9-2009

Update Sequence Stability in Graph Dynamical Systems

Matthew Macauley
Clemson University, macaule@clemson.edu

Henning S. Mortveit
Viginia Tech

Recommended Citation

Please refer to this page for the recommended citation: https://arxiv.org/abs/0909.1723

UPDATE SEQUENCE STABILITY IN GRAPH DYNAMICAL SYSTEMS

MATTHEW MACAULEY AND HENNING S. MORTVEIT

ABSTRACT. In this article, we study finite dynamical systems defined over graphs, where the functions are applied asynchronously. Our goal is to quantify and understand stability of the dynamics with respect to the update sequence, and to relate this to structural properties of the graph. We introduce and analyze three different notions of update sequence stability, each capturing different aspects of the dynamics. When compared to each other, these stability concepts yield vastly different conclusions regarding the relationship between stability and graph structure, painting a more complete picture of update sequence stability.

1. INTRODUCTION

A graph dynamical system consists of (i) an underlying graph Y where each vertex has a state from a finite set K, (ii) a sequence of vertex functions that updates each vertex state as a function of its neighbors, and (iii) an update sequence that prescribes how to compose the functions to get the global dynamical system map. Examples of graph dynamical systems in the literature include systems with synchronous update such as cellular automata [18] and Boolean networks [17], those with asynchronous updates, e.g., sequential dynamical systems [16] and functional linkage networks [9], and many probabilistic versions of the above systems. In this article, we focus on systems with asynchronous schedules, with the broad goal of understanding the sensitivity of the global dynamics with respect to changes in the update sequence.

First, it is necessary to define what update order stability means, and how to quantify such a concept. There are many possible answers to this question, and there are vastly different ways to do this based on existing literature. After a brief introduction of the main concepts in Section 2, we devote the following three sections to presentations of three different aspects of update sequence stability, each focusing on different properties of the dynamics. In Section 3, we look at the structure of the phase space and its robustness to changes in the update sequence. In Section 4, we analyze the number of attractor cycles that can be reached from a given initial state under different asynchronous update sequences. Finally, in Section 5, we look at the collection of periodic states and how they change (setwise) under different update sequences.

Many of the concepts described above are scattered throughout the literature, usually studied outside of the theme of update sequence stability. This paper is the first of its kind to bring these ideas together with this common goal in mind. It is unrealistic to expect to develop a complete unifying theory of update order stability due to the sheer size, diversity, and inherent randomness of complex systems. However, there is still much be gained from such a study. One of the themes of this paper is understanding how certain stability properties are generally correlated

with physical properties of the underlying graph, such as edge density. For example, are systems over sparse graphs or dense graphs more stable? Not surprisingly, different notions of stability can lead to vastly different conclusions. We hope that this paper will clarify many of these ideas and set the groundwork for future investigations on the stability of asynchronous finite dynamical systems.

2. PRELIMINARIES

Let Y be an undirected graph with vertex set $\mathrm{v}[Y] = \{1, \ldots, n\}$. Each vertex i has a state $x_i \in K$, where K is a finite set (frequently $K = \mathbb{F}_2 = \{0, 1\}$), and a *vertex function* f_i that updates its state as a function of the states of its neighbors (itself included). When applying these functions asynchronously, it is convenient to encode f_i as a Y-*local function* $F_i \colon K^n \longrightarrow K^n$, where

$$F_i(x_1, \ldots, x_n) = (x_1, \ldots, x_{i-1}, f_i(x_1, \ldots, x_n), x_{i+1}, \ldots, x_n) \ .$$

In the remainder of this paper, we will focus on a class of graph dynamical systems with asynchronous update schedules, called *sequential dynamical systems* [16]. These systems have a prescribed update schedule dictated by a sequence of vertices $\omega \in W_Y$, where W_Y is the set of non-empty words over $\mathrm{v}[Y]$.

