One-Dimensional Cellular Automaton Transitions and Integral Value Transformations Representing Deoxyribonucleic Acid Sequence Evolutions

Sreeya Ghosh
Electronics and Communications Sciences Unit, Indian Statistical Institute
Kolkata-700108, India
sreeya135@gmail.com

Sudhakar Sahoo
Institute of Mathematics and Applications
Bhubaneshwar-751029, India
sudhakar.sahoo@gmail.com

Sk. Sarif Hassan
Department of Mathematics, Pingla Thana Mahavidyalaya
Paschim Medinipur-721140, India
sksarifhassan@pinglacollege.ac.in

Jayanta Kumar Das
School of Medicine, Johns Hopkins University
MD-21287, USA
Applied Statistics Unit, Indian Statistical Institute
Kolkata-700108, India
dasjayantakumar89@gmail.com

Pabitra Pal Choudhury
Applied Statistics Unit, Indian Statistical Institute
Kolkata-700108, India
pabitrapalchoudhury@gmail.com

Antara Sengupta
Department of Computer Science and Engineering, University of Calcutta
Kolkata-700009, India
antara.sngpt@gmail.com

The cellular automaton (CA) and an integral value transformation (IVT) evolving in discrete time steps are two mathematical models that are well established. Theoretically, it can be suggested that a CA possesses the capacity to produce varieties of evolutionary patterns. However, computing a CA in higher dimensions or computing a nonlinear CA may be complex. In such cases, an IVT can be conveniently used. This paper presents the relation between the transition functions of a one-dimensional CA and an IVT. It also highlights the algebraic structures on the basis of binary operations for a set of transition functions of a one-dimensional CA and for a set of IVTs. The suitability of using

https://doi.org/10.25088/ComplexSystems.32.2.115
an IVT over a CA is discussed. Also, we present the evolutionary models of two deoxyribonucleic acid (DNA) sequences through IVTs and their spacetime diagrams. This can eventually bring out some characteristic features of the evolutionary sequences.

Keywords: CA; IVT; mutation and crossover operations; DNA sequence evolution

1. Introduction

A cellular automaton (CA) was introduced by J. von Neumann and S. Ulam in the 1940s for designing self-replicating discrete systems. It has applications in the domains of mathematics, computer science, complexity science, physics, biology and microstructure modeling. In a CA, each cell changes according to some transition function on receiving input from the neighboring cells [1]. Change of states of each cell at discrete time steps induces a change of the entire grid pattern. In the 1980s, Stephen Wolfram provided a classification of one-dimensional cellular automata (CAs) with respect to their evolution patterns [2]. Many interesting problems that utilize mathematical bases such as polynomial, matrix algebra and Boolean derivative [3, 4] may be solved through CA analysis. Further, the study of complex dynamic behaviors of CAs using various mathematical tools [5, 6] can be an aid for understanding and modeling various classes of complex discrete dynamical systems [7, 8].

An integral value transformation (IVT), first introduced during 2009–2010 [9–11], is a kind of discrete dynamical system. IVTs have been studied from various viewpoints, which include the understanding of integer sequence evolutions, behavioral patterns of integers at discrete points of time and number-preserving rules of CAs [10, 12]. For an IVT, updating the cells does not depend on overlapping neighborhood interaction or any boundary conditions [12], which is unlike the updating behavior of a CA. Hence, a dynamical system having more than two states can be more easily and conveniently modeled by an IVT rather than by a CA.

The relationship between the two above-mentioned one-dimensional discrete dynamical systems, CAs and IVTs, was presented in [13]. The Wolfram code for an elementary CA (ECA) and global transition functions of a one-dimensional CA could be represented through IVTs. The current paper is an extension of [13] where we include modeling through IVTs for representing evolution of biomolecular systems.

Quantitative understanding of biological systems refers to the application of quantitative techniques for analyzing experimental data and modeling biological systems. Although applications of CAs in the
biomolecular domain is a well-known concept [14–16], very little work has been performed in the light of IVT [17].

It is known that a deoxyribonucleic acid (DNA) sequence is composed of a linear string of nucleotides, or bases, which are of four types, namely adenine (A), cytosine (C), thymine (T) and guanine (G). The frequency of occurrence of these four bases in a DNA sequence is often conserved from species to species. Hence, their physico-chemical properties and mutation patterns are useful in characterizing a disease pathogenesis. Bases A and G constitute purines (R), consisting of two hydrogen-carbon rings and four nitrogen atoms. Bases C and T constitute pyrimidines (Y), consisting of one hydrogen-carbon ring and two nitrogen atoms. A point mutation that substitutes a purine to another purine nucleotide (A ↔ G) or a pyrimidine to another pyrimidine nucleotide (C ↔ T) is called a transition, while transversion refers to a point mutation substituting a single purine (A or G) by a pyrimidine (T or C), or vice versa [18, 19]. Measuring the number of occurrences of purines and pyrimidines [20, 21] or the alterations of nucleotides during mutations [19, 22] in DNA sequences plays a vital role in their characterization.

This paper presents studies on the algebraic structures of a set of transition functions of a one-dimensional CA as well as those of a set of IVTs following some binary operations in Sections 4 and 5. In Section 6, DNA sequences are visualized in terms of a four-base number sequence. Hence, evolutions of two different DNA sequences have been modeled using IVTs. The spacetime diagram of the evolution patterns with respect to three different IVT rules is shown, where one rule reflects transversion and two rules reflect transitions of nucleotides. Also, the numbers of purines and pyrimidines at each iterative step are computed.

## 2. Basic Mathematical Concepts

**Definition 1.** Let \( Q \) be a finite set of memory elements also called the state set. A global configuration is a mapping from the group of integers \( \mathbb{Z} \) to the set \( Q \) given by \( C : \mathbb{Z} \to Q \). The set \( Q^\mathbb{Z} \) is the set of all global configurations where \( Q^\mathbb{Z} = \{ C \mid C : \mathbb{Z} \to Q \} \).

**Definition 2.** A mapping \( \tau : Q^\mathbb{Z} \to Q^\mathbb{Z} \) is called a global transition function. A CA (denoted by \( C^\mathbb{Z}_\tau \)) (reported in [23–25]) is a triplet \( (Q, Q^\mathbb{Z}, \tau) \) where \( Q \) is the finite state set, \( Q^\mathbb{Z} \) is the set of all configurations and \( \tau \) is the global transition function.
Definition 3. The set \( Q^Z = \{ \tau \mid \tau : Q^Z \to Q^Z \} \) is the set of all possible global transition functions of a CA having state set \( Q \).

A mapping \( \tau \) is invertible if \( \forall C_i, C_j \in Q^Z, \exists \tau^{-1} \) such that \( \tau(C_i) = C_j \iff \tau^{-1}(C_j) = C_i \).

