

1-2009

Cycle Equivalence of Graph Dynamical Systems

Matthew Macauley

Clemson University, macaule@clemson.edu

Henning S. Mortveit

Virginia Tech

Follow this and additional works at: https://tigerprints.clemson.edu/mathematics_pubs

 Part of the [Mathematics Commons](#)

Recommended Citation

Please use the publisher's recommended citation. <http://iopscience.iop.org/article/10.1088/0951-7715/22/2/010/meta>

This Article is brought to you for free and open access by the School of Mathematical and Statistical Sciences at TigerPrints. It has been accepted for inclusion in Publications by an authorized administrator of TigerPrints. For more information, please contact kokeefe@clemson.edu.

CYCLE EQUIVALENCE OF GRAPH DYNAMICAL SYSTEMS

MATTHEW MACAULEY

HENNING S. MORTVEIT

ABSTRACT. Graph dynamical systems (GDSs) can be used to describe a wide range of distributed, nonlinear phenomena. In this paper we characterize *cycle equivalence* of a class of finite GDSs called sequential dynamical systems (SDSs). In general, two finite GDSs are cycle equivalent if their periodic orbits are isomorphic as directed graphs. Sequential dynamical systems may be thought of as generalized cellular automata, and use an *update order* to construct the dynamical system map. The main result of this paper is a characterization of cycle equivalence in terms of shifts and reflections of the SDS update order. We construct two graphs $C(Y)$ and $D(Y)$ whose components describe update orders that give rise to cycle equivalent SDSs. The number of components in $C(Y)$ and $D(Y)$ is an upper bound for the number of cycle equivalence classes one can obtain, and we enumerate these quantities through a recursion relation for several graph classes. The components of these graphs encode dynamical neutrality, the component sizes represent periodic orbit structural stability, and the number of components can be viewed as a system complexity measure.

1. INTRODUCTION

Sequential dynamical systems (SDSs) were introduced in [5, 11]. These are dynamical systems constructed from (i) a finite undirected graph Y where each vertex has a state, (ii) a sequence of vertex functions, and (iii) a word w over the vertex set of Y . The SDS map is constructed as the composition of the functions in the order specified by w . As such, they represent a useful framework for describing distributed phenomena with causal interactions. This paper is about *cycle equivalence* of finite graph dynamical systems, which we study in the context of SDSs. Two SDSs are cycle equivalent if their periodic orbits are isomorphic as directed graphs. We will study how the update order affects the structure of the periodic orbits, and thus the long-term behavior of the system. As an example, we show the surprising result that if the GDS base graph is a tree then there is only one possible periodic orbit configuration, and this holds for any fixed choice of vertex functions. SDS and SDS-like algorithms occur in many application areas such as [8, 12], and our results will provide a behavioral complexity measure for these.

2000 *Mathematics Subject Classification.* 37B99;93D99;20F55.

Key words and phrases. Finite dynamical systems over graphs, cycle equivalence, update order, generalized cellular automata, enumeration, stability, complexity.

This work was partially supported by Fields Institute in Toronto, Canada.

This paper is organized as follows. In Section 2 we describe SDS related terminology and relevant background results from [11, 13] on functional and dynamical equivalence of SDSs. In Section 3, we prove one of the main results of this paper: any two SDSs where the update orders differ by a cyclic shift are cycle equivalent, and this holds for any choice of vertex functions. Additionally, when the vertex states are taken from $\mathbb{F}_2 = \{0, 1\}$, which is the standard choice in most studies of cellular automata, then reflections of the update order also encode cycle equivalent SDSs. We also show how shifts and reflections of update orders have a natural interpretation in terms of source-to-sink operations on acyclic orientations of the GDS graph. In Section 4 we introduce the graphs $C(Y)$ and $D(Y)$ which form the basis for our analysis and characterization of cycle equivalence over general graphs. These graphs are examples of neutral networks, and we characterize some of their structural properties. We study the functions $\kappa(Y)$ and $\delta(Y)$, which count the connected components of $C(Y)$ and $D(Y)$, respectively. We show how $\delta(Y)$ is given in terms of $\kappa(Y)$ and give several results for the computation of $\kappa(Y)$ with implications to dynamics. These functions can be regarded as a measure for system complexity since they are upper bounds for the number of SDS maps up to cycle equivalence achievable through variations of the update order. As a computational example we demonstrate how $\kappa(Y)$ increases from $\Theta(n)$ for radius-1 rules (the elementary cellular automaton rules) to $\Theta(n \cdot 2^n)$ for radius-2 rules. We also show how the presence of symmetries in the base graph may allow for significantly improved bounds in certain cases. In the summary section we show how cycle equivalence of SDSs is closely related to Coxeter theory. Some of the results that we prove in this paper have a natural analog when translated into the language of Coxeter groups. This opens the door to use the rich mathematical tools and results of Coxeter theory to study sequential dynamical systems, something that has never been done before.

2. BACKGROUND AND DEFINITIONS

Let Y be a finite undirected graph with vertex set $v[Y] = \{1, \dots, n\}$, and edge set $e[Y]$. Since most graphs in this paper are finite and undirected we simply refer to this class of graphs as “graphs” and specify if needed. The *1-neighborhood* of vertex v in Y is $B_1(v; Y) = \{v' \in v[Y] \mid \{v, v'\} \in e[Y]\} \cup \{v\}$, and the *ordered 1-neighborhood* $n[v]$ of v is the sequence of vertices from $B_1(v; Y)$ ordered in increasing order. The *degree* of vertex v is written $d(v)$. Each vertex v is assigned a state $y_v \in K$ where K is a finite set. In the following y_v is called a *vertex state* and the n -tuple $\mathbf{y} = (y_1, \dots, y_n)$ is a *system state*.¹ We write

$$(2.1) \quad \mathbf{y}[v] = (y_{n[v](1)}, \dots, y_{n[v](d(v)+1)}),$$

for the restriction of the system state to the vertices in $n[v]$, and let $\mathbf{y}'[v]$ denote the same tuple but with the vertex state y_v omitted. The finite field with $q = p^k$ elements is denoted \mathbb{F}_q .

¹In the context of, e.g. cellular automata a system state is frequently called a *configuration*.

Let $\mathbf{f}_Y := (f_i)_{i \in \mathbf{v}[Y]}$ be a sequence of *vertex functions* $f_i: K^{d(i)+1} \rightarrow K$, and define the sequence of *Y-local functions* $\mathfrak{F}_Y := (F_i)_{i \in \mathbf{v}[Y]}$ with $F_i: K^n \rightarrow K^n$ by

$$(2.2) \quad F_i(y_1, \dots, y_n) = (y_1, \dots, y_{i-1}, f_i(\mathbf{y}[i]), y_{i+1}, \dots, y_n) .$$

It is clear that \mathbf{f}_Y completely determines \mathfrak{F}_Y , and vice-versa. However, there are settings when it is easier to speak of one rather than the other.