Definition 2.1. A *sequential dynamical system* (SDS) is a triple $(Y, \mathfrak{F}_Y, \omega)$, where Y is an undirected graph, $\mathfrak{F}_Y$ a sequence of Y-local functions, and $\omega \in W_Y$ a length-m update sequence. The *SDS map* is the composition

$$[\mathfrak{F}_Y, \omega] = F_{\omega_m} \circ F_{\omega_{m-1}} \circ \cdots \circ F_{\omega_2} \circ F_{\omega_1} \ .$$

Henceforth, we will primarily restrict our attention to the case where $\omega \in S_Y$, the set of permutations of $\mathrm{v}[Y]$, i.e., words where each vertex appears precisely once. To emphasize a permutation update sequence, we will use π instead of ω. All of the concepts carry over to general asynchronous systems under slight modification, though not much additional insight is gained in doing so, whereas the notation becomes more involved and the main concepts obscured.

The *phase space* of a finite dynamical system $\phi \colon K^n \longrightarrow K^n$ is the directed graph $\Gamma(\phi)$ with vertex set K^n and edges $(x, \phi(x))$ for each $x \in K^n$. Clearly, $\Gamma(\phi)$ consists of disjoint cycles (periodic states), with directed trees (transient states) attached.

3. EQUIVALENCE OF DYNAMICS

Any reasonable notion of stability of dynamics must be coupled with a notion of equivalence of dynamics. For dynamical systems, a natural place to begin is with the structure of the phase space, since this encodes the global dynamics. We present here several different types of equivalence, which vary in what is meant by having equivalent structure.

Definition 3.1. Let $\phi, \psi \colon K^n \longrightarrow K^n$ be finite dynamical systems. Then ϕ and ψ are:
  (1) *functionally equivalent* if $\phi = \psi$.
  (2) *dynamically equivalent* if $\phi \circ h = h \circ \psi$ for some bijection $h \colon K^n \longrightarrow K^n$.
  (3) *cycle equivalent* if $\phi|_{\mathsf{Per}(\phi)} \circ h = h \circ \psi|_{\mathsf{Per}(\psi)}$ for some bijection $h \colon \mathsf{Per}(\phi) \longrightarrow \mathsf{Per}(\psi)$.

Clearly, ϕ and ψ are functionally equivalent iff their phase spaces are identical, dynamically equivalent iff their phase spaces are isomorphic, and cycle equivalent iff their phase spaces are isomorphic when restricted to the periodic states. For a fixed sequence $\mathfrak{F}_Y$ of Y-local functions, different update sequences give rise to different SDS maps, and each notion of equivalence mentioned above partitions the set of all such maps into equivalence classes. The number of SDS maps up to equivalence obtainable by varying the update sequence is a measure of stability. As we will see, each type of equivalence of dynamics corresponds with a combinatorial equivalence relation on the set of update sequences S_Y (or W_Y), which in turn is strongly tied to the structure of the underlying graph Y. Thus, knowledge of the structure of these equivalence classes is central to quantifying update sequence stability, and its relationship to the base graph. We will summarize this below.

3.1. Functional Equivalence.
Every update sequence $\pi \in S_Y$ defines a partial ordering $<_\pi$ of $v[Y]$, and this is naturally represented via the set $\mathsf{Acyc}(Y)$ of acyclic orientations on Y. Explicitly, π gives rise to the acyclic orientation O_π of Y obtained by orienting each edge $\{i, j\}$ of Y as (i, j) if i occurs before j in π and as (j, i) otherwise. The partial order $<_\pi$ is defined by $i <_\pi j$ iff there is a directed path from i to j in O_Y. If $\pi, \sigma \in S_Y$ are linear extensions of the same partial ordering, then $[\mathfrak{F}_Y, \pi] = [\mathfrak{F}_Y, \sigma]$, and we say that $\pi \sim_\alpha \sigma$. In fact, we have a bijection

$$\phi_\alpha \colon S_Y/\!\sim_\alpha \longrightarrow \mathsf{Acyc}(Y) \ ,$$

hence $\alpha(Y) := |\mathsf{Acyc}(Y)|$ is an upper bound for the number of distinct (permutation) SDS maps over $\mathfrak{F}_Y$ *up to functional equivalence*. This bound is sharp (see [2]), and it is counted by the Tutte polynomial, as $\alpha(Y) = T_Y(2, 0)$. Additionally, the bijection ϕ_α allows us to construct a transversal of update sequences from S_Y, rather than considering all $n!$ update sequences.

