# The Dynamics of Directed NOR Networks

Jeremiah Rogers
with instruction by Dr. Abdul Jarrah
Virginia Bioinformatics Institute
jeremiahr@vt.edu

May 10, 2006

**Abstract**

This work investigates the dynamics of simultaneously updated boolean networks defined on directed graphs, where each vertex updates to the logical NOR value of the states of the vertices in its in-neighborhood. We prove the existence and number of limit cycles and the width of state space graphs for directed cycles, complete graphs, and complete bipartite graphs. We show that every strongly connected graph has a limit cycle of length 2, and that steady states correspond to independent sets of vertices which dominate all of the graph but themselves.

## 1 Introduction

Agent based simulations are useful for the study of a variety of otherwise difficult problems. Generally each agent of such a system is assumed to sit on a structure of some regularity. For Cellular Automata, each agent is one square of a two dimensional grid and interacts with the agents in those squares surrounding it. For the system studied here, each agent is the vertex of a graph and interacts with the agents with which it shares an edge.

Each vertex of the graph may be given either a 1 or a 0 as a *state*. At one instant every vertex updates to a new state based on the states of vertices in its neighborhood. This system is called a *boolean network*. This paper is concerned with the dynamics of this system as it updates repeatedly.

The following example may add clarity. Figure 1 shows the *dependency graph* of three vertices and their corresponding *local update functions* $f_1, f_2$, and $f_3$. The dependency graph has an edge from vertex $i$ to vertex $j$ if the state $x_i$ is an input to the update function $f_j$. The function $f_3$ is just the logical OR function on the states $x_1$ and $x_3$. So if either $x_1 = 1$ or $x_3 = 1$, then $x_3 = 1$ after the graph updates. In Figure 2 the *state space graph* $\mathsf{S}(G)$ is shown. Each vertex in the state space graph represents an entire configuration of $1, 0$ values on vertices of the graph (a *state of the graph*) and each edge is from one state to the state to which it updates. Since there are three vertices in $G$, there are $2^3 = 8$ vertices in the state space graph of $G$.

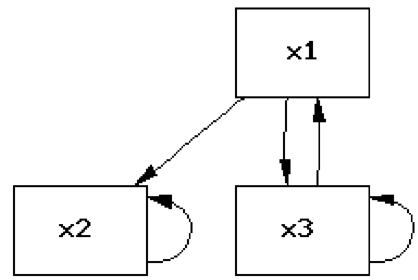
The above described system is one example of a *boolean network*. The useful output of the boolean network is the state space graph. Useful properties to know about the state space are the number of *limit cycles*, the number of *transient states*, and the time it takes to reach a limit cycle from a transient state. Limit cycles are *directed cycles* in the state space graph and transient states are states not contained in directed cycles.

The state space in Figure 1 has two limit cycles, one of length 1 and the other of length 2 (such as from 111 to 101). It also has five transient states (such as 000).

The naive method to generate the state space for such a system is to compute the successor state of each possible state. The generated state space can be analyzed for limit cycles and transient states. However, as the number of vertices grows beyond small values this approach becomes unreasonable.

$$
\begin{aligned}
f_1(x_1, x_2, x_3) &= \quad x_3 \\
f_2(x_1, x_2, x_3) &= \quad \neg(x_1 \wedge x_2) \\
f_3(x_1, x_2, x_3) &= \quad x_1 \vee x_3
\end{aligned}
$$

Local Update Rules




Dependency Graph

Figure 1: Local update rules and corresponding dependency graph

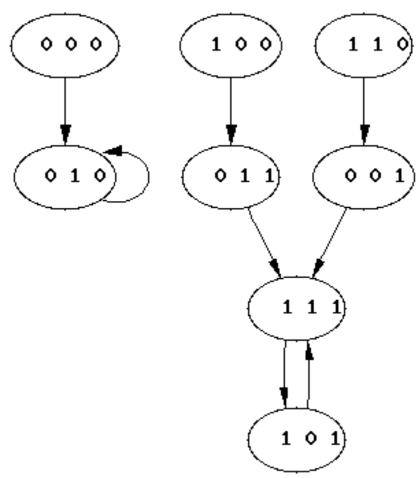



Figure 2: The state space graph for the system in Figure 1

The number of possible states, the time required for state space generation, and the time required for state space analysis all double with each vertex added to the graph. A conventional PC can generate and analyze state spaces for graphs of at most about 16 vertices using available software.