Let W_Y denote the set of words over $\mathbf{v}[Y]$.² Words are written as $w = (w_1, w_2, \dots, w_m)$, $w = w_1 w_2 \dots w_m$, $w = (w(1), w(2), \dots, w(m))$, etc. The subset of W_Y where each element of $\mathbf{v}[Y]$ occurs exactly once is denoted S_Y . The elements of S_Y may thus be thought of as permutations of $\mathbf{v}[Y]$. The symmetric group S_n acts on system states by

$$(2.3) \quad \gamma \cdot (y_1, \dots, y_n) = (y_{\gamma^{-1}(1)}, \dots, y_{\gamma^{-1}(n)}) .$$

Definition 2.1 (Sequential dynamical system). A *sequential dynamical system* (SDS) is a triple (Y, \mathfrak{F}_Y, w) where Y is a graph, $\mathfrak{F}_Y = (F_i)_{i \in \mathbf{v}[Y]}$ is a sequence of *Y-local functions*, and $w = (w_1, \dots, w_m) \in W_Y$. The associated SDS map $[\mathfrak{F}_Y, w]: K^n \rightarrow K^n$ is the function composition

$$(2.4) \quad [\mathfrak{F}_Y, w] = F_{w_m} \circ F_{w_{m-1}} \circ \dots \circ F_{w_2} \circ F_{w_1} .$$

The graph Y of an SDS is called the *base graph*, and when $w \in S_Y$, the SDS is referred to as a *permutation SDS*. A sequence of *Y-local functions* \mathfrak{F}_Y is *Aut(Y)-invariant* if $\gamma \circ F_v = F_{\gamma(v)} \circ \gamma$ for all $v \in \mathbf{v}[Y]$ and all $\gamma \in \text{Aut}(Y)$. Here, the composition of a function $K^n \rightarrow K^n$ with a permutation of K is interpreted as in (2.3). The corresponding sequence of vertex functions \mathbf{f}_Y is *Aut(Y)-invariant* if \mathfrak{F}_Y is *Aut(Y)-invariant*. The *phase space* of the map $\phi: K^n \rightarrow K^n$ is the directed graph $\Gamma(\phi)$ with vertex set K^n and edge set $\{(\mathbf{y}, \phi(\mathbf{y})) \mid \mathbf{y} \in K^n\}$. The following example illustrates these concepts.

Example 2.2 (Asynchronous Elementary Cellular Automaton rule # 1.). Let $Y = \text{Circ}_4$ which is the graph with vertex set $\{1, 2, 3, 4\}$ and edges all $\{i, i+1\}$ with indices modulo 4, and take binary vertex states. Then $y = (y_1, y_2, y_3, y_4)$, $n[1] = (1, 2, 4)$, and $y[1] = (y_1, y_2, y_4)$. If we use the Boolean *nor*-function $\text{nor}_3: \mathbb{F}_2^3 \rightarrow \mathbb{F}_2$ (i.e. ECA # 1) given by $\text{nor}_3(x, y, z) = (1+x)(1+y)(1+z)$ to induce the vertex functions we get, e.g. $F_1(y) = (\text{nor}_3(y[1]), y_2, y_3, y_4)$. With update order $\pi = (1, 2, 3, 4)$ we get the SDS map

$$(2.5) \quad [\text{Nor}_Y, \pi] = F_4 \circ F_3 \circ F_2 \circ F_1 .$$

It is easy to verify that $[\text{Nor}_Y, \pi](0, 0, 0, 0) = (1, 0, 1, 0)$. In contrast, for a parallel update scheme the state $(0, 0, 0, 0)$ would have been mapped to $(1, 1, 1, 1)$. The entire phase space of the SDS map in (2.5) is displayed on the left in Figure 1.

What follows is a short overview of functional and dynamical equivalence of SDSs. The analysis is largely concerned with the aspect of update order and characterizes SDS maps for a fixed graph Y and fixed *Y-local functions* \mathfrak{F}_Y in terms of w . It will provide the basis for cycle equivalence.

²Also referred to as the Kleene star or Kleene closure of $\mathbf{v}[Y]$.

2.1. Functional Equivalence. Two SDSs are *functionally equivalent* if their SDS maps are identical as functions. For a fixed sequence \mathfrak{F}_Y , a natural question to ask is when is $[\mathfrak{F}_Y, w] = [\mathfrak{F}_Y, w']$ for $w, w' \in W_Y$. The *update graph* $\hat{U}(Y)$ provides an answer to this. The update graph of Y has vertex set W_Y . Two words $w \neq w'$ are adjacent if they have equal length, say m , and (i) they differ *only* by a transposition of entries k and $k + 1$, and (ii) $\{w_k, w_{k+1}\} \notin e[Y]$. The finite subgraph $U(Y)$ of $\hat{U}(Y)$ induced by the vertex set S_Y is called the *permutation update graph*, and is denoted $U(Y)$. Clearly, it is a union of connected components of $\hat{U}(Y)$. Both $U(Y)$ and $\hat{U}(Y)$ are examples of neutral networks as mentioned in the introduction. The update graph $U(\text{Circ}_4)$ is shown Figure 2.

Let \sim_Y be the equivalence relation on S_Y defined by $\pi \sim_Y \pi'$ iff π and π' belong to the same connected component in $U(Y)$. We denote equivalence classes as $[\pi]_Y$ and the set of equivalence classes by S_Y/\sim_Y , i.e.

$$S_Y/\sim_Y = \{[\pi]_Y \mid \pi \in S_Y\} .$$

By construction, $\pi \sim_Y \sigma$ implies the equality $[\mathfrak{F}_Y, \pi] = [\mathfrak{F}_Y, \sigma]$. If the vertex functions are the Boolean functions as in Example 2.2, then $[\text{Nor}_Y, \pi] = [\text{Nor}_Y, \sigma]$ implies $\pi \sim_Y \sigma$ (see [11]). It follows that $|S_Y/\sim_Y|$ is a sharp upper bound for the number of functionally non-equivalent permutation SDS maps obtainable by varying the update order.