3.2. Dynamical Equivalence.
Let $\mathsf{Aut}(Y)$ denote the automorphism group of Y, and define the S_n-action on K^n by $\sigma(x_1, \ldots, x_n) = (x_{\sigma^{-1}(1)}, \ldots, x_{\sigma^{-1}(n)})$. The group $\mathsf{Aut}(Y)$ acts on the set $S_Y/\!\sim_\alpha$ by $\gamma \cdot [\pi]_\alpha = [\gamma\pi]_\alpha$. If $\mathfrak{F}_Y$ is $\mathsf{Aut}(Y)$-invariant (i.e., $\gamma \circ F_v = F_{\gamma(v)} \circ \gamma$ for each local function, and each $\gamma \in \mathsf{Aut}(Y)$), then two SDS maps over update sequences in the same orbit are conjugate, by $\gamma \circ [\mathfrak{F}_Y, \pi] \circ \gamma^{-1} = [\mathfrak{F}_Y, \gamma\pi]$ (see [16]). This puts an equivalence relation on S_Y (and hence on $\mathsf{Acyc}(Y)$) that we denote by $\sim_{\bar{\alpha}}$. Since conjugate maps are dynamically equivalent, the function $\bar{\alpha}(Y) := |\mathsf{Acyc}(Y)/\!\sim_{\bar{\alpha}}|$ is an upper bound for the number of SDS maps over $\mathfrak{F}_Y$ *up to dynamical equivalence*. This quantity, which counts the number of orbits in $\mathsf{Acyc}(Y)$ under $\mathsf{Aut}(Y)$, can be computed though Burnside's lemma, as

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

Here, $\langle\gamma\rangle \backslash Y$ denotes the *orbit graph* of the cyclic group $G = \langle\gamma\rangle$ and Y, (see [2, 3]). We believe this bound is sharp, and have shown this for special graph classes, but have no general proof.

3.3. Cycle Equivalence.
To see how cycle equivalence arises in the SDS setting, we first define a way to modify an acyclic orientation, called a *source-to-sink* operation, or a *click*. This consists of choosing a source vertex and reversing the orientations of all incident edges. The reflexive transitive closure of this puts an equivalence relation on $\mathsf{Acyc}(Y)$ (and on S_Y), denoted $\sim_\kappa$. Thus,

two acyclic orientations related by a sequence of clicks are κ-equivalent. In [15], it is shown that if $\pi \sim_\kappa \sigma$, then $[\mathfrak{F}_Y, \pi]$ and $[\mathfrak{F}_Y, \sigma]$ are cycle equivalent. Therefore, $\kappa(Y) := |\mathsf{Acyc}(Y)/\sim_\kappa|$ is an upper bound for the number of SDS maps over $\mathfrak{F}_Y$ *up to cycle equivalence*. This quantity is also counted by the Tutte polynomial, by $\kappa(Y) = T_Y(1, 0)$ (see [14]). As with α-equivalence, there is a nice transversal for κ-equivalence, arising from the bijection

$$(3.1) \qquad \phi_{\kappa,v} \colon \mathsf{Acyc}_v(Y) \longrightarrow \mathsf{Acyc}(Y)/\sim_\kappa \ .$$

where $\mathsf{Acyc}_v(Y)$ is the set of acyclic orientations of Y where vertex $v \in \mathrm{v}[Y]$ is the unique source.