With hopes of eliminating the necessity of complete state space generation, attempts have been made to deduce system dynamics based on the topology of the dependency graph. In general there is no correspondence between the dependency graph and the state space. However, for certain local update functions there is a correspondence. Functions symmetric on their inputs, such as the logical `AND`, `OR`, `NAND` and `NOR`, have one-to-one correspondences with their dependency graphs. This paper studies boolean networks where each vertex updates to the logical `NOR` of the states of vertices in its in-neighborhood.

The system `AND` on directed graphs was studied by Colon-Reyes, Laubenbacher, and Pareigis [2]. Barrett, Chen and Zheng [1] analyzed `NOR` and `OR` on undirected graphs where each vertex considers itself as an input. Little is known about arbitrary directed graphs with the `NOR` update rule.

## 2 Definitions

Let $G$ be a directed graph where $V(G) = \{1, 2, \ldots, n\}$ and $E(G) \subseteq V(G)^2$. For each vertex $i$ there is a corresponding in-neighborhood $N^-(i) = \{j \mid (j,i) \in E(G)\}$ and out-neighborhood $N^+(i) = \{j \mid (i,j) \in E(G)\}$. For each vertex $i \in G$, $f_i$ is the boolean `NOR` function of the states of the vertices in the in-neighborhood of $i$. This function can be written as the polynomial

$$f_i(s) = \prod_{j \in N^-(i)} (1 + s_j) \ ,$$

so that $f_i(s) = 1$ iff $s_j = 0$ for each $j \in N^-(i)$. Defining $F = (f_1, f_2, \ldots, f_n)$, this describes a Parallel Dynamical System using the `NOR` function (a `NOR-PDS`).

**Example** If $N^-(5) = \{2, 3\}$ then $f_5(s) = (1 + s_2)(1 + s_3) = 1 + s_2 + s_3 + s_2 s_3$. If $s_2 = s_3 = 0$, then $f_5(s) = 1$. Otherwise $f_5(s) = 0$.

**Example** Let $G = C_3$. Then the global update function $F \colon {\mathbb{F}_2}^3 \to {\mathbb{F}_2}^3$ is given by $F = (1 + x_3, 1 + x_1, 1 + x_2)$. Figure 3 shows the dependency graph and state space of $C_3$.

The state space graph $\mathsf{S}(G)$ is the directed graph with $V = \mathbb{F}_2^n$ and $E = \{(s,t) \mid F(s) = t \text{ for } s, t \in \mathbb{F}_2^n\}$. We often make statements about the state space graph or the "state space" without explicitly referencing it using graph theoretic terms. Definitions of some common terms relating to $\mathsf{S}(G)$ are given below.

**Limit cycle** A graph $G$ has a *limit cycle* of length $k$ (a $k$-limit cycle) if $\mathsf{S}(G)$ contains a $k$-cycle. Thus $k$ is the least integer where $s = F(s)^k$ for some $s \in \mathbb{F}_2^n$.