Functional equivalence can also be characterized through acyclic orientations. An orientation of Y is a map $O_Y: e[Y] \rightarrow v[Y] \times v[Y]$ that sends an edge $\{i, j\}$ to either (i, j) or (j, i) . Let $\text{Acyc}(Y)$ denote the set of acyclic orientations of Y , that is, orientations that contain no directed cycles. In [13] it is shown that there is a bijection

$$(2.6) \quad f_Y: S_Y/\sim_Y \rightarrow \text{Acyc}(Y) .$$

A permutation $\pi \in S_Y$ defines a linear order on $v[Y]$ by $\pi_k = i <_\pi j = \pi_\ell$ iff $k < \ell$. This order defines an acyclic orientation O_Y^π where $O_Y^\pi(\{v, v'\})$ equals (v, v') if $v <_\pi v'$ and (v', v) otherwise. The map f_Y in (2.6) sends $[\pi]_Y \in S_Y/\sim_Y$ to O_Y^π . It follows that

$$(2.7) \quad \alpha(Y) = |\text{Acyc}(Y)|$$

is a sharp upper bound for the number of functionally non-equivalent permutation SDSs that can be obtained by varying the update order. The result can be extended to general word update orders $w \in W_Y$. We do not review this here, but refer to [14].

2.2. Dynamical Equivalence. Two finite dynamical systems $\phi, \psi: K^n \rightarrow K^n$ are *dynamically equivalent* if there exists a bijection $h: K^n \rightarrow K^n$ such that

$$(2.8) \quad \phi \circ h = h \circ \psi .$$

With the discrete topology the concepts of dynamical equivalence and topological conjugation coincide. Thus, the difference between functional and dynamical equivalence is that in the former, the phase spaces are identical, but in the latter, the phase spaces need just be isomorphic. Update orders that are related by an automorphism of the base graph give rise

to dynamically equivalent SDSs. The number of orbits $\bar{\alpha}(Y)$ under the action of $\text{Aut}(Y)$ on S_Y/\sim_Y given by $\gamma \cdot [\pi]_Y = [\gamma * \pi]_Y$, where

$$(2.9) \quad \gamma * w = \gamma(w_1), \dots, \gamma(w_m),$$

is an upper bound for the number of SDS maps up to dynamical equivalence. This follows since for SDSs with $\text{Aut}(Y)$ -invariant vertex functions one has (see [11])

$$(2.10) \quad [\mathfrak{F}_Y, \gamma * \pi] \circ \gamma = \gamma \circ [\mathfrak{F}_Y, \pi]$$

for all $\pi \in S_Y$ and all $\gamma \in \text{Aut}(Y)$. Via the bijection in (2.6), this action carries over to an action on the set $\text{Acyc}(Y)$, and the number of orbits is given by

$$\bar{\alpha}(Y) = \frac{1}{|\text{Aut}(Y)|} \sum_{\gamma \in \text{Aut}(Y)} \alpha(\langle \gamma \rangle \setminus Y).$$

Here $\langle \gamma \rangle \setminus Y$ denotes the *orbit graph* of the cyclic group $G = \langle \gamma \rangle$ and Y , see [3, 4]. This bound is known to be sharp for certain graph classes [4], but in the general case this is still an open problem.

3. CYCLE EQUIVALENCE

Definition 3.1. Two finite dynamical systems $\phi: K^n \rightarrow K^n$ and $\psi: K^m \rightarrow K^m$ are *cycle equivalent*³ if there exists a bijection $h: \text{Per}(\phi) \rightarrow \text{Per}(\psi)$ such that

$$(3.1) \quad \psi|_{\text{Per}(\psi)} \circ h = h \circ \phi|_{\text{Per}(\phi)},$$

where $\psi|_{\text{Per}(\psi)}$ and $\phi|_{\text{Per}(\phi)}$ denote the restrictions of the maps to their respective sets of periodic points $\text{Per}(\psi)$ and $\text{Per}(\phi)$. Two systems ϕ and ψ with identical periodic orbits are *functionally cycle equivalent*.

Example 3.2. As an illustration we continue Example 2.2 with $Y = \text{Circ}_4$ with vertex functions $\text{nor}_3: \mathbb{F}_2^3 \rightarrow \mathbb{F}_2$ with update orders $\pi = (1, 2, 3, 4)$, $\pi' = (1, 4, 2, 3)$ and $\pi'' = (1, 3, 2, 4)$. The two SDS maps $[\text{Nor}_Y, \pi']$ and $[\text{Nor}_Y, \pi'']$ are cycle equivalent, which can

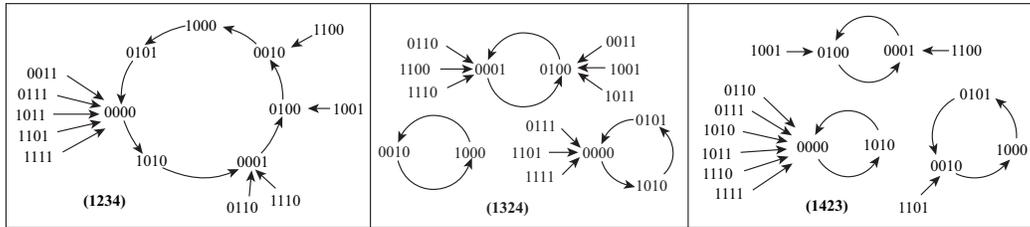


FIGURE 1. The phase spaces of Example 3.2.

be seen in the two rightmost phase spaces in Figure 1. They are not functionally cycle equivalent. Later we show that for $Y = \text{Circ}_4$ there are at most 2 cycle configurations when $K = \mathbb{F}_2 = \{0, 1\}$.

³In general one can define this where ϕ and ψ have different vertex state sets K and K' .

It is clear that both functional equivalence and dynamical equivalence imply cycle equivalence. Define $\sigma, \rho \in S_m$ to be the permutations

$$\sigma = (m, m-1, \dots, 2, 1), \quad \rho = (1, m)(2, m-1) \cdots (\lceil \frac{m}{2} \rceil, \lfloor \frac{m}{2} \rfloor + 1),$$

and let C_m and D_m be the groups

$$(3.2) \quad C_m = \langle \sigma \rangle \quad \text{and} \quad D_m = \langle \sigma, \rho \rangle .$$

Both C_m and D_m act on the set of length- m update orders via (2.3). Define the s -shift $\sigma_s(w) = \sigma^s \cdot w$, and the reflection $\rho(w) = \rho \cdot w = (w_m, w_{m-1}, \dots, w_2, w_1)$. We can now state one of the main results.