If additionally, $\mathsf{Aut}(Y)$ is nontrivial, we can use the action of this group on $\mathsf{Acyc}(Y)/\sim_\kappa$ to impose a coarser equivalence relation $\sim_{\bar\kappa}$ on $\mathsf{Acyc}(Y)$. This is clear, since dynamically equivalent SDS maps are trivially cycle equivalent. The quantity $\bar\kappa(Y) := |\mathsf{Acyc}(Y)/\sim_{\bar\kappa}|$, which is the number of orbits in $\mathsf{Acyc}(Y)/\sim_\kappa$ under $\mathsf{Aut}(Y)$, is therefore a stronger upper bound for the number of SDS maps over $\mathfrak{F}_Y$ up to cycle equivalence when $\mathfrak{F}_Y$ is $\mathsf{Aut}(Y)$-invariant.

3.4. **Summary of Equivalence.** To summarize, for each of the following notions of equivalence of SDS maps, there is a corresponding equivalence relation on S_Y (and hence on $\mathsf{Acyc}(Y)$):

- *Functional equivalence*. Relation: $\sim_\alpha$ with transversal $\mathsf{Acyc}(Y)$ via ϕ_α;
- *Dynamical equivalence*. Relation: $\sim_{\bar\alpha}$ with transversal $\mathsf{Acyc}(Y)/\sim_{\bar\alpha}$;
- *Cycle equivalence*. Relation: $\sim_\kappa$ and transversal $\mathsf{Acyc}_v(Y)$ via $\phi_{\kappa,v}$ for $v \in \mathrm{v}[Y]$.

In each of these three cases, update sequences that are equivalent give rise to SDS maps with equivalent dynamics. Additionally, it is clear that functionally equivalent systems are dynamically equivalent, which in turn are cycle equivalent. In the general case where the functions $\mathfrak{F}_Y$ are not $\mathsf{Aut}(Y)$-invariant, or when $\mathsf{Aut}(Y) = 1$ (which is true of almost all random graphs [6]), $\bar\alpha$-equivalence reduces down to α-equivalence. The difference between α- and $\bar\alpha$-equivalence only captures special symmetries in the graph. In light of this, we will focus on $\alpha(Y)$ and $\kappa(Y)$ for our discussion of stability, though $\bar\alpha(Y)$ was mentioned for sake of completeness.

Since the functions $\alpha(Y)$ and $\kappa(Y)$ bound the number of SDS maps obtainable up to equivalence under different update sequences, they can be considered as measures of stability. Computing these quantities is in general, NP-hard, since they are evaluations of the Tutte polynomial [7]. However, even though we can not compute these functions exactly for large networks, an understanding of the mathematics involved provides valuable insight about update order stability. For example, consider two extreme cases: (i) Y is an n-vertex tree, and (ii) $Y = K_n$, the complete graph on n vertices. In the first case, $\kappa(Y) = 1$, and in the latter, $\kappa(Y) = (n-1)!$. Moreover, since α and κ are Tutte-Grothendieck invariants, if $Z \leq Y$ then $\alpha(Z) \leq \alpha(Y)$ and $\kappa(Z) \leq \kappa(Y)$, i.e., α and κ monotonically increase as additional edges are introduced to the base graph. Two intermediate cases (between the tree and complete graph) were studied in [15] – the graphs corresponding to the radius-1 and radius-2 cellular automata rules. For radius-1 rules (functions over the the circular graph Circ_n), there are at most $\kappa(Y) = O(n)$ cycle structures for (permutation) SDS maps. In contrast, an SDS with a radius-2 rule has base graph $Y = \mathsf{Circ}_{n,2}$, the circular graph where every vertex is additionally connected to its distance-2 neighbors. In this case, $\kappa(Y) = O(n \cdot 2^n)$. Though we cannot compute $\kappa(Y)$ explicitly for most graphs, we see a clear correlation between edge density of the base graph and the stability of a dynamical

system over it. Thus, using α or κ-equivalence as a notion of stability, we can make the general statement:

Update order stability and edge density are positively correlated.

