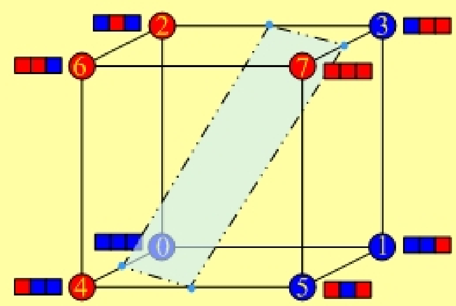
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 2 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = -1$

Formula

$y = \sigma[\,4x_{i-1} + 2x_i - 2x_{i+1} - 3\,]$

Rule 212 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

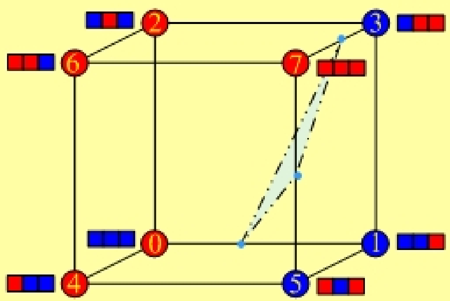
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 0$

Formula

$y = \sigma[\,2x_{i-1} + 2x_i - 2x_{i+1} - 1\,]$

Rule 213 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

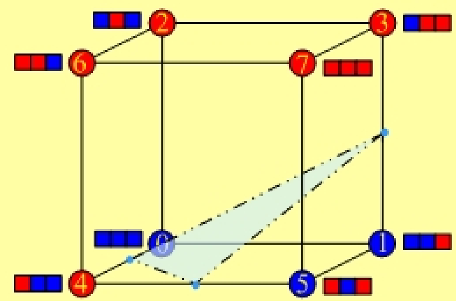
Template

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & -2 \\ 0 & 0 & 0 \end{bmatrix} \quad z = 1$$

Formula

$$y = \sigma[\, 2x_{i-1} + 2x_i - 4x_{i+1} + 1]$$

Rule 220 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

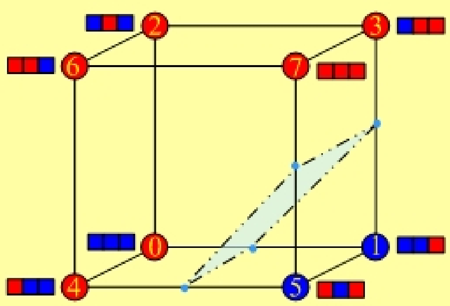
Template

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 2 & -1 \\ 0 & 0 & 0 \end{bmatrix} \quad z = 1$$

Formula

$$y = \sigma[\, 2x_{i-1} + 4x_i - 2x_{i+1} - 1]$$

Rule 221 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

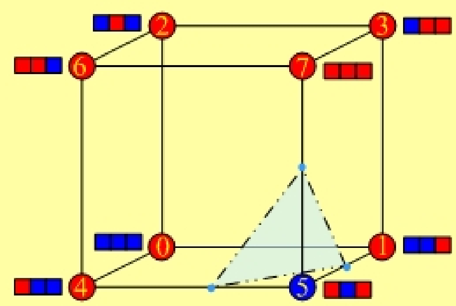
Template

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix} \quad z = 1$$

Formula

$$y = \sigma[\, 2x_i - 2x_{i+1} + 1]$$

Rule 223 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

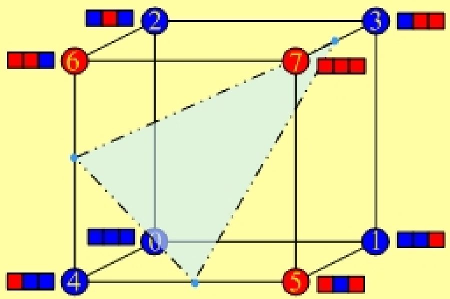
Template

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 0 & 0 \\ -1 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix} \quad z = 2$$

