Rule 110 explained as a block substitution system

Juan C. Seck-Tuoh-Mora[1]*, Norberto Hernández-Romero[1]
Genaro J. Martínez[2]

April 2009

[1] Centro de Investigación Avanzada en Ingeniería Industrial, Universidad Autónoma del Estado de Hidalgo Pachuca, Hidalgo, México.
{jseck,nhromero}@uaeh.edu.mx
[2] Bristol Institute Technology, University of the West of England, Bristol, United Kingdom.
genaro.martinez@uwe.ac.uk

Abstract

This paper presents the characterization of Rule 110 as a block substitution system of three symbols; where their production rules, their inverse behaviors and forbidden words are discussed. It is shown that every finite configuration can be partitioned in several block and the dynamics of the automaton can be modeled and reproduced as a mapping among blocks of the same size. The form of the blocks in the current configuration can be used for knowing the number and size of the partitions in the next one, in this way the evolution of random configurations, ether and gliders can be modeled.

1 Introduction

Elementary cellular automata (ECA) are defined by a binary set of states $\Sigma = \{0, 1\}$ and a neighborhood size 3 specifying a mapping $\varphi : \Sigma^3 \to \Sigma$ known as the evolution rule of the automaton. ECA's have been widely studied since their simplicity and easy implementation in computer programs, which allows to perform exhaustive analysis about the properties of these automata given the small number of possible neighborhoods (8) and evolution rules (256) produced by the previous parameters [1]. On the other hand, ECA's exhibit all range of behaviors; from fixed and periodic ones to chaotic and complex dynamics [2].

Different ECA's have been analyzed for their capacity of producing interesting structures which can interact in order to compose more complex reactions; in particular Rule 110 has been proved by Cook to be Turing-complete [3]. Rule 110 is an ECA characterized by forming from random initial conditions, interesting evolutions conforming a periodic background called ether on which a finite set of periodic structures known as gliders may collide showing different results (annihilations, solitons and productions of gliders by collisions) [6]. A typical evolution from a random initial condition is depicted in Fig. 1; ether is displayed with different color to clarify the gliders emerging from the initial chaotic reactions.

*This work was partially supported by CONACYT project CB-2007/083554

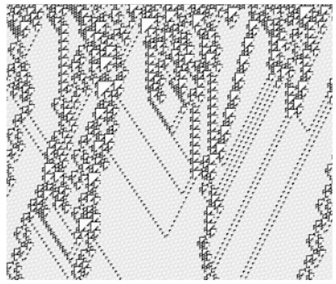


Figure 1: Typical evolution of Rule 110 from a random configuration.

Rule 110 and other rules are widely analyzed by Wolfram in [8]; one aspect investigated in his work is the application of different discrete and emergent systems for simulating ECA's.