Definition 4. For \( i \in \mathbb{Z}, r \in \mathbb{N} \), let

\[ S_i = \{ i - r, \ldots, i - 1, i, i + 1, \ldots, i + r \} \subseteq \mathbb{Z} \]

\( S_i \) is the neighborhood of the \( i^{th} \) cell; \( r \) is the radius of the neighborhood of a cell. It follows that \( \mathbb{Z} = \bigcup_i S_i \). A restriction from \( \mathbb{Z} \) to \( S_i \) induces a restriction of \( C \) to \( \bar{c}_i \) given by \( \bar{c}_i : S_i \to Q \), where \( \bar{c}_i \) may be called a local configuration of the \( i^{th} \) cell.

The mapping \( \mu_i : Q^{S_i} \to Q \) is known as a local transition function for the \( i^{th} \) cell having radius \( r \). Thus \( \forall i \in \mathbb{Z}, \mu_i(\bar{c}_i) \in Q \), and it follows that

\[ \tau(C) = \tau(\ldots, c_{i-1}, c_i, c_{i+1}, \ldots) = \ldots \mu_{i-1}(\bar{c}_{i-1}) \cdot \mu_i(\bar{c}_i) \cdot \mu_{i+1}(\bar{c}_{i+1}) \ldots \]

Definition 5. The set \( M = \{ \mu_i \mid \mu_i : Q^{S_i} \to Q, i \in \mathbb{Z} \} \) is the set of all possible local transition functions of a CA having state set \( Q \).

Definition 6. Wolfram code, introduced by Stephen Wolfram [2], is a naming system often used for an ECA (a binary CA having radius one). For a one-dimensional CA with \( Q \) states, the local rule \( \mu_i \) for some \( i^{th} \) cell, \( i \in \mathbb{Z} \) of radius \( r \) (neighborhood \( 2r + 1 \)) can be specified by a \( Q^{2r+1} \)-bit sequence. The decimal equivalent form of this sequence is known as the Wolfram code.

Thus, the Wolfram code for a particular rule is a number in the range from 0 to \( Q^{Q^{2r+1}} - 1 \), converted from \( Q \)-ary to decimal notation.

Definition 7. A \( p \)-adic \( k \)-dimensional integral value transformation (IVT) denoted by \( \text{IVT}_I^{p,k} \) for \( p \in \mathbb{N}, k \in \mathbb{N} \) is a function of \( p \)-base numbers from \( \mathbb{N}_0^k \) to \( \mathbb{N}_0 \) defined (in [13]) as

\[
\text{IVT}_I^{p,k}(n_1, \ldots, n_k) = \left(f_j(a_0^{n_1}, \ldots, a_0^{n_k})f_j(a_1^{n_1}, \ldots, a_1^{n_k})\ldots f_j(a_{l-1}^{n_1}, \ldots, a_{l-1}^{n_k})\right)_p = m,
\]

where \( \mathbb{N}_0 = \mathbb{N} \cup \{0\}, p \in \mathbb{N} \), \( n_s = (a_0^{n_s}a_1^{n_s}\ldots a_{l-1}^{n_s})_p \) for \( s = 1, 2, \ldots, k \).

\( f_j \) is a function from \( \{0, 1, \ldots, p - 1\}^k \) to \( \{0, 1, \ldots, p - 1\} \) for \( j = 0, 1, \ldots, p^p - 1 \); \( m \) is the decimal conversion of the \( p \)-base number.
Example 1. For $k = 1$, the computation of $\text{IVT}_{j}^{p,1}(n_1) = n_2$ can be visualized as follows:
\[
\begin{align*}
n_1 &= a_0^{n_1}, a_1^{n_1}, \ldots, a_{l-1}^{n_1} \\
n_2 &= a_0^{n_2}, a_1^{n_2}, \ldots, a_{l-1}^{n_2}.
\end{align*}
\]

Here, the $i^{th}$ cell $a_i^{n_2}$ is calculated using the local function $f_j(a_i^{n_1})$ applied to the cells $a_i^{n_1}$. Like a CA, starting from the initial sequence for integer $n_1$, the spacetime diagrams of different IVTs can be calculated by recursively applying an IVT to integers $n_{t-1}$ up to certain time steps $t$ as shown later in Figures 2 and 3.

Example 2. For $k = 2$, the computation of $\text{IVT}_{j}^{p,2}(n_1, n_2) = n_3$ can be visualized as follows
\[
\begin{align*}
n_1 &= a_0^{n_1}, a_1^{n_1}, \ldots, a_{l-1}^{n_1} \\
n_2 &= a_0^{n_2}, a_1^{n_2}, \ldots, a_{l-1}^{n_2} \\
n_3 &= a_0^{n_3}, a_1^{n_3}, \ldots, a_{l-1}^{n_3}.
\end{align*}
\]

Here, the $i^{th}$ cell $a_i^{n_3}$ is vertically dependent on the cells $a_i^{n_1}$ and $a_i^{n_2}$, and it is calculated using the local function $f_j(a_i^{n_1}, a_i^{n_2})$ [26, 27].

Example 3. In [28, 29], the dynamics of pairwise IVTs of the form $(\text{IVT}_{j_1}^{p,2}, \text{IVT}_{j_2}^{p,2})$ over two input sequences for integers $(n_1, n_2)$ has been studied, which generates the following pairs of integers $(n_3, n_4)$:
\[
\begin{align*}
n_1 &= a_0^{n_1}, a_1^{n_1}, \ldots, a_{l-1}^{n_1} \\
n_2 &= a_0^{n_2}, a_1^{n_2}, \ldots, a_{l-1}^{n_2} \\
(\text{IVT}_{j_1}^{p,2}, \text{IVT}_{j_2}^{p,2})(n_1, n_2) &= (\text{IVT}_{j_1}^{p,2}(n_1, n_2), \text{IVT}_{j_2}^{p,2}(n_1, n_2)) \\
&= (n_3, n_4)
\end{align*}
\]

where
\[
\begin{align*}
n_3 &= a_0^{n_3}, a_1^{n_3}, \ldots, a_{l-1}^{n_3} \\
n_4 &= a_0^{n_4}, a_1^{n_4}, \ldots, a_{l-1}^{n_4}.
\end{align*}
\]

In Section 6, the IVTs like Examples 1 and 3 are studied for $p = 4$. More specifically, $\text{IVT}_{j}^{4,1}$ and $\text{IVT}_{j}^{4,2}$ $\forall j = 0, 1, \ldots, 256$ are used for mutation and crossover operations, respectively.

Remark 1. In particular [9, 26, 27], if $\forall i = 0, 1, \ldots, l-1$, the function $f_j$ is defined as

https://doi.org/10.25088/ComplexSystems.32.2.115
1. \( f_j(a_{i_1}^{n_1}, \ldots, a_{i_k}^{n_k}) = \left\lfloor \frac{(a_{i_1}^{n_1} + \cdots + a_{i_k}^{n_k})}{p} \right\rfloor \), then \( \text{IVT}^{p,k}_j \) is known as a modified carry value transformation (MCVT). Again, MCVT with a 0 padding at the right end is known as a carry value transformation (CVT).