Formula

$$y = \sigma[\, -2x_{i-1} + 2x_i - 2x_{i+1} + 3]$$

Rule 224 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

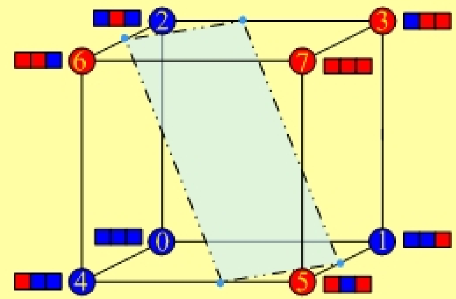
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 2 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = -1$

Formula

$y = \sigma\,[\,4x_{i-1} + 2x_i + 2x_{i+1} - 5\,]$

Rule 232 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

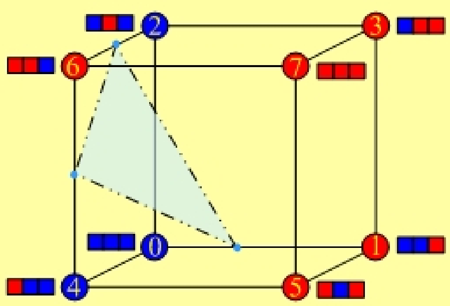
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 0$

Formula

$y = \sigma\,[\,2x_{i-1} + 2x_i + 2x_{i+1} - 3\,]$

Rule 234 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

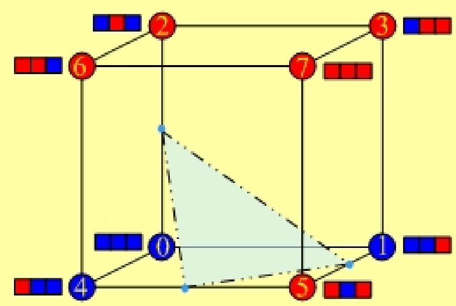
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 2 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 1$

Formula

$y = \sigma\,[\,2x_{i-1} + 2xi + 4x_{i+1} - 3\,]$

Rule 236 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

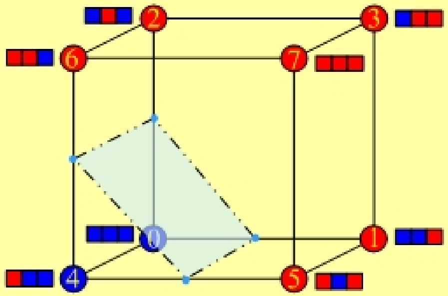
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 2 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 1$

Formula

$y = \sigma\,[\,2x_{i-1} + 4x_i + 2x_{i+1} - 3\,]$

Rule 238 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

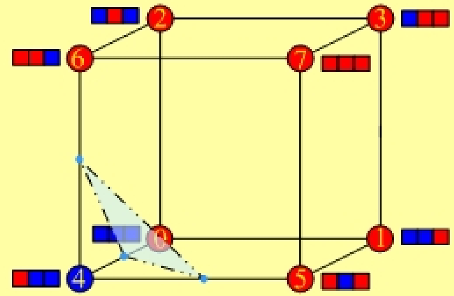
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 1$

Formula

$y = \sigma[\, 2x_i + 2x_{i+1} - 1 \,]$

Rule 239 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

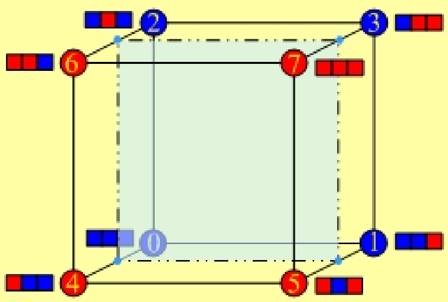
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ -1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 2$