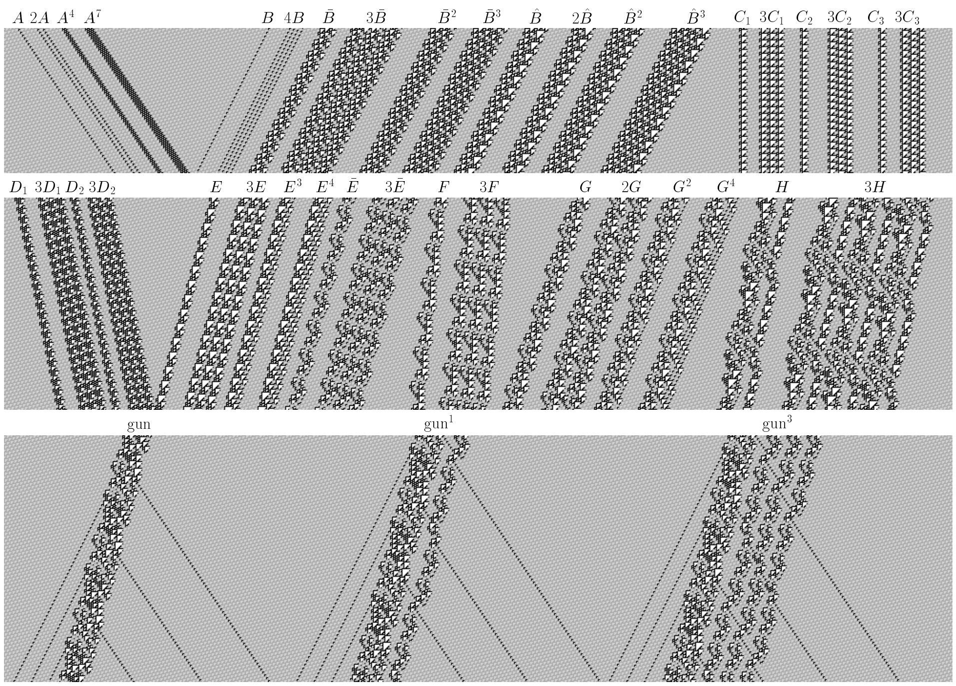


Figure 2: System of gliders in Rule 110.

Dynamics in Rule 110 is concentrated mainly on a wide variety of kinds of gliders, extensions and combinations of them. We can handle packages of them in one or several phases; thus notation nA means n copies of the A glider and not a package of A^n gliders. Fig. 2 lists all known gliders so far both in their basic representation and in packages or extensions as well.

These gliders have important characteristic useful to define distances, slopes, speeds, periods, collisions, and phases. Of course, a huge number of reactions were constructed to get complex constructions, as the cyclic tag system.

Representation of gliders have been obtained from regular expressions [7], de Bruijn diagrams [4], or by tiles assembly [5]. In fact, some techniques grown exponentially and therefore there is not an automatic procedure to construct such patterns.

In this sense, the original part of this paper is to show that there exists a substitution system with three symbols which is able to simulate the behavior of Rule 110; this substitution system has special properties useful for characterizing and understanding the periodical behavior of gliders and the conservation of information in soliton-like collisions.

2 Basic concepts

The one-dimensional cellular automaton Rule 110 is an ECA whose evolution rule φ is defined in Table 1; the rule is enumerated as 110 taking the mappings defined by φ as a binary number.

Neighborhood	111	110	101	100	011	010	001	000
Evolution	0	1	1	0	1	1	1	0

Table 1: Evolution rule defining the ECA Rule 110

As Fig. 1 shows the evolution of Rule 110 is characterized by forming mobile structures (gliders) on a periodic background (ether); this gliders can interact in such a way that a cyclic tag system can be implemented (using millions of cells) in order to prove the universality of this automaton [3]. In this way, Rule 110 is relevant for showing how simple systems based on local interactions with no central control, are able to produce a complex global behavior. One notorious feature is that the evolution rule covers the space with triangles outlined by the cells of the automaton, internally defined by 0's with margins delineated by 1's; every triangle can be identified by the length n of its largest sequence of 0's. Some examples of triangles produced by Rule 110 are depicted in Fig. 3 (as was defined in [4]).

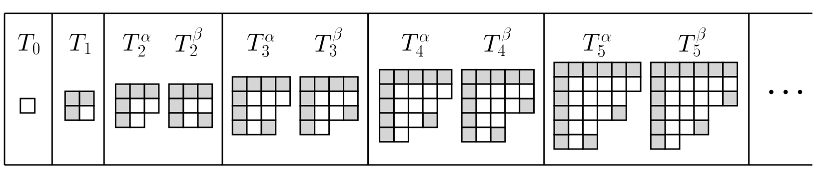


Figure 3: Family of triangles as tiles exemplifying the constructions produced by Rule 110.

The triangles in Fig. 3 show that there are two sequences defining their borders; $B_1 = 1$ (if and only if it is followed by a state 0) and $B_2 = 01$ and, every sequence of n states between any pair of borders can be considered as a string S_n; note that S_n may be equal to 0^n or 1^n. In this way, every sequence in $w \in \Sigma^2$ can be substituted by another one formed by the previously described elements; and example of such substitution is presented in Fig. 4.