3.5. **An Example.** We conclude this section with an example illustrating the difference between the various notions of equivalence described in this section.

Example 3.2. Let $Y = \mathsf{Circ}_4$, with vertex functions $\mathsf{nor}_3\colon \mathbb{F}_2^3 \longrightarrow \mathbb{F}_2$ where $\mathsf{nor}_3(0,0,0) = 1$ and $\mathsf{nor}_3(x,y,z) = 1$ for all other arguments. Using update sequences $\pi = (1,2,3,4)$, $\pi' = (1,4,2,3)$ and $\pi'' = (1,3,2,4)$, the phase spaces are shown in Figure 1. None of these SDS
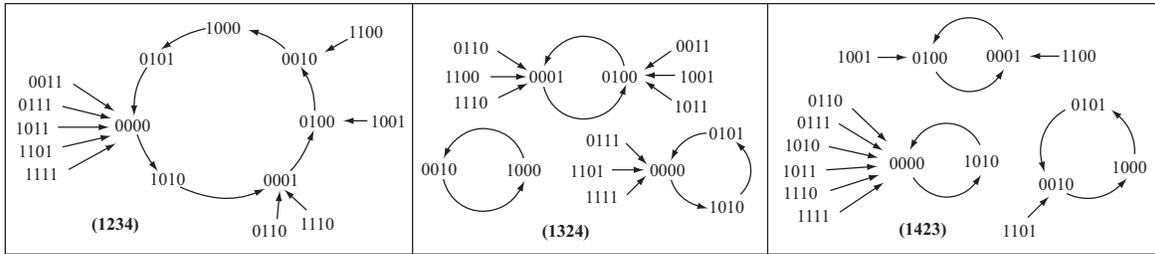

FIGURE 1. The phase spaces of Example 3.2.

maps are functionally or dynamically equivalent, but $[\mathsf{Nor}_Y, \pi']$ and $[\mathsf{Nor}_Y, \pi'']$ are cycle equivalent, as illustrated in the two rightmost phase spaces in Figure 1. In fact, it is elementary to compute $\bar{\kappa}(\mathsf{Circ}_4) = 2$, thus the two cycle configurations in Figure 1 are the *only* possible such configurations up to isomorphism when $Y = \mathsf{Circ}_4$ and $K = \mathbb{F}_2$.

4. LIMIT SETS AND REACHABILITY

Another natural measure for update sequence stability is a quantification of how many attractor cycles can be reached from a certain initial state. There are many ways to develop such a measure – one can compute the average number of periodic states reachable from a random starting point, the maximum number of reachable periodic states, the number of reachable attractor cycles, possibly normalized for size, etc. For general complex networks, such computations are intractable, and Monte Carlo type methods may be an alternative. However, it is at times possible to estimate or bound such quantities asymptotically. In this section, we will summarize some recent results of one these measures that was developed to study a the robustness of a gene annotation algorithm that basically uses a threshold SDS [9].

Threshold functions are common in discrete dynamical system models [10]. A *Boolean threshold function* on k arguments is a function $f_{k,m}\colon \mathbb{F}_2^k \longrightarrow \mathbb{F}_2$ such that $f(x_1, \ldots, x_k) = 1$ when at least m of the arguments x_i are 1, and $f(x_1, \ldots, x_k) = 0$ otherwise. An SDS is a threshold SDS if every vertex function is a threshold function. It is well-known that limit sets of threshold SDSs contain only fixed points [16], and additionally, that fixed points are independent of update sequence. This class plays a special role in complexity theory since in many cases,

threshold functions are the most general function class for which problems are computationally tractable [1, 4].