**Periodic state** If a state $s$ is contained in a limit cycle it is *periodic*.

**Transient state** If a state is not periodic it is *transient*. So for each positive integer $n$, $F^n(s) \neq s$.

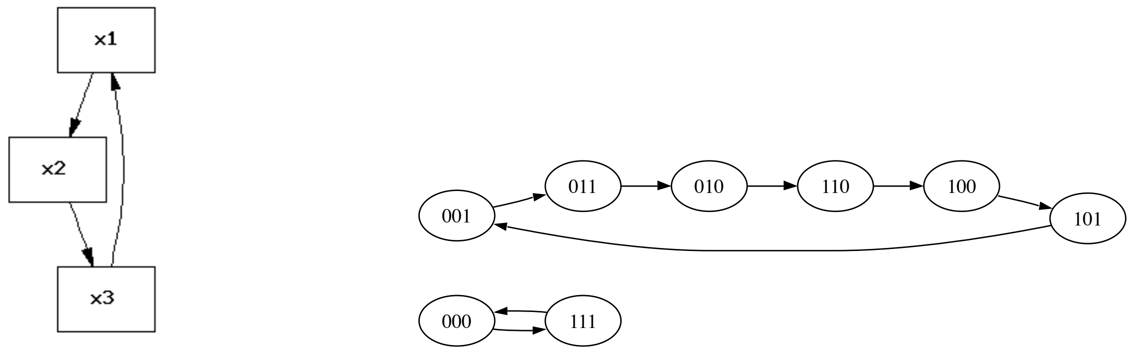
**Garden of Eden state** A state $s$ for which $F(t) \neq s$ for any state $t$ is called a *Garden of Eden* state. Such state is "unreachable" from any other state.

**Width** The *width* of state space $\mathsf{S}(G)$ is the maximum distance from any state to a periodic state. This is the minimum $w$ such that for each state $s$ there is a $k$ with $F^{w+k}(s) = F^w(s)$.

**Notation** If $s, s' \in \mathbb{F}_2^n$ such that $F(s) = s'$, we write $s \to s'$. If $s \to s'$ and $s'_i = 1$ (or $s'_i = 0$) we write $i \to 1$ (or $i \to 0$).

We also define the structure of several classes of graphs studied.

**Cycle** A graph $G = C_n$ is a *directed $n$-cycle* if $E(G) = \{(n,1)\} \bigcup \{(i, i+1) \mid 1 \leq i < n\}$. The graph $C_3$ and its state space are shown in Figure 3.




**Dependency Graph** **State Space**

Figure 3: This output from DVD (web-accessible at `http://dvd.vbi.vt.edu`) shows the dependency graph of $C_3$ and its state space. The vertex label "010" means that $x_1 = 0, x_2 = 1$ and $x_3 = 0$. The state space graph contains a 6-limit cycle and the "degenerate" 2-limit cycle.

$K_3$ $K_{2,2,2}$

Figure 4: The complete graph $K_3$ is pictured with each of its 3 partite sets shaded differently. The complete 3-partite graph $K_{2,2,2}$ has each partite set shaded differently. In Section 3.1 it is shown that each partite set of either $K_3$ or $K_{2,2,2}$ corresponds to a steady state.

**Complete partite graph** A graph $G = K_{x_1,x_2,\ldots,x_m}$ is the *complete m-partite directed graph on n vertices*, if $V(G) = X_1 \cup X_2 \cup \ldots X_m$ with each $X_i$ nonempty, each $|X_i| = x_i$, and for each $i \neq j$ every vertex in $X_i$ has an edge to every vertex in $X_j$, but no edges to any vertices in $X_i$. See Figure 4.

**Complete graph** A graph $G = K_n$ is the *complete directed graph on n vertices*, if $E(G) = \{(i,j) \in V^2 \mid i \neq j\}$. So there are edges in both directions between each pair of vertices, but no self loops, as in Figure 4. It follows that $K_n$ is an $n$-partite complete directed graph.

## 3 Subsets of vertices

To find structural explanations of system behavior we examine the sets of vertices of $G$ which "are 1" in a given time period. These are given by the map $\phi$ defined below. The notation for neighborhoods is extended to sets $W$ so that

$$N^+(W) = \bigcup_{j \in W} N^+(j) \quad \text{and}$$
$$N^-(W) = \bigcup_{j \in W} N^-(j) \quad .$$

Some properties of sets of vertices are given.