2. \( f_j(a_{i_1}^{n_1}, \ldots, a_{i_k}^{n_k}) = (a_{i_1}^{n_1} + \cdots + a_{i_k}^{n_k}) \mod p \), then \( \text{IVT}^{p,k}_j \) is known as an exclusive OR transformation (XORT).

3. \( f_j(a_{i_1}^{n_1}, \ldots, a_{i_k}^{n_k}) = \max(a_{i_1}^{n_1}, \ldots, a_{i_k}^{n_k}) \), then \( \text{IVT}^{p,k}_j \) is known as an extreme value transformation (EVT).

### 3. Wolfram Code of Cellular Automata and Integral Value Transformations

For an ECA, Wolfram code for a local rule \( \mu \) is the decimal number \( j \in \{0, 1, 2, \ldots, 255\} \) obtained from the eight-bit sequence

\[
\mu(111)\mu(110)\mu(101)\mu(100)\mu(011)\mu(010)\mu(001)\mu(000) = j.
\]

Therefore, Wolfram code for each local transition function of a three-neighborhood Boolean CA can be represented by a two-base, three-dimensional IVT [13]. The function \( \mu \) in the above eight-bit sequence can be represented by a function \( f_j : \{0, 1\}^3 \to \{0, 1\} \) that gives

\[
\text{IVT}^{2,3}_j = (f_j(111)f_j(110)f_j(101)f_j(100)f_j(011)f_j(010)f_j(001)f_j(000))_2
\]

\[
\text{IVT}^{2,3}_j(240_{10}, 204_{10}, 170_{10}) = (11110000_2, 110011002, 101010102).
\]

Hence it follows that for a three-neighborhood Boolean CA, any Wolfram code \( j \in \{0, 1, 2, \ldots, 255\} \) can be represented by

\[
\text{IVT}^{2,3}_j(240_{10}, 204_{10}, 170_{10})
\]
equivalently.

Here, we state some examples to depict representation of Wolfram codes in terms of IVTs. In [13], representation of some particular Wolfram codes through IVTs has also been discussed.

**Example 4.** Wolfram code 200 can be equivalently represented as

\[
200_{10} = (11001000)_2 = \text{IVT}^{2,3}_{200}(240_{10}, 204_{10}, 170_{10}).
\]

However, the following example shows that for \( j \in \{0, 1, 2, \ldots, 255\} \), any \( \text{IVT}^{2,3}_j \) may not correspond to a Wolfram code.
Example 5.

$$\text{IVT}_j^{2,3}(240_{10}, 204_{10}, 171_{10}) =$$
$$\text{IVT}_j^{2,3}(11 110 000_2, 11 001 100_2, 10 101 011_2) =$$
$$\left(f_j(111)f_j(110)f_j(101)f_j(100)f_j(011)f_j(010)f_j(001)f_j(001)\right)_2.$$

In this eight-bit sequence, $f_j(000)$ is missing and so this cannot correspond to any Wolfram code.

For a three-state, three-neighborhood CA, the local rule can be represented by a function $f_j : \{0, 1, 2\}^3 \rightarrow \{0, 1, 2\}$.

Example 6. CA transition rule 200 can be equivalently represented in a three-base, three-neighborhood system as

$$200_{10} = (21 102)_3 =$$
$$\text{IVT}_j^{3,3}(222 222 222 111 111 111 111 000 000 000_3, 222 111 000 222 111 000 222 111 000_3, 210 210 210 210 210 210 210 210 210 210_3) =$$
$$\text{IVT}_j^{3,3}(7 625 403 764 90110, 7 479 532 539 76510, 6 159 136 430 18110)_3.$$

For a two-state, five-neighborhood CA, the local rule can be represented by a function $f_j : \{0, 1\}^5 \rightarrow \{0, 1\}$.

Example 7. CA transition rule 69800 can be equivalently represented in a two-base, five-neighborhood system as

$$69 800_{10} = (10 001 000 010 101 000) = \text{IVT}_j^{2,5}_{69 800}$$

$$\left(11 111 111 111 111 110 000 000 000 000_2, 11 111 111 000 000 001 111 111 110 000 000_2, 11 110 000 111 100 001 111 000 011 110 000_2, 11 001 100 110 011 001 100 110 011 001 100_2, 10 101 010 101 010 101 010 101 010 101 010_2\right) =$$
$$\text{IVT}_j^{2,5}_{69 800}(4 294 901 76010, 4 278 255 36010, 4 042 322 16010, 3 435 973 83610, 2 863 311 53010).$$

4. Transition Function of a Cellular Automaton and Integral Value Transformation

Definition 8. The set $T_{p,k} = \left\{\text{IVT}_j^{p,k} \mid \text{IVT}_j^{p,k} : \mathbb{N}_0^k \rightarrow \mathbb{N}_0\right\}$ is the set of all $p$-base, $k$-dimensional IVTs for $p \in \mathbb{N}$, $k \in \mathbb{N}$.
Definition 9. The \( \nu \)-fold Cartesian product \( T^{p,k}_\nu \times \cdots \times T^{p,k}_\nu \) for \( \nu \in \mathbb{N} \), denoted by \( T^\nu \) is given by \( T^\nu = \{ (IVT^{p,k}_{j_1}, IVT^{p,k}_{j_2}, \ldots, IVT^{p,k}_{j_\nu}) \} \), where \( IVT^{p,k}_{j_i} : \mathbb{N}_0^k \to \mathbb{N}_0 \in T^{p,k} \), for \( s = 1, 2, \ldots, \nu( \in \mathbb{N}) \).

Definition 10. Let a restriction on \( IVT^{p,k}_{j_i} \) from \( \mathbb{N}_0^k \) to \( Q^k = \{ 0, 1, \ldots, p-1 \} \) for \( p \in \mathbb{N} \) be denoted by \( IVT^{p,k}_{j_i} \). The set \( T^{p,k}_|Q| = \{ IVT^{p,k}_{j_i} \mid IVT^{p,k}_{j_i} : Q^k \to Q \} \) is the set of all \( p \)-base, \( k \)-dimensional IVTs when \( \mathbb{N}_0^k \) is restricted to the subset \( Q^k \). Therefore, for \( (\eta_1, \eta_2, \ldots, \eta_k) \in Q^k \) we get
\[
IVT^{p,k}_{j_i}(\eta_1, \eta_2, \ldots, \eta_k) = f_i(\eta_1, \eta_2, \ldots, \eta_k)_p = m \in Q.
\]

Definition 11. The set \( T^\nu|_Q = \{ (IVT^{p,k}_{j_1}, IVT^{p,k}_{j_2}, \ldots, IVT^{p,k}_{j_\nu}) \} \) is the \( \nu \)-fold Cartesian product \( T^{p,k}_|Q| \times \cdots \times T^{p,k}_|Q| \) when \( \mathbb{N}_0^k \) is restricted to subset \( Q^k \), where \( IVT^{p,k}_{j_i} : Q^k \to \{ 1, 2, \ldots, \nu( \in \mathbb{N}) \} \).