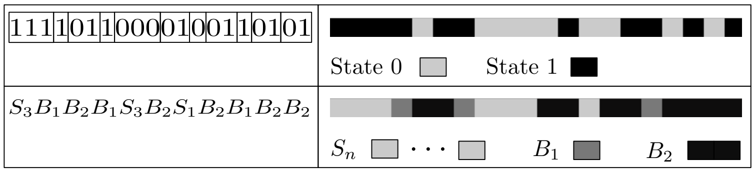


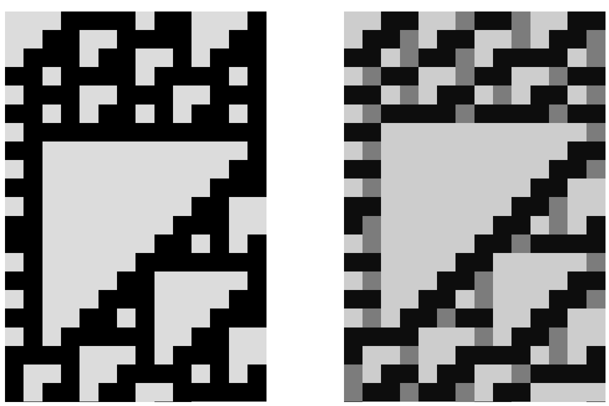
Figure 4: Sequence 11110110000100110101 and its substitution using S_n, B_1 and B_2.

3 Substitution system modeling Rule 110

The elements S_n, B_1 and B_2 can be used for implementing a substitution system in order to simulate the dynamics of Rule 110; the production rules associated to these elements are the following ones:

1. $S_nB_1 \rightarrow S_{n-1}B_2$
2. $S_nB_2 \rightarrow S_{n-1}B_2B_1$
3. $S_nB_{\alpha_1}B_{\alpha_2}\dots B_{\alpha_m} \rightarrow S_{n-1}B_2S_pB_1$ where $p = \sum_{i=1}^{m} \alpha_i - 2$

The rules are obtained following the behavior to form triangles specified by Rule 110; each string S_n is decremented in one element after one step of the substitution system ($S_n \rightarrow S_{n-1}$); meanwhile borders B_1 and B_2 are used to close the margins of the triangles. In the production rules we can see as well that a sequence of borders generates a new string; continuing the formation of triangles in the evolution space. Figure 5 shows the normal evolution of Rule 110 from a random configuration and the evolution of the substitution system codifying the initial condition with S_n, B_1 and B_2 and using the production rules applying periodic boundary conditions in both cases.



a) Original b) Substitution system

Figure 5: Evolution of Rule 110 from a random initial condition taking both the original evolution rule and the production rules of the substitution system.

Figure 5 exposes that we have similar evolutions taking both the evolution rule or the production rules, in fact the second case is one step forward in relation to the first one on forming the different triangles in the evolution space. This feature can be explained taking into account that the substitution of the initial condition is similar to perform a one-step evolution of the automaton; thereby the dynamics of the substitution system is one step forward that the one developed by the original evolution rule.

In the definition of every element in the substitution system, the associated subscript indicates its length; in this way we can see that the production rules establish mappings between sequences of elements of equal lengths; this indicates that the evolution of Rule 110 can be characterized by mappings between blocks with the same number of states.

In the production rules; it is clear that a B_1 is always produced preceded by a B_2 element (with or without a S_n in the middle). This indicates that the sequence $B_1 S_n B_1$ cannot be produced by the evolution of the substitution system for each $n \in \mathbb{N}$; this is the clearest forbidden word in the substitution system. Now let us take the sequence $S_m B_2 S_n B_2 S_p B_2$, following the production rules we have that the prefix $S_m B_2 S_n B_2$ can be only produced by $S_{m+1} B_1 S_{n+1} B_1$ which is not possible. Therefore another forbidden word is $B_2 S_n B_2 S_p B_2$.

The previous forbidden words indicate that the substitution system is not reversible, because there are sequences which cannot be yielded by the production rules; let us analyze in detail the non-reversible part of the system. Since the production rules are established between sequences with equal lengths, a naive approximation for reversing them can be achieved taking the inverse direction as follows:

1. $S_n B_2 \rightarrow S_{n+1} B_1$
2. $S_n B_2 B_1 \rightarrow S_{n+1} B_2$
3. $S_n B_2 S_p B_1 \rightarrow S_{n+1} B_{\alpha_1} B_{\alpha_2} \ldots B_{\alpha_m}$ where $\sum_{i=1}^{m} \alpha_i = p + 2$

The first two production rules are reversible because in both cases the mapping is bijective, however the third rule explains the non-reversibility of the system due to the sequence $B_{\alpha_1} B_{\alpha_2} \ldots B_{\alpha_m}$ is not uniquely determined, that is, it can be specified in several ways; taking into account that the forbidden words defined above cannot appear as part of it.