**The map phi** Define $\phi \colon \mathbb{F}_2^n \to G$ such that $\phi(s) = \{i \in V \mid s_i = 1\}$ for each $s \in \mathbb{F}_2^n$. This map is bijective, so the inverse is defined and $\phi^{-1}(\phi(s)) = s$.

**Dominating set** If all of $G$ is in $N^+(I)$, or if $N^+(I)^c = \emptyset$, then $I$ is *dominating*.

**Dominating and independent** If $I$ is an independent which dominates all of $G$ but itself, then $I$ is *dominating and independent*. So $N^+(I)^c = I$ and $N^+(I) \cap I = \emptyset$.

**Update notation** For $A \subseteq V$ the notation $A \to 1$ (or $A \to 0$) means that for each vertex $i \in A$, $i \to 1$ (or $i \to 0$). For $W, W' \subseteq V$ such that $s = \phi^{-1}(W)$ and $s' = \phi^{-1}(W')$, the notation $W \to W'$ means that $F(s) = s'$.

**Theorem 3.1 (Set Update Rule)** *Put $W, W' \subseteq V$ and $s, s' \in \mathbb{F}_2^n$ such that $\phi(s) = W$ and $\phi(s') = W'$. Then $F(s) = s'$ iff $N^+(W)^c = W'$.*

**Proof** Assume $s \to s'$ and pick $i \in W'$. Then $s'_i = 1$. So $f_i(s) = 1$. Thus each vertex in the in-neighborhood of $i$ is not in $W$, and $\{i\} \cap N^+(W) = \emptyset$. So $i \in W' \subseteq N^+(W)^c$. Additionally if $j \in W'^c$, then $s_j = 0$ and $f_j(s) = 0$. So $j \in N^+(W)$ and $j \notin N^+(W)^c$. That is: $N^+(W)^c = W'$. ∎

**Corollary 3.2 (The Degenerate Cycle)** *Every strongly-connected graph $G$ has a 2-limit cycle, termed the* degenerate *limit cycle.*

**Proof** We note that by $G$ strongly connected no vertex has an empty in or out-neighborhood. So it is straightforward that $N^+(V)^c = V \setminus V = \emptyset$ and that $N^+(\emptyset)^c = V \setminus \emptyset = V$. This establishes a 2-limit cycle where $V \to \emptyset \to V$. ∎

Corollary 3.2 provides that all graphs studied in this paper have the degenerate 2-limit cycle. Hereafter the existence of this limit cycle will not be explicitly stated unless necessary.

**Corollary 3.3** *For $W \subseteq V$, $W \to W$ iff $W$ is a dominating and independent subset of $V$.*

**Proof** Note $W$ is *dominating and independent* iff $N^+(W)^c = W$ iff $W \to W$ by Theorem 3.1. ∎

## 3.1 Two graphs with only steady states

Two examples of graphs with only steady states (and the degenerate cycle, of course) are given below.

**Theorem 3.4** *The complete $m$-partite graph on $n$ vertices, $G = K_{x_1, x_2, \ldots, x_m}$, has $m$ steady states and width 1.*

**Proof** Let $X_1, X_2, \ldots X_m$ be the partite sets of $G$. By Corollary 3.3 $G$ has a steady state for each dominating and independent set, and by $G$ complete $m$-partite, each partite set $X_i$ is dominating and independent. So there are $m$ steady states corresponding to each $\phi^{-1}(X_i)$.

We show that these are the only non-degenerate limit cycles. Let $W \subseteq V$ such that $W \neq X_i$ for all $i$. There are three cases to consider:

**Case 1:** If $W = \emptyset$ then $\phi^{-1}(W)$ is in the degenerate cycle.

**Case 2:** If $W \neq \emptyset$ and is entirely contained in one of the independent sets $X_i$, then by $G$ complete $X_j \subset N^+(W)$ and $X_j \to 0$ for each $j \neq i$. Additionally $N^+(W) \cap X_i = \emptyset$, so $X_i \to 1$ and $N^+(W)^c = X_i$ and $W \to X_i$.