Now, a local transition function of any \( i \)-th cell of a one-dimensional CA having \( p( \in \mathbb{N}) \) states and radius \( r( \in \mathbb{N}) \) can be represented by some \( IVT^{p,k}_{j_i} \), where \( p \in \mathbb{N} \), \( k = 2r + 1 \) and
\[
j_i \in \{ 0, 1, 2, \ldots, (p^{p_i^i} - 1) \} \text{ such that } \forall \ i \in \mathbb{Z}, \mu_i(c_i) \equiv IVT^{p,k}_{j_i}(c_i).
\]

If \( \tau \) is the global transition function of any \( \nu( \in \mathbb{N} \geq k) \)-celled CA, then for a global configuration \( C = (c_1, c_2, \ldots, c_\nu) \), it follows that (see [13]),
\[
\tau(C) \equiv (\ldots, IVT^{p,k}_{j_{i-1}}(c_{i-1}), IVT^{p,k}_{j_i}(c_i), IVT^{p,k}_{j_{i+1}}(c_{i+1}), \ldots),
\]
where \( \ldots, j_{i-1}, j_i, j_{i+1}, \ldots \in \{ 0, 1, 2, \ldots, (p^{p_i^i} - 1) \} \).

Example 8. Let the initial configuration of a CA having state set \{0, 1, 2\} be \( C_0 = (01201)_3 \) and the transition function be given by \( \tau(01201)_3 = \mu_1(101)\mu_2(012)\mu_3(120)\mu_4(201)\mu_5(010) = (02001)_3 \), where \( \mu_1, \mu_3, \mu_5 \) follow Wolfram code 377 and \( \mu_2, \mu_4 \) follow Wolfram code 588 for a three-state CA.

A local transition function can be equivalently represented by some \( IVT^{3,3}_{j_i} \), and thus it follows that \( \tau(01201)_3 \) is equivalent to \( IVT^{3,3}_{377}(1, 0, 1) \).
\[ \text{IVT}_{3,3}^{588}(0, 1, 2) \text{IVT}_{377}^{3,3}(1, 2, 0) \text{IVT}_{588}^{3,3}(2, 0, 1) \text{IVT}_{377}^{3,3}(0, 1, 0). \]

Conversely, for any \( \text{IVT}_{j}^{p,k} \) if \( k \) be odd, that is, if \( k = 2r + 1 \) for some \( r < k(\in \mathbb{N}) \), then \( \text{IVT}_{j}^{p,k}(\eta_1, \eta_2, \ldots, \eta_{2r+1}) \) will be equivalent to a local transition function of the \((r + 1)\)th cell in a CA with \( p(\in \mathbb{N}) \) states and \( 2r + 1 \) neighborhood whose underlying Wolfram code is \( j \in \{0, 1, \ldots, p^{bk} - 1\} \), given by
\[
\text{IVT}_{j}^{q,l}(\eta_1, \eta_2, \ldots, \eta_{2r+1}) = \text{IVT}_{j}^{p,k}(\eta_{r+1}) \equiv \mu_{r+1}(\eta_{r+1}).
\]

5. Some Algebraic Results on Cellular Automata and Integral Value Transformations

In this section, we present the theorems on the algebraic structures of the sets \( Q^\mathbb{Z} \) and \( T^\nu \). The proofs of the following theorems have been established in [13].

**Theorem 1.** \((Q^\mathbb{Z}, \circ)\) forms a monoid with regard to composition of global transition functions.

**Corollary 1.** \((\overline{Q^\mathbb{Z}}, \circ)\) forms a group with regard to composition of global transition functions. \( \overline{Q^\mathbb{Z}} \subseteq Q^\mathbb{Z} \) is the set of all invertible global transition functions of a CA having state set \( Q \).

**Theorem 2.** \((T^\nu, \circ)\) forms a monoid with regard to composition of IVTs.

**Definition 12.** Modular addition and multiplication of two local transition functions for a \( p(\in \mathbb{N}) \)-state CA are defined \( \forall i \in \mathbb{Z} \) as
\[
(\mu_i^1 \oplus_p \mu_i^2)(\overline{c}_i) = \mu_i^1(\overline{c}_i) \oplus_p \mu_i^2(\overline{c}_i) \quad \text{and} \quad (\mu_i^1 \otimes_p \mu_i^2)(\overline{c}_i) = \mu_i^1(\overline{c}_i) \otimes_p \mu_i^2(\overline{c}_i).
\]

**Definition 13.** Modular addition and multiplication of two \( p(\in \mathbb{N}) \)-base, \( k(\in \mathbb{N}) \)-dimensional IVTs (see [13]) are defined \( \forall (n_1, n_2, \ldots, n_k) \in \mathbb{N}^k_0 \) and \( j_1, j_2 \in \{0, 1, \ldots, p^{bk} - 1\} \), as
\[
(\text{IVT}_{j_1}^{p,k} \oplus_p \text{IVT}_{j_2}^{p,k})(n_1, \ldots, n_k) = \text{IVT}_{j_1}^{p,k}(n_1, \ldots, n_k) \oplus_p \text{IVT}_{j_2}^{p,k}(n_1, \ldots, n_k) = \left(f_{j_1}(a_{n_1}^{n_1}, \ldots, a_{n_k}^{n_k}) \oplus_p f_{j_2}(a_{0}^{n_1}, \ldots, a_{0}^{n_k}) \ldots \right) f_{j_1}(a_{n_1}^{n_1}, \ldots, a_{n_k}^{n_k}) \oplus_p f_{j_2}(a_{0}^{n_1}, \ldots, a_{0}^{n_k})
\]

and

https://doi.org/10.25088/ComplexSystems.32.2.115
\[
\left( \text{IVT}_{i_1}^{p,k} \otimes_p \text{IVT}_{i_2}^{p,k} \right)
\]
\[
(n_1, \ldots, n_k) = \text{IVT}_{i_1}^{p,k}(n_1, \ldots, n_k) \otimes_p \text{IVT}_{i_2}^{p,k}(n_1, \ldots, n_k) =
\]
\[
(f_{i_1}(a_0^{n_1}, \ldots, a_k^{n_k}) \otimes_p f_{i_2}(a_0^{n_1}, \ldots, a_k^{n_k})) = f_{i_1}(a_0^{n_{i_1}}, \ldots, a_k^{n_{i_k}}) \otimes_p f_{i_2}(a_0^{n_{i_1}}, \ldots, a_k^{n_{i_k}}),
\]
\[
\text{where}
\]
\[
n_s = (a_0^{n_s}, a_1^{n_s}, \ldots, a_k^{n_s})_p
\]
\[
\text{for } s = 1, 2, \ldots, k.
\]

