Boolean networks and polynomial dynamical systems

Matthew Macauley

Department of Mathematical Sciences
Clemson University
http://www.math.clemson.edu/~macaule/

Math 4500, inputterm

Boolean functions

Let $\mathbb{F}_2 = \{0, 1\}$. By a Boolean function, we usually mean a function $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$.

There are several standard ways to write Boolean functions:

1. As a logical expression, using $\wedge$, $\vee$, and $\neg$ (or $\overline{\ }$)
2. As a polynomial, using $+$, and $\cdot$.
3. As a truth table.

Example

The following are three different ways to express the function that outputs 0 if $x = y = z = 1$, and 1 otherwise.

- $f(x, y, z) = \overline{x \wedge y \wedge z}$
- $f(x, y, z) = 1 + xyz$
-

x	1	1	1	1	0	0	0	0
y	1	1	0	0	1	1	0	0
z	1	0	1	0	1	0	1	0
$f(x, y, z)$	0	1	1	1	1	1	1	1

By counting the number of truth tables, there are $2^{(2^n)}$ n-variable Boolean functions.

Boolean algebra

Boolean operation	logical form	polynomial form
AND	$z = x \wedge y$	$z = xy$
OR	$z = x \vee y$	$z = x + y + xy$
NOT	$z = \overline{x}$	$z = 1 + x$
XOR	$z = x \oplus y = (x \wedge \overline{y}) \vee (\overline{x} \wedge y)$	$z = x + y$

We rarely use XOR. Other Boolean operaions such as NAND, NOR, and XNOR exist but are seldom used.

Over $\mathbb{F}_2$, we have identities such as $x^2 = x$, and $x(1 + x) = 0$.

Theorem

Every Boolean function $f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2$ is a polynomial in the quotient ring $\mathbb{F}_2[x_1, \ldots, x_n]/I$, where $I = \langle x_1^2 - x_1, \ldots, x_n^2 - x_n \rangle$.

Proposition

There are $2^{(2^n)}$ Boolean functions on n variables.

Proof 1: Count the number of truth tables. □

Proof 2: Since $x_i^2 = x_i$, there are 2^n monomials in $x_1, \ldots, x_n$. Every Boolean function is uniquely determined by a subset of these. □

Boolean networks

A Boolean network (BN) on n nodes is an n-tuple $f = (f_1, \ldots, f_n)$ of Boolean functions, where $f_i : \mathbb{F}_2^n \to \mathbb{F}_2$. This defines a map

$$f : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^n, \qquad x = (x_1, \ldots, x_n) \longmapsto \big(f_1(x), \ldots, f_n(x)\big).$$

Any function from a finite set to itself can be described by a directed graph with every node having out-degree 1. For a BN, this graph is called the *phase space*, or *state space*.

Definition

The phase space of a BN is the digraph with vertex set $\mathbb{F}_2^n$ and edges $\{(x, f(x)) \mid x \in \mathbb{F}_2^n\}$.

Proposition

Every function $f : \mathbb{F}_2^n \to \mathbb{F}_2^n$ can be expressed uniquely as a Boolean network: $f = (f_1, \ldots, f_n)$.

Proof

Clearly, every BN is a function $\mathbb{F}_2^n \to \mathbb{F}_2^n$. To prove the converse, it suffices to show that these sets have the same cardinality.

To count the number of functions $\mathbb{F}_2^n \to \mathbb{F}_2^n$, we count phase spaces. Each of the 2^n nodes has 1 out-going edge, and 2^n destinations. Thus, there are $(2^n)^{2^n} = 2^{n2^n}$ phase spaces.

To count BNs: there are $2^{(2^n)}$ choices for each f_i, and so $(2^{(2^n)})^n = 2^{n2^n}$ possible BNs. □

Polynomial dynamical systems

Corollary

Every function $f = \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^n$ can be written as an n-tuple of square-free polynomials over $\mathbb{F}_2$. That is,

$$f = (f_1, \ldots, f_n), \qquad f_i \in \mathbb{F}_2[x_1, \ldots, x_n]/\langle x_1^2 - x_1, \ldots, x_n^2 - x_n \rangle.$$

Everything we've done carries over to a generic finite field.

Definition

A polynomial dynamical system (PDS) over a finite field K is a function

$$f = (f_1, \ldots, f_n) \colon K^n \longrightarrow K^n,$$

where the coordinate functions are $f_i \in K[x_1, \ldots, x_n]$.

Iteration of f results in a time-discrete dynamical system.

Remark

If $\operatorname{char} K = p$, then there is a bijection between coordinate functions over K and elements of the quotient ring $K[x_1, \ldots, x_n]/\langle x_1^p - x_1, \ldots, x_n^p - x_n \rangle$.