Theorem 3.3. *For any $w \in W_Y$, the SDS maps $[\mathfrak{F}_Y, w]$ and $[\mathfrak{F}_Y, \sigma_s(w)]$ are cycle equivalent.*

Proof. Set $P_k = \text{Per}[\mathfrak{F}_Y, \sigma_k(w)]$. By the definition of an SDS map, the following diagram commutes

$$(3.3) \quad \begin{array}{ccc} P_{k-1} & \xrightarrow{[\mathfrak{F}_Y, \sigma_{k-1}(w)]} & P_{k-1} \\ F_{w^{(k)}} \downarrow & & \downarrow F_{w^{(k)}} \\ P_k & \xrightarrow{[\mathfrak{F}_Y, \sigma_k(w)]} & P_k \end{array}$$

for all $1 \leq k \leq m = |w|$. Thus we obtain the inclusion $F_{w^{(k)}}(P_{k-1}) \subset P_k$, and since the restriction map $F_{w^{(k)}}: P_{k-1} \rightarrow F_{w^{(k)}}(P_{k-1})$ is an injection, it follows that $|P_{k-1}| \leq |P_k|$. We therefore obtain the sequence of inequalities

$$|\text{Per}[\mathfrak{F}_Y, w]| \leq |\text{Per}[\mathfrak{F}_Y, \sigma_1(w)]| \leq \cdots \leq |\text{Per}[\mathfrak{F}_Y, \sigma_{m-1}(w)]| \leq |\text{Per}[\mathfrak{F}_Y, w]| ,$$

from which it follows that all inequalities are, in fact, equalities. Since the graph and state space are finite all the restriction maps $F_{w^{(k)}}$ in (3.3) are bijections. Clearly (3.1) holds with $h = F_{w^{(k)}}$, and the proof follows. \square

Theorem 3.3 shows that acting on the update order by the cyclic group C_m preserves the cycle structure of the phase space. We point out that this result holds for any finite set K . For $K = \mathbb{F}_2$ the cycle structure is also preserved under the action of D_m , and is a consequence of:

Proposition 3.4 ([11]). *Let (Y, \mathfrak{F}_Y, w) be an SDS over \mathbb{F}_2 with periodic points $P \subset \mathbb{F}_2^n$. Then*

$$(3.4) \quad ([\mathfrak{F}_Y, w]|_P)^{-1} = [\mathfrak{F}_Y, \rho(w)]|_P .$$

This result follows since for each vertex function f_i the restriction $f_i(-; \mathbf{y}'[i]): \mathbb{F}_2 \rightarrow \mathbb{F}_2$ is a bijection for each fixed choice of $\mathbf{y}'[i]$. There are only two such maps: the identity map $y_i \mapsto y_i$ and the map $y_i \mapsto 1 + y_i$. From this it follows that composing the two maps in (3.4) in either order gives the identity map, see [11]. The next proposition is now clear:

Proposition 3.5. *For $K = \mathbb{F}_2$ the SDS maps $[\mathfrak{F}_Y, w]$ and $[\mathfrak{F}_Y, \rho(w)]$ are cycle equivalent.*

Thus, for any $g \in G = C_m$ the SDS maps $[\mathfrak{F}_Y, w]$ and $[\mathfrak{F}_Y, g \cdot w]$ are cycle equivalent, where $|w| = m$. If $K = \mathbb{F}_2$ the same statement holds for $G = D_m$. We now have the following situation: elements π and π' with $[\pi]_Y \neq [\pi']_Y$ generally give rise to functionally non-equivalent SDS maps. If there exists $g \in G$, $\bar{\pi} \in [\pi]_Y$ and $\bar{\pi}' \in [\pi']_Y$ such that $g \cdot \bar{\pi} = \bar{\pi}'$, then the classes $[\pi]_Y$ and $[\pi']_Y$ give rise to cycle equivalent SDS maps.

Let Star_n be the graph with vertex set $v[\text{Star}_n] = \{0, 1, \dots, n\}$ and edge set $e[\text{Star}_n] = \{\{0, i\} \mid 1 \leq i \leq n\}$. As a particular example we have:

Corollary 3.6. *Let $Y = \text{Star}_n$ with $n \geq 2$. For a fixed sequence \mathfrak{F}_Y of $\text{Aut}(Y)$ -invariant Y -local maps all permutation SDS maps of the form $[\mathfrak{F}_Y, \pi]$ are cycle equivalent. Any SDS map of the form $[\text{Nor}_Y, \pi]$ with $\pi \in S_Y$ has precisely one periodic orbit of size three, and $2^{n-1} - 1$ periodic orbits of size two.*

Proof. We have $\text{Aut}(\text{Star}_n) \cong S_n$ since the automorphisms of Star_n are precisely the elements of S_Y that fix the vertex 0. An orbit of $\text{Aut}(\text{Star}_n)$ in S_Y/\sim_Y contains all equivalence classes $[\pi]_Y$ for which the position of 0 in π coincide, i say. Thus for $0 \leq i \leq n$ all permutations that have vertex 0 in the i^{th} position give rise to dynamically equivalent SDS maps. Pick $\pi = (0, 1, 2, \dots, n)$. By Corollary 3.5, all permutations that are shifts of π give cycle equivalent SDS maps. The second part now follows by inspection of one of the possible phase spaces. They are all listed in [11], but without enumerations of periodic orbits. \square

4. COMBINATORIAL CONSTRUCTIONS FOR CYCLE EQUIVALENCE

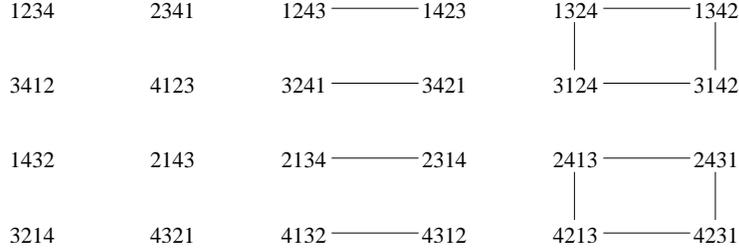
4.1. Neutral Networks. In the remainder of this paper we will only consider permutation update orders, although it is not hard to see how this can be extended to systems with general word update orders. To start, we define two graphs over S_Y/\sim_Y whose connected components give rise to cycle equivalent SDSs for a fixed graph Y and a fixed sequence \mathfrak{F}_Y . Since cycle equivalence is a coarsening of functional equivalence, it is natural to construct these graphs using S_Y/\sim_Y as vertex set rather than S_Y .