4 Block mapping modeling the behavior of Rule 110

The production rules defining the substitution systems are based in sequences such as $S_n B_{\alpha_1} B_{\alpha_2} \ldots B_{\alpha_m}$ for $n \in \mathbb{N}$ and $\alpha_i \in \{1, 2\}$. In this way every configuration $w \in \Sigma^*$ can be partitioned in blocks with the previous specification, such that every block maps into another one with the same length using the production rules. For instance, let us take the configuration 0001110111100011, applying the evolution rule with periodic boundary conditions we obtain the configuration 0011011100100111. The same process can be obtained in a equivalent way codifying the first configuration with the three elements of the substitution system, producing the initial sequence $S_2 B_2 S_1 B_1 B_2 S_2 B_1 S_2 B_2 B_1$; using the production rules we get the sequence $S_1 B_2 B_1 B_2 S_1 B_1 S_1 B_2 S_1 B_2 S_1 B_1$, the block partitions and mappings representing this process are depicted in Fig. 6, both with symbols and in a graphical way for visualizing the mappings between blocks of identical size.

In order to continue with the evolution of the system, the blocks of the last sequence in Fig. 6 must be reordered to be able of using again the production rules; thus the sequence

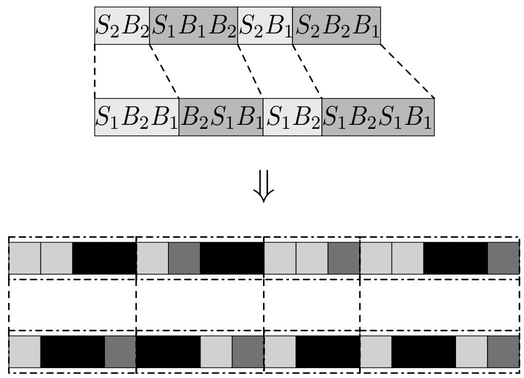


Figure 6: The evolution of Rule 110 can be represented by a mapping between blocks of same size.

$S_1B_2B_1B_2S_1B_1S_1B_2S_1B_2S_1B_1$ yields the new one $B_2S_3B_1B_2B_2B_1B_2B_1B_2$, which describes the configuration 0111110101101101 in the classical evolution of the automaton.

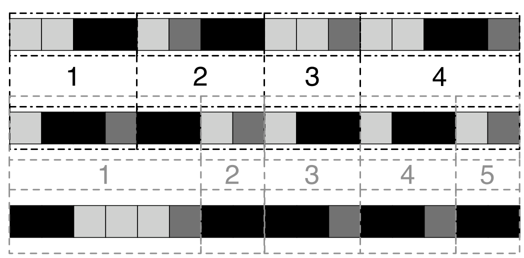


Figure 7: Reordering of the blocks partitioning the second sequence for yielding the next evolution of the system, passing from 4 blocks in the first mapping to 5 blocks in the second one.

Therefore the evolution of Rule 110 can be modeled by block mappings which are reordered every step according to the formation and concatenation of symbols B_α in each new sequence; in this way the number of partitions defining the mappings of the substitution system may change as its dynamics advances in time. For instance, in Fig. 7 the first sequence is partitioned in 4 blocks (described by dotted black lines); after the mapping, the second one is partitioned in 5 blocks (depicted by dotted gray lines) to achieve the next evolution of the system. Note that the last sequence delineates a single block if it is desired to continue with the evolution of the automaton.

The size and number of the blocks in the next sequence can be determined analyzing the composition of the partitions in the current one, following the subsequent conditions:

1. The number of blocks is decreased in one element in the next sequence for every block S_1B_α in the current one.
2. The number of blocks is increased in one element in the next sequence for every block $B_{\alpha_1}\dots B_{\alpha_m}$ in the current one when $\sum \alpha_i > 2$.

3. For $\gamma \in \{1,2\}$ and $n > 1$, if there is a block $S_n B_\gamma S_{\beta_1} B_{\alpha_1} S_{\beta_2} B_{\alpha_2} \dots S_{\beta_m} B_{\nu_1 \dots \nu_p}$ such that every $\beta_i = 1$ and each $\alpha_j \in \{1,2\}$, hence the block $S_n B_\gamma$ grows as $S_{n-1} B_{\omega_1} \dots B_{\omega_q}$ in the next sequence such that $\sum \omega_k = (\gamma + 1) + \sum(\alpha_j + 1) + 2$.