**Theorem 3.** \((Q^2, \oplus_p, \otimes_p)\) forms a commutative ring with identity under the operations \(\oplus_p\) and \(\otimes_p\) defined as
\[
(\tau_1 \oplus_p \tau_2)(C) = \cdots (\mu_{i-1}^1 \oplus_p \mu_{i-1}^2) (c_{i-1} \cdot \mu_{i-1}^1 \oplus_p \mu_{i}^2) (c_{i-1} \cdot \mu_{i}^1 \oplus_p \mu_{i+1}^2) (c_{i+1} \cdot \mu_{i+1}^1 \oplus_p \mu_{i+2}^2) \cdots,
\]
\[
(\tau_1 \otimes_p \tau_2)
\]
\[
(C) = \cdots (\mu_{i-1}^1 \otimes_p \mu_{i-1}^2) (c_{i-1} \cdot \mu_{i-1}^1 \otimes_p \mu_{i}^2) (c_{i-1} \cdot \mu_{i}^1 \otimes_p \mu_{i+1}^2) (c_{i+1} \cdot \mu_{i+1}^1 \otimes_p \mu_{i+2}^2) \cdots,
\]
\[
\text{where } \tau_1, \tau_2 \in Q^2, C \in Q^2, \oplus_p \text{ denotes addition modulo } p \text{ and } \otimes_p \text{ denotes multiplication modulo } p \text{ for } |Q| = p (\in \mathbb{N}).
\]

**Remark 2.** The following example shows that \((Q^2, \oplus_p, \otimes_p)\) will not form a field under the operations \(\oplus_p\) and \(\otimes_p\) even if \(|Q| = p\) is prime.

**Example 9.** Consider an initial configuration \(C = (10011)\). Let
\[
\tau(10001)_2 = (\mu(110)\mu(100)\mu(000)\mu(001)\mu(011))_2.
\]
Let \(\mu\) follow Wolfram code 3(\(\sim (c_{i-1} \lor c_i)\)). Then
\[
\tau(10001)_2 = (\mu_3(110)\mu_3(100)\mu_3(000)\mu_3(001)\mu_3(011))_2 = (00110)_2.
\]
Now,
\[
\tau_{id}(10001)_2 = (\mu_{id}(110)\mu_{id}(100)\mu_{id}(000)\mu_{id}(001)\mu_{id}(011))_2 = (11111)_2.
\]
But \(\not\exists \tau^{-1}\) such that \((\tau \otimes_2 \tau^{-1})(10001)_2 = (11111)_2\) since \(\not\exists \mu^{-1}\) such that for all local configurations \(\mu \otimes_2 \mu^{-1} = 1\).

**Theorem 4.** \((T^v, \oplus_p, \otimes_p)\) forms a commutative ring under the operations \(\oplus_p\) and \(\otimes_p\) defined as
\[
\left( \left( \text{IVT}_{i_1}^{p,k} \right) \oplus_p \left( \text{IVT}_{i_2}^{p,k} \right) \right)(n_1, \ldots, n_v) = \left( \left( \text{IVT}_{i_1}^{p,k} \otimes_p \text{IVT}_{i_2}^{p,k} \right) \right)(n_1, \ldots, n_v) = \left( \left( \text{IVT}_{i_1}^{p,k} \right) \otimes_p \left( \text{IVT}_{i_2}^{p,k} \right) \right)(n_1, \ldots, n_v)
\]
and
\[
\left(\left(\text{IVT}^{p,k}_{i_1}, \ldots, \text{IVT}^{p,k}_{i_v}\right) \otimes_p \left(\text{IVT}^{p,k}_{j_1}, \ldots, \text{IVT}^{p,k}_{j_v}\right)\right)(n_1, \ldots, n_v) = \\
\left(\left(\text{IVT}^{p,k}_{i_1} \otimes_p \text{IVT}^{p,k}_{j_1}\right), \ldots, \left(\text{IVT}^{p,k}_{i_v} \otimes_p \text{IVT}^{p,k}_{j_v}\right)\right)(n_1, \ldots, n_v) = \\
\left(\left(\text{IVT}^{p,k}_{i_1} \otimes_p \text{IVT}^{p,k}_{j_1}(n_1), \ldots, \left(\text{IVT}^{p,k}_{i_v} \otimes_p \text{IVT}^{p,k}_{j_v}(n_v)\right)\right),
\right)
\]
where
\[i_s, j_s \in \{0, 1, \ldots, p^{p^k} - 1\}, p, k \in \mathbb{N}, \eta_s = (\eta^1_s, \ldots, \eta^k_s)\]
for \(s = 1, \ldots, v\).

**Theorem 5.** \(\left(T^p_{[\mathcal{O}], \otimes_p, \otimes_p}\right)\) forms a commutative ring with identity under the operations \(\otimes_p\) and \(\otimes_p\).

## 6. Modeling of Dynamical Systems Using Integral Value Transformation

It is a known practice to represent one-dimensional dynamical systems using two-state, three-neighborhood CAs [1]. For a system with \(s\) states having \(k\) neighborhoods, there can be \(s^k\) different transition rules. Consequently, computation using \(s(\in \mathbb{N} > 2)\)-state, \(k(\in \mathbb{N} \geq 3)\)-neighborhood CAs can be complex. For example, any system in one variable having base three can have \(3^3 = 27\) different functions, among which \(3^1 = 3\) functions are linear and the rest nonlinear. In such cases where \(s(\in \mathbb{N} > 2)\) and vertical dependency (see Example 2) among cells exists, IVT can serve as a suitable alternative. Various properties, applications and spacetime patterns of one-, two- and three-dimensional IVTs are discussed in [12, 26–29]. Here, we state the suitability of using IVTs for modeling of DNA sequence evolution.

### 6.1 Suitability of Using an Integral Value Transformation over a Cellular Automaton for DNA Sequence Evolution

1. Analogous to biological mutation, mutation in a genetic algorithm is a genetic operator for maintaining genetic diversity being transferred from one generation to the next of a population [30]. Thus, point mutation in a DNA sequence evolution is an unary operator. Now, a one-neighborhood, one-dimensional CA transition rule is practically not feasible since the transition of cell states of a CA depends on the neighboring cells. However, in the domain of IVTs like Example 1, computations using one-dimensional IVTs involving neighborhood-independent cells can be a suitable choice for mutation.

https://doi.org/10.25088/ComplexSystems.32.2.115
2. The chromosomal crossover, or crossing over, refers to the exchange of genetic materials between paired homologous chromosomes (one from each parent) during meiosis [31]. In evolutionary computation, it is [31] a binary operator that combines the genetic information of two parents for the generation of a new offspring. In a one-point crossover, a point on the chromosomes of both parents is picked randomly, and it is designated to be a “crossover point.” Between the two parent chromosomes, DNA sequences to the right of the crossover point are swapped. This results in the generation of two offspring, each of which carries some genetic information from both parents. In the example of a one-point crossover shown in Figure 1(a), a random crossover point is selected and a swapping between the tails of its two parents is done to get some new offspring.

In a two-point crossover, two crossover points are randomly picked from the parent chromosomes. Between the parent organisms, the DNA sequences in between the two crossover points are swapped. An example of a two-point crossover is given in Figure 1(b). Generalization of this strategy is a k-point crossover for any positive integer k, where the k crossover points [32] picked are similar to the computing process of IVTs explained in Example 3.