Let $C(Y)$ and $D(Y)$ be the graphs defined by

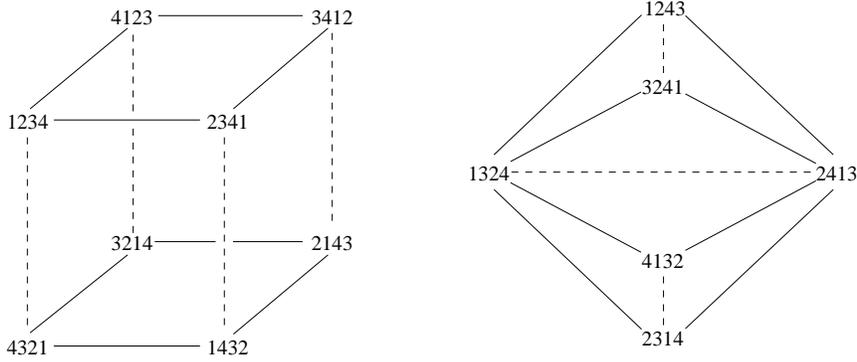
$$\begin{aligned} v[C(Y)] &= S_Y/\sim_Y, & e[C(Y)] &= \{ \{[\pi]_Y, [\sigma_1(\pi)]_Y\} \mid \pi \in S_Y \}, \\ v[D(Y)] &= S_Y/\sim_Y, & e[D(Y)] &= \{ \{[\pi]_Y, [\rho(\pi)]_Y\} \mid \pi \in S_Y \} \cup e[C(Y)]. \end{aligned}$$

Define $\kappa(Y)$ and $\delta(Y)$ to be the number of connected components of $C(Y)$ and $D(Y)$, respectively. It is clear that $C(Y)$ is a subgraph of $D(Y)$, and that $\delta(Y) \leq \kappa(Y)$. By Theorem 3.3, $\kappa(Y)$ is a general upper bound for the number of different SDS cycle equivalence classes obtainable through update order variations. For $K = \mathbb{F}_2$ it follows from Proposition 3.4 that $\delta(Y)$ is an upper bound as well.

Example 4.1. As in Example 2.2 let $Y = \text{Circ}_4$. The permutation update graph $U(\text{Circ}_4)$ has $\alpha(\text{Circ}_4) = 14$ connected components as shown in Figure 2. The graphs $C(\text{Circ}_4)$ and

FIGURE 2. The update graph $U(\text{Circ}_4)$.

$D(\text{Circ}_4)$ are shown in Figure 3 where the dashed lines represent the edges in $e[D(\text{Circ}_4)] \setminus e[C(\text{Circ}_4)]$. The vertices in Figure 3 are labeled by a permutation in the corresponding

FIGURE 3. The graphs $C(\text{Circ}_4)$ and $D(\text{Circ}_4)$. The dashed lines are edges in $D(\text{Circ}_4)$ but not in $C(\text{Circ}_4)$.

equivalence class in S_Y/\sim_Y . The vertices of the cube-shaped component are all singletons in S_Y/\sim_Y . The equivalence classes $[1324]_{\text{Circ}_4}$ and $[2413]_{\text{Circ}_4}$ both consist of four permutations, while the remaining four vertices on that component are equivalence classes that contain precisely two permutations. Clearly, $\kappa(\text{Circ}_4) = 3$ and $\delta(\text{Circ}_4) = 2$.

The following result gives insight into the how κ - and δ -equivalent permutations are distributed across the vertices of the update graph $U(Y)$.

Proposition 4.2. *Let Y be a connected graph on n vertices and let $g, g' \in C_n$ with $g \neq g'$. Then $[g \cdot \pi]_Y \neq [g' \cdot \pi]_Y$. If $g, g' \in D_n$ with $g \neq g'$ then $[g \cdot \pi]_Y = [g' \cdot \pi]_Y$ holds if and only if Y is bipartite.*

The proof, which can be found in [10], is by contradiction. We remark that if Y is bipartite if and only if $|\{[g \cdot \pi]_Y : g \in D_n\}| = 2n - 1$.

4.2. Source-Sink Operations and Reflections of Acyclic Orientations. In this section we show how the component structure of $C(Y)$ is precisely captured through

source-sink operations on acyclic orientations. The bijection in (2.6) identifies $[\pi]_Y$ with $O_Y^\pi \in \text{Acyc}(Y)$. For any $\pi \in [\pi']_Y$, the orientation $O_Y^{\sigma_1(\pi)}$ is constructed from O_Y^π by converting vertex π_1 from a source to a sink. Following [16] we call such a conversion a *source-sink operation* or a *click*. It can be easily verified that this gives rise to an equivalence relation \sim_κ on $\text{Acyc}(Y)$. More precisely, two orientations $O_Y, O'_Y \in \text{Acyc}(Y)$ where O_Y can be transformed into O'_Y by a sequence of clicks are said to be κ -equivalent. This observation along with Theorem 3.3 shows that permutations that belong to κ -equivalent acyclic orientations induce cycle equivalent SDSs. By construction, the source-sink operation precisely encodes adjacency in the graph $C(Y)$, and the connected components are in 1–1 correspondence with the κ -equivalence classes. Therefore, the number of equivalence classes in $\text{Acyc}(Y)$ under the source-sink relation equals $\kappa(Y)$, and is thus an upper bound for the number of cycle equivalent permutation SDS maps $[\mathfrak{F}_Y, \pi]$ for a fixed sequence \mathfrak{F}_Y .

If $K = \mathbb{F}_2$ then Proposition 3.4 shows that reflections of update orders also induce cycle equivalent SDSs. On the level of acyclic orientations this corresponds to reversing all orientations. Through the bijection (2.6) this identifies O_Y^π with the reverse orientation $O_Y^{\rho(\pi)}$, the unique orientation that satisfies $O_Y^\pi(\{i, j\}) \neq O_Y^{\rho(\pi)}(\{i, j\})$ for every $\{i, j\} \in e[Y]$. If two acyclic orientations are related by a sequence of source-sink operations and reflections, then they are said to be δ -equivalent.

The notion of κ - and δ -equivalence carries over naturally to update orders as well. Two update orders in S_Y belonging to \sim_Y classes on the same connected component in $C(Y)$ [resp. $D(Y)$] are called κ -equivalent [resp. δ -equivalent]. For two κ -equivalent update orders π and π' , there is a sequence of adjacent non-edge transpositions and cyclic shifts that map π to π' . This is simply a consequence of the definition of S_Y/\sim_Y and $C(Y)$.

We remark that from here there is a close connection to the structure of conjugacy classes of Coxeter elements, something we explain more in Section 7. The case of $K = \mathbb{F}_2$ and reflections does not seem to play any role in Coxeter theory.

5. ENUMERATION FOR $\kappa(Y)$ AND $\delta(Y)$

It is not difficult to show that $\delta(Y)$ may be characterized in terms of $\kappa(Y)$.