4. If there is a sequence $S_1 B_{\alpha_1} B_{\alpha_2} \dots B_{\alpha_m}$ such that every $\alpha_i \in \{1,2\}$, hence this block is decremented to $S_n B_1$ such that $n = \sum \alpha_i - 2$.

Conditions 3 indicates that the size of a block is incremented in at least two units when it has one or more S_1 sequences on the left and, condition 4 explains that the size of a block is decremented in two units when it is composed by several B_α symbols. This gaining or lost of size provides the reordering of blocks; sometimes this behavior works as a shift of the initial blocks in configurations with periodic boundary conditions, similar to the process characterizing the dynamics of reversible one-dimensional cellular automata based on blocks permutations and shifts [9].

5 Examples of the block mapping induced by the substitution system

This section presents three cases of the block mapping modeling the dynamics of Rule 110, the first one taking a random configuration, then using the configuration representing ether and finally employing the configuration corresponding to the E glider. In these examples, every block in a partition will have associated an integer $a \in \{-1,0,1\}$ indicating if the block decrements, does not change or increments the number of blocks in the next sequence according to conditions 1 and 2. Let us notice that in the second case, a block does not affect the number of blocks in the following sequence due to it does not hold conditions 1 and 2 or it fulfills both conditions at the same time.

5.1 Block mapping for a random configuration

Let us take the configuration 11111101001010001101, this one can be represented in the substitution system by $S_5 B_1 B_2 S_1 B_2 B_2 S_2 B_2 B_1 B_2$. Ten evolutions of the block mapping system are presented in Fig. 8, in this example there is associated an integer below each block indicating how this one modifies the number of blocks partitioning the next sequence.

5.2 Block mapping for ether configuration

Let us take now the configuration 11111000100110 which evolves in the periodic background called "ether" in Rule 110; with the substitution system this configuration is encoding as $S_3 B_1 S_2 B_2 S_1 B_2 B_1 B_2$. The block mapping associated to this sequence for five evolution steps is depicted in Fig. 9.

Figure 9 shows that the block mapping acts as a shift in four positions to the left over the initial sequence; this implies that a sequence is periodic in one step if for each of its blocks, there is one or more contiguous blocks constructing it in the same sequence by the action of the production rules; explaining the way in which the conservation of information works in the substitution system in one iteration.

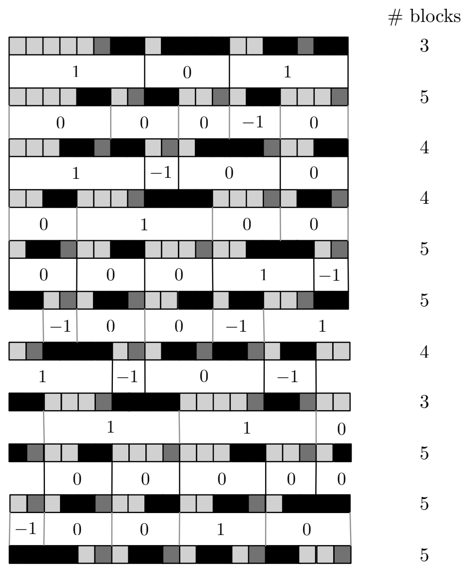


Figure 8: Ten evolutions of the sequence 11111101001010001101 given by the substitution system, the number, specification and reordering of blocks are described in every step.

5.3 Block mapping characterizing the E glider

Finally, let us take the configuration 0011011111111110001 which composes a periodic structure (known as E glider); this configuration is codified in the substitution system as $S_1B_2B_1B_2S_8B_1S_2B_2$. Figure 10 displays fifteen steps of the substitution system; showing how the blocks are reordered to obtain a periodic behavior.

In this case, the block mapping acts as a shift of four places to the left after fifteen steps, also it is clear that the number of blocks is not conserved in the evolution of the system, but at the end, it is able to recovery the original blocks.