Since genetic operators like mutation and crossover do not require any neighborhood dependency, IVT architecture is a straightforward approach to experiment with these types of sequences.

![Figure 1](image.png)

**Figure 1.** (a) Example of one-point crossover. (b) Example of two-point crossover.
3. An IVT having four bases and one dimension can have four corresponding configurations, namely 0, 1, 2 and 3, producing $4^4 = 256$ different functions. For example, rule 59 for an IVT is represented by $0 \rightarrow 3$, $1 \rightarrow 2$, $2 \rightarrow 3$, $3 \rightarrow 0$.

But the simplest CA has three neighborhoods, which gives $4^3$ different transition functions. For defining any transition function for such a CA, $4^3 = 64$ local configurations are required. The configurations 0, 1, 2 and 3 in the IVT domain correspond to local configuration 000, 001, 002 and 003 in the CA domain, and rule 59 for a CA is represented by $000 \rightarrow 3$, $001 \rightarrow 2$, $002 \rightarrow 3$, $003 \rightarrow 0$ and $c_i \rightarrow 0$, where $c_i$ is any of the other 60 local configurations.

Thus for representing neighborhood-independent systems, a four-base, one-dimensional IVT can be more conveniently used compared to a four-state, three-neighborhood CA.

4. A DNA sequence can be visualized to be analogous to a linear sequence of a four-base number system. Therefore, DNA evolution can be modeled using a one-dimensional CA with rule matrix multiplication for linear CA (as reported in [33]). The DNA sequence has a correspondence with the CA lattice and the deoxyribose sugars (nucleotides) with the CA cells such that the four bases A, C, T and G correspond to four possible cell states of the CA.

However, for a long DNA sequence, rule matrix multiplication is difficult. Moreover, in general, for CA transition rules that are nonlinear, the corresponding rule matrix may not be obtained. Hence, DNA sequence evolution modeling in terms of one-dimensional IVTs having base four is a suitable alternative (see [17]).

### 6.2 Modeling Mutation Using Integral Value Transformation

Transitions and transversions are both point mutations that substitute a single nucleotide by another single nucleotide in a DNA sequence. A and G nucleotides are purines. C and T nucleotides are pyrimidines. Transitions are the point mutations that change a nucleotide of purine to another purine, or a nucleotide of pyrimidine to another pyrimidine. Whereas transversions substitute nucleotides between purines (having two hydrogen-carbon rings in their structures) and pyrimidines (having one hydrogen-carbon ring in their structures). Hence, transversions facilitate huge structural changes. But nature tries to resist such huge changes, and so in reality transitions occur more frequently than transversions [18]. To support this, we give the tabulated data of Table 5 in the Appendix on the basis of 39,680 genomic sequences of SARS-CoV-2 for Asian countries, reported in January 2021, collected from covidcg.org. Motivated by this important biological information, here we have tried to implement our methodology accordingly.

https://doi.org/10.25088/ComplexSystems.32.2.115
6.3 Implementation of Our Methodology
Here we represent DNA evolutions of two nucleotide sequences using IVTs for 100 iterations. The number of the four nucleotides A, C, T and G occurring at each iteration is counted. Hence the number of purines (A & G) and pyrimidines (C & T) occurring at each step of the evolution is calculated. DNA sequence-1 and DNA sequence-2 are of length 228 nucleotides each. Here nucleotides in these two sequences have variations in the 47th and 68th positions, as indicated by the underlined bold letters.

DNA Sequence-1
ATGTACTCATTCGTTTCGGAAGACAGGTACGTTAATAGTT
AATAACGTACTTTTTTCTTGCTTCTGTGGATTTCTTGCTA
GTACACTAGCCATCCTACTGCCTTGATTGTGTCGTAC
TGCTGGAATATTGTTAACGTGAGTCTTGTATAATTGTT
CTCGTATCTCCTGGTCTAAAATCTGAATTCCTGCTAGTT
CCTGATCTTCTGGTCTAA

DNA Sequence-2
ATGTACTCATTCGTTTCGGAAGACAGGTACGTTAATAGTT
AATAACGTACTTTTTTCTTGCTTCTGTGGATTTCTTGCTA
GTACACTAGCCATCCTACTGCCTTGATTGTGTCGTAC
TGCTGGAATATTGTTAACGTGAGTCTTGTATAATTGTT
CTCGTATCTCCTGGTCTAAAATCTGAATTCCTGCTAGTT
CCTGATCTTCTGGTCTAA

6.4 Numerical Representation of Nucleotides of a DNA Sequence
To evaluate the hidden characteristics of a set of DNA sequences, transformation, or mapping of the genomic data to a numeric representation, is required, as it simplifies the process of their quantitative analysis [34]. In this present study, the four nucleotides (A, C, G, T) of the DNA sequences are represented by numbers as follows:

\[ A \rightarrow 0, \ C \rightarrow 1, \ G \rightarrow 2, \ T \rightarrow 3 \]

Color representations of the nucleotides in the diagrams that follow:

\[ A \rightarrow \text{Red}, \ C \rightarrow \text{Blue}, \ G \rightarrow \text{Black}, \ T \rightarrow \text{Green} \]

The nucleotides corresponding to one-variable (one-neighborhood), four-state CA rules 59, 78 and 108 are given in Table 1. Rules 59 and 78 are nonlinear, and rule 108 is linear in nature. According to the numerical values of the nucleotides considered here, rule 59 reflects transversion, whereas rules 78 and 108 reflect transitions.
Table 1. Nucleotides corresponding to some rules for one-neighborhood, four-state systems.

Output: Spacetime Diagrams of the DNA Sequences Using Rules 59, 78 and 108

The spacetime diagrams of the evolution of the two DNA sequences for 100 iterations following rule 59 are shown in Figure 2, whereas those of rules 78 and 108 are shown in Figure 3. Four different colors, red, blue, black and green, are used for the four nucleotides A, C, G and T, respectively. The output also contains the number of nucleotide bases along with the purine and pyrimidine profiles at each iteration. These counts for DNA sequences 1 and 2 for some iterative steps under rules 59, 78 and 108 have been provided in Tables 2, 3 and 4, respectively.

![Figure 2](https://doi.org/10.25088/ComplexSystems.32.2.115)

**Figure 2.** (a) Shows the spacetime diagram of DNA sequence-1 using rule 59. (b) Shows the spacetime diagram of DNA sequence-2 using rule 59.
Figure 3. (a) Spacetime diagram of DNA sequence-1 using rule 78 or 108. (b) Spacetime diagram of DNA sequence-2 using rule 78 or 108.