Formula

$y = \sigma[\, -2x_{i-1} + 2x_i + 2x_{i+1} + 1 \,]$

Rule 240 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

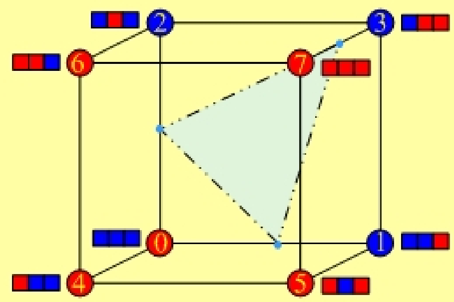
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 0$

Formula

$y = \sigma[\, 2x_{i-1} - 1 \,]$

Rule 241 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

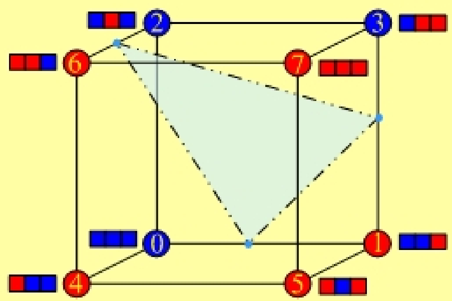
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 2 & -1 & -1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 1$

Formula

$y = \sigma[\, 4x_{i-1} - 2x_i - 2x_{i+1} + 1 \,]$

Rule 242 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

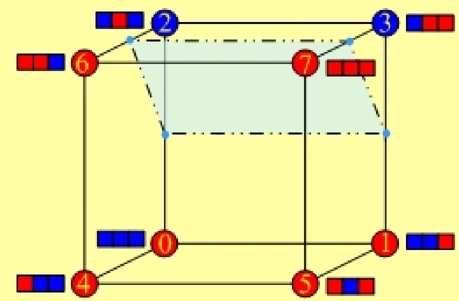
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 2 & -1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 1$

Formula

$y = \sigma[\, 4x_{i-1} - 2x_i + 2x_{i+1} - 1\,]$

Rule 243 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

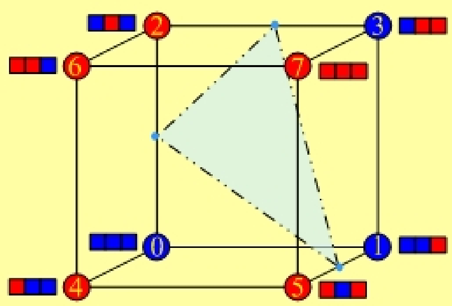
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 1$

Formula

$y = \sigma[\, 2x_{i-1} - 2x_i + 1\,]$

Rule 244 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

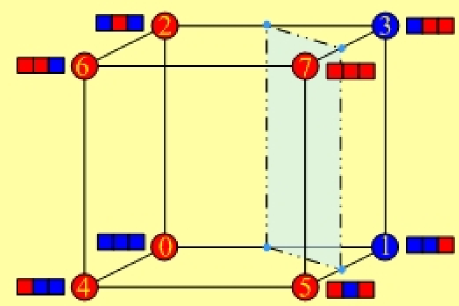
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 2 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 1$

Formula

$y = \sigma[\, 4x_{i-1} + 2x_i - 2x_{i+1} - 1\,]$

Rule 245 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

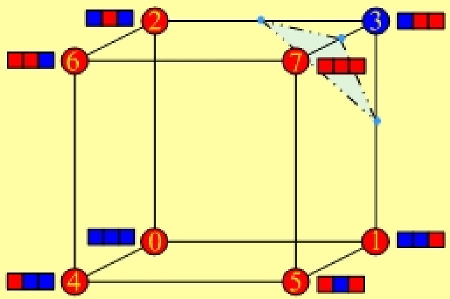
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & -1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 1$

Formula

$y = \sigma[\, 2x_{i-1} - 2x_{i+1} + 1\,]$

Rule 247 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

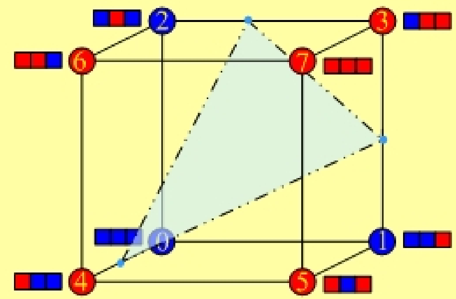
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & -1 & -1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 2$