Asynchronous Boolean networks

Consider an n-tuple of Boolean functions $(f_1, \ldots, f_n)$, where $f_i \in \mathbb{F}_2[x_1, \ldots, x_n]$.

The Boolean network that they determine can be thought of as the results of composing the functions synchronously.

We can also compose them asynchronously. For each coordinate function f_i, define the local function

$$F_i : \mathbb{F}_2^n \longrightarrow \mathbb{F}_2^n, \qquad x = (x_1, \ldots, x_i, \ldots, x_n) \longmapsto (x_1, \ldots, f_i(x), \ldots, x_n).$$

Definition

The asynchronous phase space of $(f_1, \ldots, f_n)$ is the digraph with vertex set $\mathbb{F}_2^n$ and edges $\{(x, F_i(x)) \mid i = 1, \ldots, n,\ x \in \mathbb{F}_2^n\}$.

Remarks

- Clearly, this graph has $n \cdot 2^n$ edges, though self-loops are often omitted.
- Every non-loop edge connect two vertices that differ in exactly one bit. That is, all non-loops are of the form $(x, x + e_i)$, where e_i is the i^{th} standard unit basis vector.
- It is elementary to extend this concept from BNs to general PDSs over finite fields.

Asynchronous Boolean networks

Recall that every function $\mathbb{F}_2^n \longrightarrow \mathbb{F}_2^n$ (i.e., phase space) can be realized as a BN.

Similarly, every digraph with vertex set $\mathbb{F}_2^n$ that could be the asynchronous phase space of a Boolean network, is one.

Theorem

Let $G = (\mathbb{F}_2^n, E)$ be a digraph with the following property:

For every $x \in \mathbb{F}_2^n$ and $i = 1, \ldots, n$: E contains either the self-loop (x, x) or the edge $(x, x + e_i)$ but not both.

Then G is the asynchronous phase space of some Boolean network $(f_1, \ldots, f_n)$.

Proof

As before, it suffices to show there there are 2^{n2^n} distinct graphs G with this property.

Each of the 2^n nodes has n out-going edges (including loops). Each of these edges has 2 possible destinations: x or $x + e_i$.

This gives 2^n choices at each node, for all 2^n nodes, for $(2^n)^{2^n} = 2^{n2^n}$ graphs in total. □

Phase spaces: synchronous vs. asynchronous

The synchronous phase space of a BN $f = (f_1, \ldots, f_n)$ has two types of nodes $x \in \mathbb{F}_2^n$:

- *transient points*: $f^k(x) \neq x$ for all $k \geq 1$.
- *periodic points*: $f^k(x) = x$ for some $k \geq 1$. ($k = 1$: *fixed point*)

Thus, the phase space consists of periodic cycles and directed paths leading into these cycles.

The asynchronous phase space of $f = (f_1, \ldots, f_n)$ can be more complicated.

For $x, \in y \in \mathbb{F}_2^n$, define $x \sim y$ iff there is a directed path from x to y and from y to x.

The resulting equivalence classes are the strongly connected components (SCC) of the phase space. An SCC is terminal if it has no out-going edges from it.

A point $x \in \mathbb{F}_2^n$:

- *is transient* if it is not in a terminal SCC.
- *lies on a cyclic attractor* if its terminal SCC is a k-cycle ($k = 1$: *fixed point*).
- *lies on a complex attractor* otherwise.

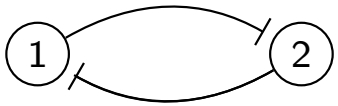
Proposition

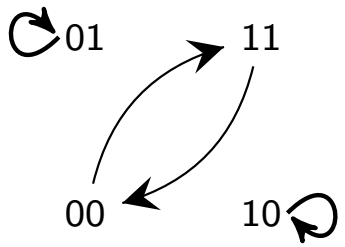
The fixed points of a BN are the same under synchronous and asynchronous update. □

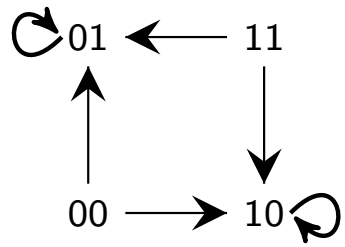
Phase spaces: synchronous vs. asynchronous

Consider the following Boolean network:

$$\begin{cases} f_2(x_1, x_2) = \overline{x_2} \\ f_1(x_1, x_2) = \overline{x_1} \end{cases}$$







The asynchronous phase space is on the far right, and to the left of that is the synchronous phase space.