Proposition 5.1 ([10]). *Let Y be a connected graph. If Y is not bipartite then $\delta(Y) = \frac{1}{2}\kappa(Y)$. If Y is bipartite then $\delta(Y) = \frac{1}{2}(\kappa(Y) + 1)$.*

The proof uses the fact that $\rho: S_Y \rightarrow S_Y$ extends to an involution

$$(5.1) \quad \rho^*: \text{Acyc}(Y)/\sim_\kappa \rightarrow \text{Acyc}(Y)/\sim_\kappa .$$

The result now follows since ρ^* has no fixed points if Y is not bipartite, and has precisely one fixed point if Y is bipartite. As a corollary, a connected graph is bipartite if and only if $\kappa(Y)$ is odd. In light of Proposition 5.1 we focus on the computation of $\kappa(Y)$ in the following. It can be shown that $\kappa(Y)$ does not depend on bridge edges, i.e., edges not contained in a cycle.

Proposition 5.2 ([9]). *If Y is the disjoint union of graphs Y_1 and Y_2 , or if Y is a graph with $e = \{v, w\}$ a bridge connecting the subgraphs Y_1 and Y_2 , then*

$$(5.2) \quad \kappa(Y) = \kappa(Y_1)\kappa(Y_2) .$$

For the computation of $\kappa(Y)$ we may therefore assume that Y is connected, and that every edge is a cycle-edge. Note that for the empty graph on n vertices E_n we have $\kappa(E_n) = 1$ since $\alpha(E_n) = 1$. The following corollary is immediate.

Corollary 5.3. *Let Y be a forest. Then $\kappa(Y) = \delta(Y) = 1$.*

From Corollary 5.3 we get the following perhaps surprising results on dynamics of SDSs over trees:

Proposition 5.4. *Let Y be a forest and \mathfrak{F}_Y be a sequence of Y -local functions. Then all permutation SDS maps $[\mathfrak{F}_Y, \pi]$ are cycle equivalent.*

The proof is clear since κ -equivalent update orders induce cycle equivalent systems. So in other words, when Y is a forest, all permutation SDSs of the form $[\mathfrak{F}_Y, \pi]$ for fixed \mathfrak{F}_Y share the same cycle configuration. This result may not be that significant if the SDS only has fixed points, or only has a small number of periodic points. However, for other functions, such as invertible ones, it is very powerful. The parity functions $\text{par}_k: \mathbb{F}_2^k \rightarrow \mathbb{F}_2$ are defined as $\text{par}(\mathbf{y}) = \sum_i y_i$, modulo 2, and are invertible for every graph Y (see [11]). Let Par_Y be the sequence of Y -local functions induced by the parity vertex functions.

Corollary 5.5. *If Y is a forest then for any $\pi, \sigma \in S_Y$ the maps $[\text{Par}_Y, \pi]$ and $[\text{Par}_Y, \sigma]$ are dynamically equivalent.*

The same result holds for the logical negation of the parity function, which is also invertible.

We now give examples of the computation of κ . Even though some of these results may be derived as special cases of more general results, they are needed for central examples in Section 6. We begin with a result for $\kappa(Y \oplus v)$, the vertex-join of Y and the vertex v . Recall that the graph $Y \oplus v$ is constructed from Y by adding to Y the vertex v and joining this new vertex to each vertex of Y .

Proposition 5.6. *If Y is a graph with $e[Y] \neq \emptyset$, then*

$$(5.3) \quad \kappa(Y \oplus v) = 2\delta(Y \oplus v) = \alpha(Y) .$$

Proof. Each κ -class of $\text{Acyc}(Y \oplus v)$ contains a unique acyclic orientation where v is a source [9]. It follows that there is a bijection between $\text{Acyc}(Y \oplus v)/\sim_\kappa$ and $\text{Acyc}(Y)$, hence (5.3). The complete graph on n vertices is simply the vertex-join of the complete graph on $n - 1$ vertices, and thus we get the following corollary. \square

Corollary 5.7. *Let K_n denote the complete graph on n vertices. For $n \geq 2$ we have $\kappa(K_n) = (n - 1)!$.*

Proof. There are $2^{\binom{n}{2}}$ orientations of K_n , and by the bijection in (2.6), precisely $\alpha(K_n)$ of these are acyclic, and this is equal to the number of components of the update graph $U(K_n)$. Since $U(K_n)$ consists of the $n!$ singleton vertices in S_Y , $\alpha(K_n) = n!$. By Proposition 5.6, $\kappa(K_n) = \alpha(K_{n-1}) = (n-1)!$. \square

The quantity $\kappa(Y)$ is in fact a *Tutte-Grothendieck invariant*:

Theorem 5.8 ([9]). *Let e be a cycle-edge of Y . Then*

$$(5.4) \quad \kappa(Y) = \kappa(Y'_e) + \kappa(Y''_e),$$

where Y'_e is the graph obtained from Y by deleting e , and Y''_e is the graph obtained from Y by contracting e .

The proof of Theorem 5.8 is quite involved, and along with Proposition 5.2, it implies that $\kappa(Y) = T(Y, 1, 0)$, where $T(Y, x, y)$ is the Tutte polynomial [17]. In contrast, it is well-known that the number of acyclic orientations of a graph satisfies $\alpha(Y) = T(Y, 2, 0)$.

A graph Y has an n -handle if it is of the form $Y = Y' \cup \text{Circ}_n$ where Y' and Circ_n share precisely one edge.

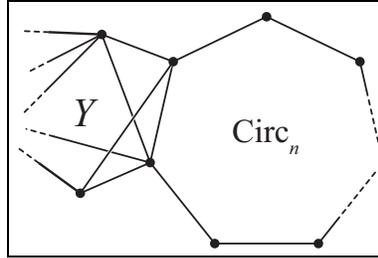


FIGURE 4. A graph Y with an n -handle.

Proposition 5.9. *Let Y be a graph with an n -handle where $Y = Y' \cup \text{Circ}_n$. Then*

$$(5.5) \quad \kappa(Y) = (n-1)\kappa(Y').$$

Proof. Let $e' = \{v, v'\}$ be the edge shared by Y' and Circ_n and let e be the edge in Circ_n incident with v . By applying Theorem 5.8 and Proposition 5.2 for bridge edges we obtain

$$\kappa(Y' \cup \text{Circ}_n) = \kappa(Y') + \kappa(Y' \cup \text{Circ}_{n-1}).$$

Equation (5.5) follows through repeated applications of this process. \square

As a simple, special case of Proposition 5.9 we obtain $\kappa(\text{Circ}_n) = n-1$. Just take Y' to be the graph with vertex set $v[Y'] = \{1, n\}$ and edge set $\{\{1, n\}\}$ in Proposition 5.9.