Functional linkage networks (FLNs) and the gene annotation algorithm of [9] were among the motivations for considering the type of update sequence stability measures discussed in this section. An FLN (for a fixed gene ontology function f) is a graph where the vertices represent proteins, and an edge is present if two proteins are thought to share the same function, with an edge weight w_{ij} describing certainty. The vertices are assigned states from $K = \{1, -1, 0\}$ denoting whether they are annotated with f (state 1), are not annotated with f (state -1), or if annotation is unknown (state 0). Each vertex i is updated by changing its state to 1 if $\sum w_{ij}x_j \geq T$, where the sum is taken over all neighboring vertices, and T is some fixed threshold, and to -1 otherwise. The algorithm updates the vertices in some fixed order chosen at random, and terminates when a fixed point is reached.

The FLN algorithm uses an asynchronous update schedule to ensure that a fixed point is reached (there may be periodic cycles of length > 1 under a synchronous update schedule). In light of this convenient choice of update scheme, and the random choice of update sequence, one would hope that such a gene annotation algorithm would exhibit update order stability. In other words, the final (fixed point) state reached should be robust with respect to changes in update sequence and perturbations of the initial state, in that the fixed point reach has little or no dependence on the update sequence. Unfortunately, such stability cannot always be guaranteed. As an example, there are general classes of threshold SDSs sharing many of the essential FLN algorithm properties, but which exhibit exponential update sequence instability [11] for many choices of the initial state. Not surprisingly, such instability strongly depends on the choice of underlying graph.

The ω-*limit set* of $\phi \colon K^n \longrightarrow K^n$ from $x \in K^n$ is the set $\omega(\phi; x)$ of periodic states $x \in K^n$ such that $\phi^m(x) = y$ for some $m \geq 0$. This is a specialization of the classical definition of ω-limit set to the case of finite phase spaces. If $\mathcal{P} \subset S_Y$ is a collection of update sequences, then the ω-limit set of $\mathfrak{F}_Y$ from x with respect to $\mathcal{P}$ is defined as

$$\omega_{\mathcal{P}}(x) = \bigcup_{\pi \in \mathcal{P}} \omega([\mathfrak{F}_Y, \pi]; x) .$$

If there are large periodic cycles in the phase space of the SDS, then the size of an ω-limit set may not necessarily be too insightful. However, if there only fixed points, as is common in many applications, then $\omega_{\mathcal{P}}(x)$ is a direct measure of stability that describes how many fixed points can be reached from x by choosing the update schedule from $\mathcal{P}$. For a set of functions $\mathfrak{F}_Y$, define

$$\omega(\mathfrak{F}_Y) = \max |\{\omega_{S_Y}(x) \mid x \in K^n\}| .$$

This counts the maximum possible number of periodic points that can be reached from any state by variation of the update order. As in the previous section, we begin by considering two extreme cases; when the base graph is a tree, and when it is the complete graph K_n. The following result is proven in [11].

Theorem 4.1. Let $\mathfrak{F}_Y$ be a sequence of 2-threshold functions. If Y is the star-graph on n vertices, then $\omega(\mathfrak{F}_Y) = 2^n - n$. If $Y = K_n$, then $\omega(\mathfrak{F}_Y) = n + 1$.

Additionally, this is extended to the random graph model $G(n, p)$ for ranges of p relevant to many applications. The main result is that for sparse graphs, $\omega(\mathfrak{F}_Y) = \Theta(2^n)$ (with high probability), but for dense graphs, $\omega(\mathfrak{F}_Y) = \Theta(n)$. While these systems are deliberately constructed to possess dynamical instability, they nonetheless offer valuable insight into possible dynamics of such systems. With this notion of stability with respect to update sequence, one can make the following general statement, which is in direct contrast to the conclusion of the previous section:

Update order stability and edge density are negatively correlated.