Formula

$y = \sigma[\,2x_{i-1} - 2x_i - 2x_{i+1} + 3\,]$

Rule 248 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

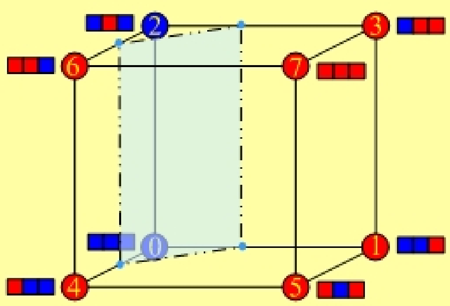
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 2 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 1$

Formula

$y = \sigma[\,4x_{i-1} + 2x_i + 2x_{i+1} - 3\,]$

Rule 250 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

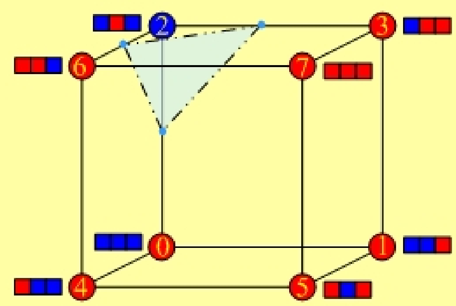
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 1$

Formula

$y = \sigma[\,2x_{i-1} + 2x_{i+1} - 1\,]$

Rule 251 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

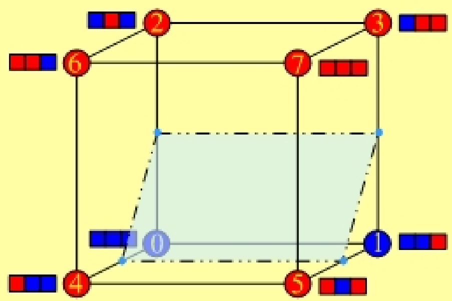
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & -1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 2$

Formula

$y = \sigma[\,2x_{i-1} - 2x_i + 2x_{i+1} + 1\,]$

Rule 252 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

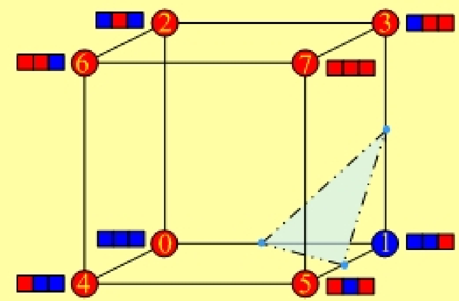
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 1$

Formula

$y = \sigma\,[\,2x_{i-1} + 2x_i - 1\,]$

Rule 253 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

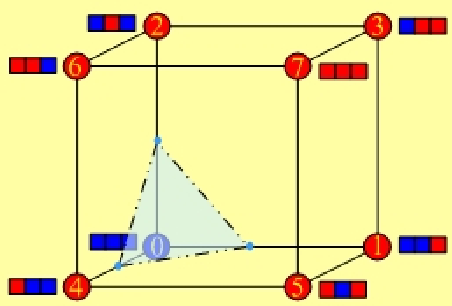
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 2$

Formula

$y = \sigma\,[\,2x_{i-1} + 2x_i - 2x_{i+1} + 1\,]$

Rule 254 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

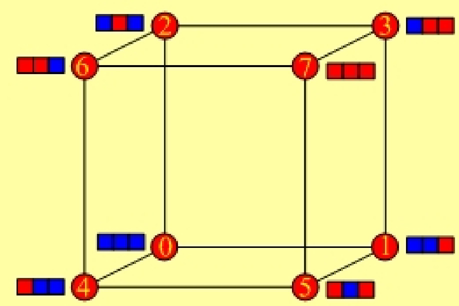
Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 2$