6. $\kappa(Y)$ AS A COMPLEXITY MEASURE

The number of possible orbit structures that one can obtain by varying the update order is a natural measure for system complexity. As we have shown, $\kappa(Y)$ is a general upper bound for this number, and so is $\delta(Y)$ in the case of binary states. Since these bounds are graph measures we can characterize complexity in terms of the GDS base graphs. As we have seen, bridge edges do not contribute to periodic orbit variability at all, and so it suffices to consider the cycles of the graph. As can be seen in the case of Circ_n , increasing the size of a cycle does not contribute much, e.g. $\kappa(\text{Circ}_{n+1}) = \kappa(\text{Circ}_n) + 1$. However, from the result on graphs with handles it follows that even the addition of a minimal handle *doubles* the measure κ , i.e. $\kappa(Y \cup \text{Circ}_3) = 2\kappa(Y)$, where Y and Circ_3 share precisely one edge. The following example shows the effect on complexity that results from increasing the radius of the rules for elementary cellular automata.

Example 6.1 (CA rule radius vs. periodic orbit complexity). We have seen that $\kappa(\text{Circ}_n) = n - 1$. Thus, for any fixed sequence of radius-1 vertex functions the number of distinct periodic orbit configurations is $O(n)$. This changes dramatically for radius-2 rules. In this case the GDS base graph is $\text{Circ}_{n,2}$ with

$$v[\text{Circ}_{n,2}] = \{1, 2, \dots, n\}, \quad \text{and} \quad e[\text{Circ}_{n,2}] = \{\{i, j\} \mid 1 \leq |i - j| \leq 2\},$$

with index arithmetic modulo n . The auxiliary graph $\text{Circ}'_{n,2}$ is obtained from $\text{Circ}_{n,2}$ by deleting the edge $\{2, n\}$. The case $n = 7$ is illustrated in Figure 5.

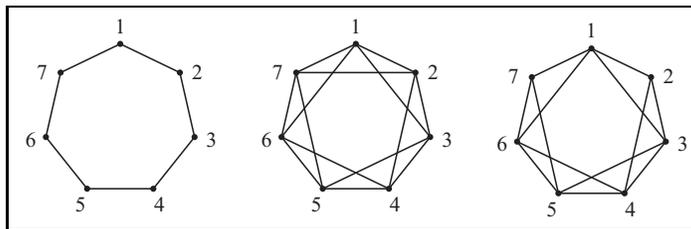


FIGURE 5. From left to right: The graphs Circ_7 , $\text{Circ}_{7,2}$ and $\text{Circ}'_{7,2}$.

For simplicity we set $g_n = \kappa(\text{Circ}_{n,2})$ and $c_n = \kappa(\text{Circ}'_{n,2})$. Successive uses of the recurrence (5.4) with edges $e_1 = \{1, n\}$ and $e_2 = \{1, n - 1\}$ for both $\text{Circ}_{n,2}$ and $\text{Circ}'_{n,2}$ gives

$$c_n = c_{n-1} + 2c_{n-2} + 2^{n-2}, \quad \text{and} \quad g_n = g_{n-2} + c_n + 2c_{n-2},$$

where $c_5 = 18$, $c_6 = 46$, $g_5 = 24$, and $g_6 = 64$. These recurrence relations are straightforward to solve with

$$\begin{aligned} \kappa(\text{Circ}'_{n,2}) &= [(3n - 5)2^n - 4(-1)^n]/18 \quad \text{and} \\ \kappa(\text{Circ}_{n,2}) &= [(2n - 6)2^n + 9 - (2n - 3)(-1)^n]/6. \end{aligned}$$

Thus, by increasing the rule radius from 1 to 2 we see that the number of distinct periodic orbit configurations is $O(n \cdot 2^n)$. The corresponding bounds for δ are easily obtained from Proposition 5.1.

We have seen how non-trivial symmetries in the base graph give rise to dynamically equivalent SDS maps when the vertex functions are $\text{Aut}(Y)$ -invariant. Since dynamical equivalence implies cycle equivalence we can construct a bound $\bar{\kappa}(Y)$ in the same manner as for $\bar{\alpha}(Y)$. This bound $\bar{\kappa}(Y)$ thus reflects the added cycle equivalence that are due to symmetries and that arise for $\text{Aut}(Y)$ -invariant vertex functions.

We close with an example that illustrates this and the results of Theorem 5.8, and Propositions 5.2 and 5.6 and 5.9.

Example 6.2. Let $Y = Q_2^3$ be the binary 3-cube, which has automorphism group isomorphic to $S_4 \times \mathbb{Z}_2$. It is shown in [3] that $\alpha(Q_2^3) = 1862$ and that $\bar{\alpha}(Q_2^3) = 54$. Thus, there are at most 1862 functionally nonequivalent permutation SDSs over Q_2^3 for a fixed sequence of vertex functions. Likewise, there are at most 54 dynamically nonequivalent $\text{Aut}(Q_2^3)$ -invariant permutation SDSs. It is known that the bound $\bar{\alpha}(Q_2^3)$ is sharp, since it is realized for SDSs induced by, e.g. the nor_4 -function.

The number of cycle equivalence classes is bounded above by $\kappa(Q_2^3)$, and from the recursion relation (5.4) we get (with some foresight at each step)

$$\begin{aligned} \kappa(\text{cube}) &= \kappa(\text{cube}_1) + \kappa(\text{cube}_2) = \kappa(\text{cube}_1) + 2\kappa(\text{cube}_3) + \kappa(\text{cube}_4) \\ &= \kappa(\text{cube}_1) + 2\kappa(\text{cube}_5) + 2\kappa(\text{cube}_6) + \kappa(\text{cube}_7) + \kappa(\text{cube}_8) \\ &= \kappa(\text{cube}_1) + 4\kappa(\text{cube}_9) + 2\kappa(\text{cube}_{10}) + \kappa(\text{cube}_{11}) + \kappa(\text{cube}_{12}) \\ &= 27 + 64 + 16 + 12 + 14 = 133, \end{aligned}$$

where Propositions 5.6 and 5.9 were used in the last step. Since Q_2^3 is bipartite we also derive $\delta(Q_2^3) = (133 + 1)/2 = 67$, and thus in the case of $K = \mathbb{F}_2$ there are at most 67 cycle classes for a fixed sequence of vertex functions. Straightforward (but somewhat lengthy) calculations show that $\bar{\kappa}(Q_2^3) = \bar{\delta}(Q_2^3) = 8$. In conclusion, we have

$$\alpha(Q_2^3) = 1862, \quad \bar{\alpha}(Q_2^3) = 54, \quad \kappa(Q_2^3) = 133, \quad \delta(Q_2^3) = 67, \quad \bar{\kappa}(Q_2^3) = \bar{\delta}(Q_2^3) = 8.$$

Thus if \mathfrak{F}_Y is a sequence of $\text{Aut}(Q_2^3)$ -invariant Y -local functions, then there are at most eight different periodic orbit configurations for permutation SDS maps $[\mathfrak{F}_Y, \pi]$ up to isomorphism. Moreover, because $\bar{\kappa}(Q_2^3) = \bar{\delta}(Q_2^3)$ taking vertex states from $K = \mathbb{F}_2$ does not improve this bound.