5. PERIODIC POINTS AND WORD INDEPENDENCE

We come at last to our third and final stability concept for asynchronous dynamical systems, which stems from recent work on word independent systems [8, 12, 13]. Like cycle equivalence, this considers invariance properties of the periodic points of SDSs, rather than the transient states. A sequence of Y-local functions $\mathfrak{F}_Y$ is *word independent* if the set of periodic states $\mathsf{Per}([\mathfrak{F}_Y, \omega])$ is the same for all *complete words* $\omega \in W_Y$ (words over $\mathrm{v}[Y]$ where each vertex occurs at least once). Note that word independence only considers the periodic states as a set, and ignores the orbit structure. Word independence was first studied in [8], where it was shown that most classic symmetric functions such as the logical AND, OR, NAND, NOR functions, along with their sums, were word independent, for asynchronous cellular automata (i.e., over the circular graph). Other common functions, such as parity, minority, and majority functions had this property as well. Recently, in a series of two papers [12, 13], we have expanded this study to all 256 elementary cellular automata rules, and classified the word independent ones, which are independent of the size of the underlying (circular) graph.

Theorem 5.1. Precisely 104 of the 256 elementary cellular automata rules induce a word independent SDS over Circ_n, for all $n > 3$.

The classification of these 104 rules is contained in [13], and the structure is further developed in [12] with an analysis of their *dynamics groups*, which describe how the periodic states are permuted within these systems as the local functions are applied. The latter paper additionally contains more general results on dynamics groups. For example, when the state space is $K = \{0, 1\}$, as is typical in finite dynamical systems research, the dynamics group is the homomorphic image of a Coxeter group [5], and for general K, an Artin group.

The papers above are only the beginning of this line of work. There is much to be done to better understand systems that fail to be word independent. Invariance of periodic states under different update sequences can be used as a notion of update sequence stability. This can be further quantified as follows. When a sequence of functions is not word independent, we can define a measure that captures how "close" it is to being word independent, by defining

$$\rho(\mathfrak{F}_Y) = \frac{|\, x \in \mathsf{Per}[\mathfrak{F}_Y, \pi] \text{ for all } \pi \in S_Y|}{|\, x \in \mathsf{Per}[\mathfrak{F}_Y, \pi] \text{ for some } \pi \in S_Y|} \ .$$

Clearly, $0 < \rho(\mathfrak{F}_Y) \le 1$, with $\rho(\mathfrak{F}_Y) = 1$ iff $\mathfrak{F}_Y$ is word independent. This quantity measures how stable the periodic point sets are as a whole, to changes in the update sequence. Under

this definition, word independent sequences of functions are the most stable. In general, it is difficult to determine if a sequence of functions is word independent. Though there may exist a correlation between $\rho(\mathfrak{F}_Y)$ and the topology (e.g., edge density) of Y, there is no good reason to believe that such a link should exist. Rather, there are many examples of functions that are word independent for all graphs, such as threshold functions, monotone functions, functions with only fixed points, parity functions, and the logical NOR functions. Therefore, it is not far-fetched to make the following hypothesis:

There is little, if any, correlation between update order stability and edge density.

6. Concluding Remarks

This article should raise a red flag for anyone anyone attempting to study and characterize stability with respect to update sequence variation. We saw how different notions of stability can lead to different conclusions of the relationship to graph structure. In summary, we looked at three properties of the dynamics and how robust they are to changes in the update sequence. In particular, we considered:

- The variation of structural properties of the phase space as the update sequence changes;
- The variation of reachable limit sets as the update sequence changes;
- The variation of the sets of periodic states as the update sequence changes.

For each of these notions, we defined natural measures quantifying stability, and then inquired about the relationship between these measures and the structure of the underlying graph. Though such a question is quite open-ended, we began by looking at the extreme cases of sparse graphs (e.g., trees) and dense graphs (e.g., complete graphs) to gain some insight, before considering intermediate cases. Up until now, these three notions of stability have only been studied independently, and moreover, outside of the setting of stability. In this article, we see first-hand how differently they behave with respect to a basic property such as edge density of the base graph. Perhaps there are ways to draw connections between them to paint a more complete picture of stability. It is our hope that this article will help lay the groundwork for such future research.