Formula

$y = \sigma\,[\,2x_{i-1} + 2x_i + 2x_{i+1} - 1\,]$

Rule 255 $\kappa = 1$



x_{i-1}	x_i	x_{i+1}	y_i
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

Template

$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$ $z = 1$

Formula

$y = 1$

3 Concluding Remarks

After compiling a library of optimal CNN templates for linearly-separable 1-D CA, a next step would be to optimize the current library for 2-D CA. Interested readers may browse the Appendix to use the algorithm to implement the optimal CNN templates for linearly-separable 1-D CA, and then move on to investigating 2-D CA. In the process of compiling the optimal CNN templates for linearly-separable 1-D CA, differences between the optimal CNN templates and those provided in [Chua, et al, 2002] may be noted. In particular, the templates for Rule 63 are incorrect, and should be revised as follows: [b_1, b_2, b_3] should equal to [-1, -1, 0] instead of [0, -1, 0]. To check the validity of the templates generated by the algorithm, the CANDY simulator was run and the dynamic firing patterns were compared.

4 Acknowledgements

Many thanks to Ram Rajagopal for helping with SVM concepts and Akos Zarandi for showing us how to use CANDY to simulate CA.

References

Anafocus Ltd. [2007] The Q-Eye CMOS Focal-Plane Image Sensor-Processor (visual microprocessor). http://www.anafocus.com.

Chua, L. O. & Roska, T. [2002] *Cellular Neural Networks and Visual Computing: Foundations and Applications* (Cambridge University Press).

Chua, L.O., Yoon, S., & Dogaru, R. [2002] "A Nonlinear Dynamics Perspective of Wolfram's New Kind of Science. Part I: Threshold of Complexity," *Int. J. of Bifurcation and Chaos* **12**(12), 2655-2766.

Chua, L.O., Sbitnev, V.I., & Yoon, S. [2003] "A Nonlinear Dynamics Perspective of Wolfram's New Kind of Science. Part II: Universal Neuron," *Int. J. of Bifurcation and Chaos*, **13**(9), 2377-2491.

Dogaru, R. & Chua, L. O. [1998] "CNN Genes for one-dimensional cellular automata: A multi-nested piecewise-linear approach," *Int. J. Bifurcation and Chaos* **8**(10), 1987-2001.

Moore, A. [2007] Support Vector Machines. http://www.autonlab.org/tutorials/svm15.pdf

Schwaighofer, A. [2002] SVM toolbox for MATLAB.
http://ida.first.fraunhofer.de/~anton/software.html

Wolfram, S. [2002] *A New Kind of Science* (Wolfram Media, Inc., Champaign Illinois, USA).

Appendix

In order to find the optimal separating plane, we use the concept of Support Vector Machines [Moore, 2007]. Often we are interested in classifying data. These data points may not necessarily be points in $\Re^2$ but may be multidimensional $\Re^n$ points. We are interested in whether we can separate them by a n-1 dimensional hyperplane. This is a typical form of linear classifier. There are many linear classifiers that might satisfy this property. However, we are additionally interested to find out if we can achieve maximum separation (margin) between the two classes. Now, if such a hyperplane exists, the hyperplane is clearly of interest and is known as the maximum-margin hyperplane.