<table>
<thead>
<tr>
<th>Iterative Step</th>
<th>Base(s)</th>
<th>DNA Sequence 1</th>
<th>DNA Sequence 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A</td>
<td>50</td>
<td>49</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>45</td>
<td>45</td>
</tr>
<tr>
<td></td>
<td>G</td>
<td>41</td>
<td>43</td>
</tr>
<tr>
<td></td>
<td>T</td>
<td>92</td>
<td>91</td>
</tr>
<tr>
<td></td>
<td>purine</td>
<td>91</td>
<td>92</td>
</tr>
<tr>
<td></td>
<td>pyrimidine</td>
<td>137</td>
<td>136</td>
</tr>
<tr>
<td>2</td>
<td>A</td>
<td>92</td>
<td>91</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>G</td>
<td>45</td>
<td>45</td>
</tr>
<tr>
<td></td>
<td>T</td>
<td>91</td>
<td>92</td>
</tr>
<tr>
<td></td>
<td>purine</td>
<td>137</td>
<td>136</td>
</tr>
<tr>
<td></td>
<td>pyrimidine</td>
<td>91</td>
<td>92</td>
</tr>
<tr>
<td>3</td>
<td>A</td>
<td>91</td>
<td>92</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>G</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>T</td>
<td>137</td>
<td>136</td>
</tr>
<tr>
<td></td>
<td>purine</td>
<td>91</td>
<td>92</td>
</tr>
<tr>
<td></td>
<td>pyrimidine</td>
<td>137</td>
<td>136</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>100</td>
<td>A</td>
<td>137</td>
<td>136</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>G</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>T</td>
<td>91</td>
<td>92</td>
</tr>
<tr>
<td>100</td>
<td>purine</td>
<td>137</td>
<td>136</td>
</tr>
<tr>
<td></td>
<td>pyrimidine</td>
<td>91</td>
<td>92</td>
</tr>
</tbody>
</table>

Table 2. Count of bases following transversion rule 59.
### Table 3. Count of bases following transition rule 78.

<table>
<thead>
<tr>
<th>Iterative Step</th>
<th>Base(s)</th>
<th>DNA Sequence 1</th>
<th>DNA Sequence 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A</td>
<td>50</td>
<td>49</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>45</td>
<td>45</td>
</tr>
<tr>
<td></td>
<td>G</td>
<td>41</td>
<td>43</td>
</tr>
<tr>
<td></td>
<td>T</td>
<td>92</td>
<td>91</td>
</tr>
<tr>
<td>1</td>
<td>purine</td>
<td>91</td>
<td>92</td>
</tr>
<tr>
<td></td>
<td>pyrimidine</td>
<td>137</td>
<td>136</td>
</tr>
<tr>
<td>2</td>
<td>A</td>
<td>41</td>
<td>43</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>92</td>
<td>91</td>
</tr>
<tr>
<td></td>
<td>G</td>
<td>50</td>
<td>49</td>
</tr>
<tr>
<td></td>
<td>T</td>
<td>45</td>
<td>45</td>
</tr>
<tr>
<td>2</td>
<td>purine</td>
<td>91</td>
<td>92</td>
</tr>
<tr>
<td></td>
<td>pyrimidine</td>
<td>137</td>
<td>136</td>
</tr>
<tr>
<td>3</td>
<td>A</td>
<td>50</td>
<td>49</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>45</td>
<td>45</td>
</tr>
<tr>
<td></td>
<td>G</td>
<td>41</td>
<td>43</td>
</tr>
<tr>
<td></td>
<td>T</td>
<td>92</td>
<td>91</td>
</tr>
<tr>
<td>3</td>
<td>purine</td>
<td>91</td>
<td>92</td>
</tr>
<tr>
<td></td>
<td>pyrimidine</td>
<td>137</td>
<td>136</td>
</tr>
<tr>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
</tr>
<tr>
<td>100</td>
<td>A</td>
<td>41</td>
<td>43</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>92</td>
<td>91</td>
</tr>
<tr>
<td></td>
<td>G</td>
<td>50</td>
<td>49</td>
</tr>
<tr>
<td></td>
<td>T</td>
<td>45</td>
<td>45</td>
</tr>
<tr>
<td>100</td>
<td>purine</td>
<td>91</td>
<td>92</td>
</tr>
<tr>
<td></td>
<td>pyrimidine</td>
<td>137</td>
<td>136</td>
</tr>
</tbody>
</table>

#### 6.5 Observation

It can be observed from the output that spacetime diagrams of transition rules 78 and 108 for both the DNA sequences show visual similarities. Whereas rule 59 depicting transversion portrays a pattern that is considerably different from the patterns of transition rules 78 and 108 applied here. It is evident that our methodology can identify and represent the nucleotide changes occurring in a DNA sequence. It is also possible to count numbers of purines and pyrimidines. Moreover, transitions and transversions can also be identified with the existing model. Apart from the visual similarity, some quantification of similarity in DNA sequences can be measured by calculating an average Hamming distance or through an average Manhattan distance.

https://doi.org/10.25088/ComplexSystems.32.2.115
Next, one-dimensional quaternary value rules are applied on some genomics sequences after transformation to 0/1/2/3, which is a numerical sequence. Previously, similar work on understanding dynamics of state transition diagrams (STDs) for all rules for some \( p \) and \( k \) values has been done using IVTs. STD dynamics of any one-dimensional rule is trivial; we can talk about spacetime dynamics of numerical sequences on successive iterations easily. Either the spacetime diagram will converge to steady state (or fixed) after at most \( k + 1 \) iterations or will repeat the pattern. For example, STD dynamics of rule 57 follows something like this: 3 \( \rightarrow \) 0, 2 \( \rightarrow \) 3, 1 \( \rightarrow \) 2, 0 \( \rightarrow \) 1. So, starting from the initial numerical sequence, in some position if there is 3, it follows 3 \( \rightarrow \) 0 \( \rightarrow \) 1 \( \rightarrow \) 2 \( \rightarrow \) 3 \( \rightarrow \) 0 \( \rightarrow \) 1 \( \rightarrow \) 2 \( \rightarrow \) 3 \( \cdots \) (repeated pattern); if in some position there is 1, it follows 1 \( \rightarrow \) 2 \( \rightarrow \) 3 \( \rightarrow \) 0 \( \rightarrow \) 1 \( \rightarrow \) 2 \( \rightarrow \) 3 \( \rightarrow \) 0 \( \cdots \) (after two iterations again fixed repeated pattern); and so on. So the spacetime diagram will be a fixed repetitive pattern after at most four