References

1. C. L. Barrett, H. H. Hunt, M. V. Marathe, S. S. Ravi, D. Rosenkrantz, R. Stearns, and P. Tosic, *Gardens of eden and fixed point in sequential dynamical systems*, Discrete Models: Combinatorics, Computation and Geometry AA (DM-CCG), 2001, pp. 95–110.
2. C. L. Barrett, H. S. Mortveit, and C. M. Reidys, *Elements of a theory of simulation III: Equivalence of SDS*, Appl. Math. Comput. **122** (2001), 325–340.
3. C. L. Barrett, H. S. Mortveit, and C. M. Reidys, *Elements of a theory of simulation IV: Fixed points, invertibility and equivalence*, Appl. Math. Comput. **134** (2003), 153–172.
4. C. L. Barrett, H. B. Hunt, M. V. Marathe, S. S. Ravi, D. J. Rosenkrantz, and R. E. Stearns, *Predecessor and permutation existence problems for sequential dynamical systems*, Disc. Math. Theor. Comp. Sci. (DMTCS) 2003, pp. 69–80.

5. A. Björner and F. Brenti, *Combinatorics of Coxeter groups*, Springer-Verlag, New York, 2005.
6. B. Bollobás, *Random graphs*, Cambridge University Press, 2001.
7. D. L. Vertigan F. Jaeger and D. J. A. Welsh, *On the computational complexity of the Jones and Tutte polynomials*, Math. Proc. Cambridge Phil. Soc. **108** (1990), 35–53.
8. A. Å. Hansson, H. S. Mortveit, and C. M. Reidys, *On asynchronous cellular automata*, Adv. Complex Syst. **8** (2005), no. 4, 521–538.
9. U. Karaoz, T. M. Murali, S. Letovsky, Y. Zheng, C. Ding, C. R.Cantor, and S. Kasif, *Whole-genome annotation by using evidence integration in functional-linkage networks*, Proc. Nat. Acad. Sci. **101** (2004), no. 9, 2888–2893.
10. W. O. Kermack and A. G. McKendrick, *A contribution to the mathematical theory of epidemics*, Proc. Royal Soc. London A **115** (1927), 700–721.
11. V. S. A. Kumar, M. Macauley, and H. S. Mortveit, *Limit set reachability in asynchronous graph dynamical systems*, Reachability Problems (RP) 2009 (Berlin/Heidelberg), Lecture Notes in Computer Science, vol. 5797, Springer, 2009, pp. 217–232.
12. M. Macauley, J. McCammond, and H. S. Mortveit, *Dynamics groups of asynchronous cellular automata*, 2008, Submitted. arXiv:0808:1238.
13. M. Macauley, J. McCammond, and H. S. Mortveit, *Order independence in asynchronous cellular automata*, J. Cell. Autom. **3** (2008), no. 1, 37–56.
14. M. Macauley and H. S. Mortveit, *On enumeration of conjugacy classes of Coxeter elements*, Proc. Amer. Math. Soc. **136** (2008), no. 12, 4157–4165.
15. M. Macauley and H. S. Mortveit, *Cycle equivalence of graph dynamical systems*, Nonlinearity **22** (2009), 421–436.
16. H. S. Mortveit and C. M. Reidys, *An introduction to sequential dynamical systems*, Universitext, Springer Verlag, 2007.
17. I. Shmulevich, E. R. Dougherty, and W. Zhang, *From Boolean to probabilistic Boolean networks as models of genetic regulatory networks*, Proc. IEEE **90** (2002), no. 11, 1778–1792.
18. S. Wolfram, *Theory and applications of cellular automata*, Advanced Series on Complex Systems, vol. 1, World Scientific Publishing, 1986.

DEPARTMENT OF MATHEMATICAL SCIENCES, CLEMSON UNIVERSITY, CLEMSON, SC 29634, USA
E-mail address: macaule@clemson.edu

DEPARTMENT OF MATHEMATICS, VIRGINIA TECH, BLACKSBURG, VA 24061, USA
E-mail address: henning@vt.edu