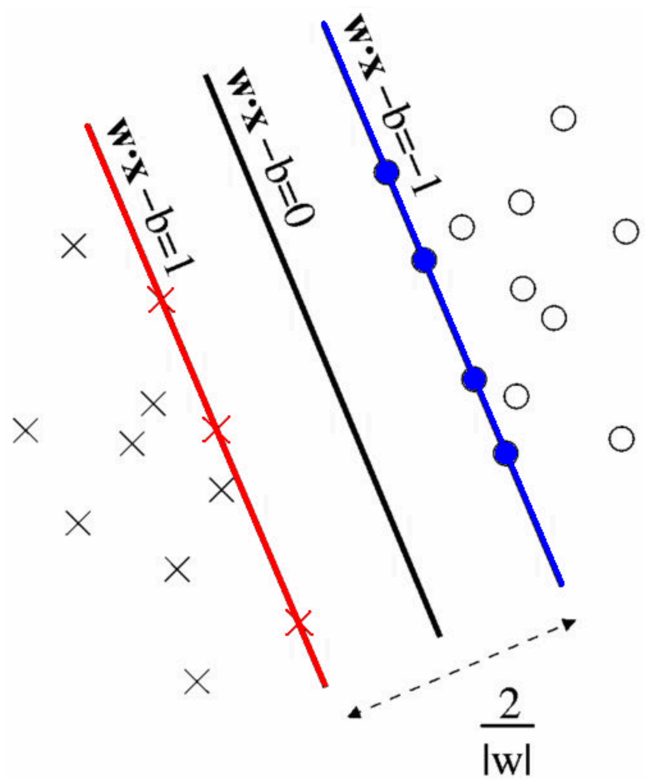


Figure 12. Maximum-margin hyperplanes for a SVM trained with samples from two classes. Samples along the hyperplanes are called support vectors

Consider Figure 12. The goal is to separate the "x"s from the "o"s[3] using a hyperplane that is at maximum distance between the two classes. We can consider the data points to be of the form:

$$\{(x_1, c_1), (x_2, c_2), \ldots, (x_n, c_n)\}$$

[3] In our case, the "x"s could be the red vertices and the "o"s could be the blue vertices. We have thus colored the "x"s and "o"s that would give rise to the support vectors.

Here the c_i is either 1 or -1. This constant denotes the class to which point x_i belongs to (for instance, if the point is an "x" then c_i is 1 and if the point is a "o" then c_i is -1). Each x_i is a n-dimensional real vector, usually of scaled [0,1] or [-1,1] values. Now, the dividing hyperplane takes the form:

$$w \cdot x - b = 0$$

The vector w points perpendicular to the separating hyperplane. Adding the offset parameter b allows us to increase the margin. In its absence, the hyperplane is forced to pass through the origin, restricting the solution. As we are interested in the maximum margin, we are interested in the support vectors and the parallel hyperplanes closest to these support vectors in either class, refer to Figure 12 It can be shown that these hyperplanes can be described by equations:

$$w \cdot x - b = 1,$$
$$w \cdot x - b = -1$$

In our case, the points are linearly separable. Therefore we can select the hyperplanes so that there are no points between them and then try to maximize their distance. By using geometry, we find the distance between the hyperplanes is $\frac{2}{|w|}$, so we want to minimize $|w|$. To exclude data points, we need to ensure that for all i either:

$$w \cdot x_i - b \geq 1, \quad or$$
$$w \cdot x_i - b \leq -1$$

This can be rewritten as:

$$c_i(w \cdot x_i - b) \geq 1, \quad 1 \leq i \leq n \qquad \textbf{(1)}$$

The problem now is to minimize $|w|$ subject to the constraint in **(1)**. That is:

Minimize $|w|$ subject to $c_i(w \cdot x_i - b) \geq 1, \quad 1 \leq i \leq n$

The equation above can be solved using a mathematical package. We use the MATLAB toolbox from [Schwaighofer, 2002] (you need to have the Optimzation toolbox from Mathworks). First download and unzip the toolbox from [Schwaighofer, 2002]. Here are MATLAB commands to find the normal vector and the offset of the optimal-separating plane for Rule 95. The first line sets up the input vector, the second line the output vector corresponding to rule 95. The last two lines use the SVM toolbox to obtain the normal vector w and the offset b for the optimal separating plane.

```
>> U = [-1 -1 -1;-1 -1 1;-1 1 -1; ...
-1 1 1;1 -1 -1;1 -1 1;1 1 -1;1 1 1];
>> Y95 = [1;1;1;1;1;-1;1;-1];
>> net_setup = svm(3,'linear',[],10);
>> net95 = svmtrain(net_setup,U,Y95)
```