**Case 3:** If $W$ contains points from at least two distinct $X_i$, $X_j$ then for each $u \in V$ either $u \in N^+(W \cap X_i)$ or $u \in N^+(W \cap X_j)$. So $u \to 0$ and $N^+(W)^c = \emptyset$. So $W \to \emptyset$, which is in the degenerate cycle.

So each state $\phi^{-1}(W)$ a steady state, is in the degenerate cycle, or is a predecessor of a steady state or the degenerate cycle. So $G$ has no limit cycles other than those mentioned above, and $\mathsf{S}(G)$ has width 1. ∎

**Corollary 3.5** *The complete graph $K_n$ has $n$ steady states and its state space has width* 1.

**Proof** Note $K_n$ has $n$ single vertex dominating and independent sets so is complete $n$-partite. The rest follows by Theorem 3.4. ∎

### 3.2 Acyclic graphs

Thus far we have only considered strongly connected graphs, graphs where each vertex has a directed path to every other vertex. It is equivalent that each vertex is contained in a directed cycle. A solution for acyclic graph exists, so it is given below.

Behavior has so far been undefined for vertices with no in-degree. We define that if a vertex $i$ has no in-degree, that is if $N^-(i) = \emptyset$, then $f_i \equiv x_i$ (the identity function). Two other possible definitions would be that $f_i \equiv 0$ or $f_i \equiv 1$, but the first definition encompasses both of these cases.

**Theorem 3.6** *An acylic graph $G$ on $n$ vertices has width equal to the length of its longest path and $2^m$ steady states, where $m$ is the number of vertices of in-degree* 0.

**Proof** Put $A_0 = \{i \in G \mid |N^-(i)| = 0\}$, and let $l$ be the length of the longest path in $G$. For $i = 1, 2, \ldots, l$, let $A_i = \{i \in G \mid N^-(i) \cap A_{i-1} \neq \emptyset\}$. Then

$$G = \bigcup_{i=0}^{l} A_i \qquad \text{and let}$$

$$H_i = A_i \setminus (\bigcup_{j=i+1}^{l} A_j) \qquad \text{so that}$$

$$G = \bigcup_{i=0}^{l} H_i \qquad \text{and}$$

$$H_i \neq H_j \qquad \text{for } i \neq j$$

Put $s \in \mathbb{F}_2^n$. For $i \in H_0$, $N^-(i) = \emptyset$ so $i$ never changes state. Assume that for $0 \leq i < k$, each vertex in $H_i$ has fixed state after $i$ steps. Then each vertex $j \in N^-(H_k)$ is uniquely determined after $k-1$ steps, and each vertex in $H_k$ is determined after $k$ steps. So the whole graph reaches a steady state after $l$ steps, and $G$ has width $l$.

Note that $m = |A_0|$ and that there are $2^m$ possible initial states for vertices in $A_0$. Since the state of the graph is uniquely determined by the states of the vertices in $A_0$, $G$ has $2^m$ steady states. ∎

## 4 The directed cycle

In this section we provide a complete solution for the directed cycle $C_n$. Each vertex has in-degree 1, so this is the solution to a linear system. A general solution for all linear systems exists, so the properties of $C_n$ described below may be considered as an example of the complicated behavior of `NOR` systems.

For Lemma 4.2 it is necessary to build and dissect state vectors and to understand how verticies in the cycle update. Hence the following lemma and notation.

**Simple path** Let $u, v \in G$. There is a *simple-path* from $u$ to $v$ (written $u \in P_k(v)$) if there is some path $u = p_1, p_2, \cdots, p_n = v$ such that for $1 < i \leq n$, $|N^-(p_i)| = 1$.