This example is only meant as an illustration, and a systematic treatment incorporating the analysis of the functions $\bar{\kappa}$ and $\bar{\delta}$ for general graphs will be pursued elsewhere.

7. SUMMARY

In this paper we have shown how shifts and reflections of update orders give rise to sequential dynamical systems with isomorphic periodic orbit configurations. We have also shown how to bound the number of periodic orbit configurations, and have derived several

properties of this bound $\kappa(Y)$. For binary states we have shown how $\delta(Y)$ applies to give a sharper bound. Both quantities κ and δ serve as measures for dynamical complexity.

We also have $\kappa(Y) = T(Y, 1, 0)$ where T denotes the Tutte polynomial. There are other mathematical quantities counted by $T(Y, 1, 0)$, and thus by $\kappa(Y)$. For example, the set $\text{Acyc}_v(Y)$ consisting of all the acyclic orientation of Y with v as a unique source is also counted by $\kappa(Y)$, see [7]. In fact, for each κ -equivalence class there is unique acyclic orientation with v as the only source. This allows one to construct a complete set of representatives for permutations realizing the possible periodic orbit configurations, see [9].

The notion of source-to-sink conversions also shows up in the context of Coxeter theory (see, e.g. [6] for definitions). For a Coxeter group with Coxeter graph Y the number of conjugacy classes of Coxeter elements (see [15]) is also bounded above by $\kappa(Y)$, e.g. [16]. In general it is not known if $\kappa(Y)$ is a sharp bound, but it is known for special classes of graphs such as Circ_n as shown by Shi in [16]. This connection between Coxeter theory and SDSs could potentially be very helpful in further exploring the properties of asynchronous GDSs.

In this paper we have not explored the question of when κ (and δ when $K = \mathbb{F}_2$) is a sharp bound. That is, for an arbitrary graph Y does there exist a sequence of vertex function whose number of non-equivalent orbit configurations equals $\kappa(Y)$? Proving this would require one to construct such functions for any given graph. We have also omitted computational aspects related to cycle equivalence. Given two SDSs, what is the computational complexity of determining if they are cycle equivalent? Related questions have been asked for, e.g. fixed point reachability in [2], but see also [1] for similar questions. Additional future work includes extending our results from permutations update orders to general word update orders as well as further exploring the effects of symmetries in the graph and the computation of the bounds $\bar{\kappa}$ and $\bar{\delta}$ as illustrated in Example 6.2.

Acknowledgments

The first author would like to thank Jon McCammond for many helpful discussions. Both authors are grateful to the NDSSL group at Virginia Tech for the support of this research. This work was partially supported by Fields Institute in Toronto, Canada.

REFERENCES

- [1] C. L. Barrett, H. B. Hunt III, M. V. Marathe, S. S. Ravi, D. J. Rosenkrantz, and R. E. Stearns. Complexity of reachability problems for finite discrete sequential dynamical systems. *J. Computer and System Sciences*, 72:1317–1345, 2006.
- [2] C. L. Barrett, H. H. Hunt III, M. V. Marathe, S. S. Ravi, D. Rosenkrantz, R. Stearns, and P. Tasic. Gardens of eden and fixed point in sequential dynamical systems. In *Discrete Models: Combinatorics, Computation and Geometry*, pages 95–110, 2001.
- [3] C. L. Barrett, H. S. Mortveit, and C. M. Reidys. Elements of a theory of simulation III, equivalence of sds. *Applied Mathematics and Computation*, 122:325–340, 2001.

- [4] C. L. Barrett, H. S. Mortveit, and C. M. Reidys. Elements of a theory of simulation IV: Fixed points, invertibility and equivalence. *Applied Mathematics and Computation*, 134:153–172, 2003.
- [5] C. L. Barrett and C. M. Reidys. Elements of a theory of simulation I: Sequential CA over random graphs. *Applied Mathematics and Computation*, 98(2–3):241–259, 1999.
- [6] A. Björner and F. Brenti. *Combinatorics of Coxeter Groups*, volume 231 of *GTM*. Springer Verlag, 2005.
- [7] E. Gioan. Enumerating degree sequences in digraphs and a cycle-cocycle reversing system. *European Journal of Combinatorics*, 28:1351–1366, 2007.
- [8] U. Karaoz, T. Murali, S. Letovsky, Y. Zheng, C. Ding, C. R. Cantor, and S. Kasif. Whole-genome annotation by using evidence integration in functional-linkage networks. *Proceedings of the National Academy of Sciences*, 101(9):2888–2893, 2004.
- [9] M. Macauley and H. S. Mortveit. On enumeration of conjugacy classes of Coxeter elements. *Proceedings of the American Mathematical Society*, 2008. Accepted. arXiv:0711.1140.
- [10] M. Macauley and H. S. Mortveit. Equivalences on acyclic orientations. Submitted. arXiv:0709.0291.
- [11] H. S. Mortveit and C. M. Reidys. Discrete, sequential dynamical systems. *Discrete Mathematics*, 226:281–295, 2001.
- [12] S. Pautz. An algorithm for parallel S_n sweeps on unstructured meshes. *Nuclear Science and Engineering*, 140:111–136, 2002.
- [13] C. M. Reidys. Acyclic orientations of random graphs. *Advances in Applied Mathematics*, 21(2):181–192, 1998.
- [14] C. M. Reidys. Sequential dynamical systems over words. *Annals of Combinatorics*, 10:481–498, 2006.
- [15] J. Y. Shi. The enumeration of Coxeter elements. *Journal of Algebraic Combinatorics*, 6:161–171, 1997.
- [16] J. Y. Shi. Conjugacy relation on Coxeter elements. *Advances in Mathematics*, 161:1–19, 2001.
- [17] W. T. Tutte. A contribution to the theory of chromatic polynomials. *Canadian Journal of Mathematics*, 6:80–91, 1954.