6 Final discussion

This paper has demonstrated that the dynamics of Rule 110 can be modeled as a mapping between blocks of equal sizes, in this way the complexity in its behavior is given by the reordering of blocks in very step when symbols B_α from different mappings are concatenated through the evolution of the system. The number and size of the blocks depend on the symbols S_1 and the sequences $B_{\alpha_1}B_{\alpha_2}\dots B_{\alpha_m}$; in particular analyzing the production rules, we can see that each symbol B_2 runs from right to left in the evolution space generating a B_1 every two steps and, each B_2 stops up to reach other B_1 or B_2 symbol.

In some cases the substitution system conserves the information of the original sequence, although the number of blocks may vary in this process, thus we can explain the complex dynamics of Rule 110 by a combination of block mappings and reorderings; meanwhile the block mappings provides order to the system, their reorderings induce the adequate degree of disorder to

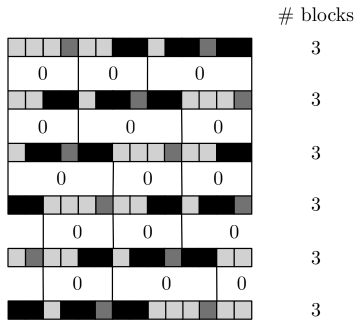


Figure 9: Five evolutions of the ether sequence using the substitution system, in this case the behavior of the blocks is given by a left shift of four positions.

produce the complex behavior.

Further research include the analysis of the production rules as a dynamical system, in the sense of formalizing its properties about periodic and transitive points and orbits, repetitivity, ergodic and mixing properties; and how these features are able to obtain a deeper understanding and classification of the dynamics of Rule 110. On the other hand, another subsequent work may be the study of glider interactions and soliton-like collisions from the perspective of the substitution system, in order to typify the lost, regeneration and preservation of information in these kind of situations.

References

[1] Wolfram, S. (1986) *Theory and Aplications of Cellular Automata*, World Scientific Press, Singapore.

[2] Wuensche, A. (1994) Complexity in one-d cellular automata, *Santa Fe Institute* working paper 94-04-025.

[3] Cook, M. (2004) Universality in Elementary Cellular Automata, *Complex Systems* **15 (1)** 1–40.

[4] McIntosh, H.V. (1999) Rule 110 as it relates to the presence of gliders, `http://delta.cs.cinvestav.mx/~mcintosh/oldweb/pautomata.html`

[5] Martínez, G.J. & McIntosh, H.V. (2001) ATLAS: Collisions of gliders like phases of ether in Rule 110, `http://delta.cs.cinvestav.mx/~mcintosh/comun/s2001/s2001.html`

[6] Martínez, G.J., McIntosh, H.V., & Seck-Tuoh-Mora, J.C. (2006) Gliders in Rule 110, *Int. J. of Unconventional Computing* **2 (1)** 1–49.

[7] Martínez, G.J., McIntosh, H.V., Seck-Tuoh-Mora, J.C., & Chapa-Vergara, S.V. (2008) Determining a regular language by glider-based structures called *phases* $f_{i}\text{-}1$ in Rule 110, *Journal of Cellular Automata* **3 (3)** 231–270.

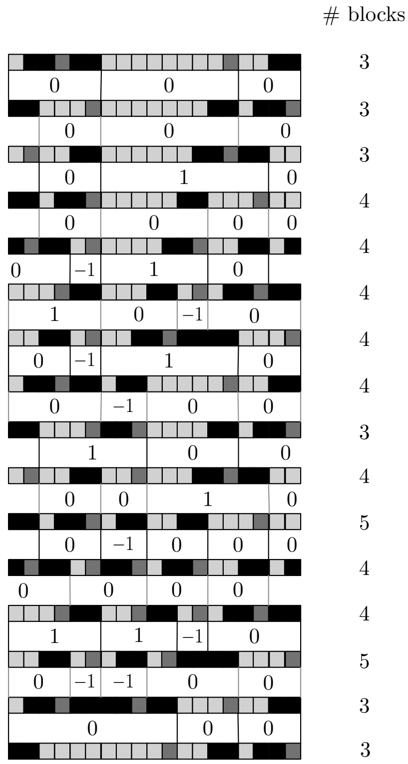


Figure 10: Fifteen evolutions of the substitution system on the sequence codifying the E glider.

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

[9] Kari, J. (1996) Representation of Reversible Cellular Automata with Block Permutations, *Mathematical Systems Theory* **29** 47–61.