**Lemma 4.1** *Let $v_i, v_j \in G$ such that $v_i \in P_k(v_j)$. Then for any $s \in \mathbb{F}_2^n$, $F^k(s_j) = k + s_i \pmod 2$.*

| $n$ | $k$-limit cycles | $n$ | $k$-limit cycles |
|---|---|---|---|
| 1 | 2 | 11 | 2, 22 |
| 2 | 1, 2 | 12 | 1, 2, 3, 4, 6, 12 |
| 3 | 2, 6 | 13 | 2, 26 |
| 4 | 1, 2, 4 | 14 | 1, 2, 7, 14 |
| 5 | 2, 10 | 15 | 6, 10, 30 |
| 6 | 1, 2, 3, 6 | 16 | 1, 2, 4, 8, 16 |
| 7 | 2, 14 | 17 | 2, 34 |
| 8 | 1, 2, 4, 8 | 18 | 1, 2, 3, 6, 9, 18 |
| 9 | 2, 6, 18 | 19 | 2, 38 |
| 10 | 1, 2, 5, 10 | 20 | 1, 2, 4, 5, 10, 20 |

Table 1: Limit cycle lengths of some $C_n$. Limit cycle lengths are determined by the divisors of $n$.

**Proof** By $v_i \in P_k(v_j)$ choose a $v_i, v_j$ simple-path $v_i = p_0, p_1, \ldots, p_k = v_j$. Then for $1 \leq l \leq k$, $N^-(p_l) = p_{l-1}$ and $F(s_{p_l}) = 1 + s_{p_{l-1}}$. So $F^k(s_j) = 1 + F^{k-1}(s_{p_k-1}) = \cdots = k + F(s_{p_0}) = 1 + F(s_i)$. ∎

**Partition notation** Put $s \in \mathbb{F}_2^n$. For integers $i \leq j$ we denote the states $(s_i, \ldots, s_j)$ by the *partition* $[i, j]$. In the case where $|j - i| = |q - p|$ we will write $[i, j] = [p, q]$ or $[i, j] = 1 + [p, q]$ to mean that for each $0 \leq x \leq i - j$ either $s_{i+x} = s_{p+x}$ or $s_{i+x} = 1 + s_{p+x} \pmod 2$ (respectively).

**Example** So if $[1, 2] = (1, 0)$ then $1 + [1, 2] = (0, 1)$. If $[1, 2] = [3, 4]$ then $[1, 4] = (1, 0, 1, 0)$.

Lemma 4.2 shows which limit cycle lengths are possible, and Theorem 4.3 shows that these are the only possible limit cycles.

**Lemma 4.2** *Let $G$ be $C_n$ and $q$ an integer such that $q \mid n$. If $n$ is even, $G$ has a limit cycle of length dividing $k = q$. If $n$ is odd, $G$ has a limit cycle of a length dividing $k = 2q$. The number of states contained in limit cycles of length dividing $k$ is $2^q$.*

**Proof** Choose $q, l \in \mathbb{N}$ such that $ql = n$. Since $q \mid n$ the states $[1, n]$ divide into the $l$ partitions $[1, q], [q + 1, 2q], \ldots, [(l - 1)q, n]$. We assign values to each partition and verify that after $k$ steps the graph returns to its original state.

Assign arbitrary values from $\{0, 1\}$ to each of the vertices in $[1, q]$. Now for each $1 \leq j < l$, assign values to $[qj, q(j + 1) - 1]$ according to the following two cases.

**Case 1:** *If $n$ is even and $q$ is even, or if $n$ is odd*, then set $[qj, q(j + 1) - 1] = [1, q]$. Then the partitions are all the same ($[1, n] = [1, q], [1, q], \ldots, [1, q]$).

**Case 2:** *If $n$ is even and $q$ is odd*, then put $[qj, q(j + 1) - 1] = 1 + [q(j - 1), qj - 1]$. Then the partitions alternate values ($[1, n] = [1, q], 1 + [1, q], \ldots, 1 + [1, q]$). Since $l$ is even we know that the final partition is $1 + [1, q]$.

Now to verify that $F^k(s_i) = s_i$ for each $i \in G$. This follow directly from the construction and that by Theorem 4.1 each path in $C_n$ is simple. If $n$ is even and $q$ is even, then $f_i^k(s) = f_i^q(s) = q + s_{i-q} = 0 + s_i = s_i$. If $n$ is even and $q$ is odd, then $f_i^k(s) = f_i^q(s) = q + s_{i-q} = 1 + (1 + s_i) = s_i$. If $n$ is odd, then $f_i^k(s) = f_i^{2q}(s) = 2q + s_{i-2q} = 0 + (s_i) = s_i$.

So $G$ has a limit cycle of length dividing $k$. By construction of $s$ there are $2^q$ states contained in such limit cycles. ∎

As an example, possible limit cycle lengths for some values of $n$ are given in Table 1.

**Theorem 4.3** *Let $G$ be $C_n$ and $q$ an integer such that $q \mid n$. If $n$ is even let $k = q$, otherwise let $k = 2q$. Let $l_1, l_2, \ldots, l_m$ be the proper divisors of $q$, and $k_1, k_2, \ldots, k_m$ the values $k$ corresponding to each of these divisors. Then $f(k)$, the number of limit cycles of length $k$, is given by*

$$f(k) = \frac{2^q - \sum_{i=1}^{m} k_i * f(k_i)}{k} .$$

| $k$ | $n$ even | $n$ odd | $k$ | $n$ even | $n$ odd |
|---|---|---|---|---|---|
| 1 | 2 | 0 | 11 | 186 | 0 |
| 2 | 1 | 1 | 12 | 336 | 0 |
| 3 | 2 | 0 | 13 | 630 | 0 |
| 4 | 3 | 0 | 14 | 1,161 | 9 |
| 5 | 6 | 0 | 15 | 2,182 | 0 |
| 6 | 9 | 1 | 16 | 4,080 | 0 |
| 7 | 18 | 0 | 17 | 7,710 | 0 |
| 8 | 30 | 0 | 18 | 14,535 | 28 |
| 9 | 56 | 0 | 19 | 27,594 | 0 |
| 10 | 99 | 3 | 20 | 52,377 | 0 |

Table 2: Occurances of some $k$-limit cycles for $C_n$. If a directed cycle $C_n$ has a $k$-limit cycle, it has the same number of such limit cycles as any other directed of the same parity with a $k$-limit cycle.

*The graph has no limit cycles of lengths corresponding to a $q$ which does not divide $n$, and has width* 0.

**Proof** For arbitrary $q \mid n$, by Lemma 4.2 there are $2^q$ possible states which are in cycles of length dividing $k$. Removing all states that are in limit cycles of length properly dividing $q$, we have $2^q - \sum_{i=1}^{m} l_i * f(l_i)$ states strictly contained in $k$-limit cycles. Dividing by $k$ gives the number of limit cycles.

Since $G$ has only $2^n$ possible state configurations this counting exhausts the state space. So $G$ cannot have limit cycles of lengths $k$ corresponding to a $q$ which does not divide $n$. Additionally, no state is transient and $\mathsf{S}(G)$ has width 0. ∎

So if $n, m$ are of the same parity and both have a $k$-limit cycle, both have the same number of $k$-limit cycles. The occurrences of some limit cycle lengths are given in Table 2.

# References

[1] Barrett, Chen, and Zheng. Discrete dynamical systems on graphs and boolean functions. *Math. Comput. Simulation*, 66:487–497, 2004.

[2] Omar Colon-Reyes, Reinhard Laubenbacher, and Bodo Pareigis. Boolean monomial dynamical systems. *Annals of Combinatorics*, 8:425–439, 2004.