<table>
<thead>
<tr>
<th>Iterative Step</th>
<th>Base(s)</th>
<th>DNA Sequence 1</th>
<th>DNA Sequence 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A</td>
<td>50</td>
<td>49</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>45</td>
<td>45</td>
</tr>
<tr>
<td></td>
<td>G</td>
<td>41</td>
<td>43</td>
</tr>
<tr>
<td></td>
<td>T</td>
<td>92</td>
<td>91</td>
</tr>
<tr>
<td>1</td>
<td>purine</td>
<td>91</td>
<td>92</td>
</tr>
<tr>
<td></td>
<td>pyrimidine</td>
<td>137</td>
<td>136</td>
</tr>
<tr>
<td>2</td>
<td>A</td>
<td>50</td>
<td>49</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>92</td>
<td>91</td>
</tr>
<tr>
<td></td>
<td>G</td>
<td>41</td>
<td>43</td>
</tr>
<tr>
<td></td>
<td>T</td>
<td>45</td>
<td>45</td>
</tr>
<tr>
<td>2</td>
<td>purine</td>
<td>91</td>
<td>92</td>
</tr>
<tr>
<td></td>
<td>pyrimidine</td>
<td>137</td>
<td>136</td>
</tr>
<tr>
<td>3</td>
<td>A</td>
<td>50</td>
<td>49</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>45</td>
<td>45</td>
</tr>
<tr>
<td></td>
<td>G</td>
<td>41</td>
<td>43</td>
</tr>
<tr>
<td></td>
<td>T</td>
<td>92</td>
<td>91</td>
</tr>
<tr>
<td>3</td>
<td>purine</td>
<td>91</td>
<td>92</td>
</tr>
<tr>
<td></td>
<td>pyrimidine</td>
<td>137</td>
<td>136</td>
</tr>
<tr>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
<td>⋮</td>
</tr>
<tr>
<td>100</td>
<td>A</td>
<td>50</td>
<td>49</td>
</tr>
<tr>
<td></td>
<td>C</td>
<td>92</td>
<td>91</td>
</tr>
<tr>
<td></td>
<td>G</td>
<td>41</td>
<td>43</td>
</tr>
<tr>
<td></td>
<td>T</td>
<td>45</td>
<td>45</td>
</tr>
<tr>
<td>100</td>
<td>purine</td>
<td>91</td>
<td>92</td>
</tr>
<tr>
<td></td>
<td>pyrimidine</td>
<td>137</td>
<td>136</td>
</tr>
</tbody>
</table>

*Table 4. Count of bases following transition rule 108.*
iterations. This fact is true for all the rules, and that should be clear from the STD of the selected rule. So, analysis of both STD and spacetime diagrams of IVT rules can reveal a great deal of hidden information encoded in a DNA sequence.

Data processing through different transformations always plays a vital role in the field of computer science and other scientific disciplines. As an example, clinical testing over blood samples is used to analyze various chemical constituents for diagnosis and monitoring diseases. Similarly, from the taste of rice, which is a processed form of paddy, one can infer the quality of paddy as well. In this paper, we have processed different disease sequences with the help of IVT rules to generate a stack of sequences for further experiments with the DNA sequence.

7. Conclusion

This paper shows that cellular automaton (CA) transition rules can be conveniently represented through integral value transformations (IVTs) for specific applications like DNA evolution. Here, the algebraic structures under some binary operations for a set of transition functions of a one-dimensional CA and those of a set of IVTs have been presented. It can be observed that an odd-dimensional IVT becomes equivalent to a local transition function of a CA when the IVTs are restricted from positive integers to \( p( \in \mathbb{N}) \)-base numbers.

Modeling of dynamical systems with higher states can be conveniently done using IVTs, since transitions in IVTs can be independent of neighborhood interaction, unlike those of cellular automata (CAs). Evolution patterns of two DNA sequences have been analyzed with respect to the purine and pyrimidine counts at each iterative step, along with their spacetime diagrams. Transition and transversion have also been identified from the transformation rules used in our model. The degree of randomness among sequences can also be measured in various DNA evolution. In nature, point mutations (transversion and transition) do not take place at all the positions at the same time with respect to a particular nucleotide. Hence, our next effort will be focused on designing an IVT model capable of capturing and representing any point mutation more realistically. In the future, we will model the different crossover operations using IVTs and also apply the neighborhood-dependent CA rules versus neighborhood-independent IVT rules for a comparative analysis of both the models generating spacetime patterns over DNA sequences.

https://doi.org/10.25088/ComplexSystems.32.2.115
Appendix

In Table 5, which is based on the information from covidcg.org, the column “Position” specifies the position in the SARS-CoV-2 DNA sequence where the substitution mutations have occurred. The column “Ref Codon” refers to the codon existing in the specified position in a wild type (disease-free) sequence. The column “Alt Codon” specifies the new codon appearing due to point mutation.

As an example, consider position 241 in the wild type SARS-CoV-2 DNA sequence. Here, at position 241 the codon was CGT. Due to point mutation, the nucleotide “C” at the first position of the codon CGT is substituted by “T.” Since pyrimidine “C” is substituted by another pyrimidine “T,” here the occurring type of mutation is “transition.” From the table, it is clear that practically, point mutation occurs for a single nucleotide of a DNA sequence and it is a random process. Moreover, selection of nucleotides for substitutions (mutation) is a natural selection.

<table>
<thead>
<tr>
<th>Position</th>
<th>Ref Codon</th>
<th>Alt Codon</th>
<th>Position</th>
<th>Ref Codon</th>
<th>Alt Codon</th>
</tr>
</thead>
<tbody>
<tr>
<td>241</td>
<td>CGT</td>
<td>TGT</td>
<td>18027</td>
<td>GTG</td>
<td>GTT</td>
</tr>
<tr>
<td>313</td>
<td>CAG</td>
<td>TAG</td>
<td>18167</td>
<td>CCT</td>
<td>CTT</td>
</tr>
<tr>
<td>1059</td>
<td>CAC</td>
<td>CAT</td>
<td>18877</td>
<td>CTA</td>
<td>TTA</td>
</tr>
<tr>
<td>1163</td>
<td>AAT</td>
<td>ATT</td>
<td>19002</td>
<td>TTA</td>
<td>TTG</td>
</tr>
<tr>
<td>1738</td>
<td>GAA</td>
<td>AAA</td>
<td>19018</td>
<td>CCA</td>
<td>TCA</td>
</tr>
<tr>
<td>2455</td>
<td>CAT</td>
<td>TAT</td>
<td>19524</td>
<td>CTC</td>
<td>CTT</td>
</tr>
<tr>
<td>2836</td>
<td>CTC</td>
<td>TTC</td>
<td>20675</td>
<td>CAA</td>
<td>CTA</td>
</tr>
<tr>
<td>3037</td>
<td>CTA</td>
<td>TTA</td>
<td>21518</td>
<td>AGA</td>
<td>ATA</td>
</tr>
<tr>
<td>3267</td>
<td>TAC</td>
<td>TAT</td>
<td>21575</td>
<td>TCT</td>
<td>TTT</td>
</tr>
<tr>
<td>3634</td>
<td>CAA</td>
<td>TAA</td>
<td>22020</td>
<td>GAT</td>
<td>GAC</td>
</tr>
<tr>
<td>4346</td>
<td>CTC</td>
<td>CCC</td>
<td>22444</td>
<td>CCC</td>
<td>TCC</td>
</tr>
<tr>
<td>4354</td>
<td>GAA</td>
<td>AAA</td>
<td>23403</td>
<td>GGA</td>
<td>GGG</td>
</tr>
<tr>
<td>4510</td>
<td>GGT</td>
<td>AGT</td>
<td>23587</td>
<td>GAC</td>
<td>TAC</td>
</tr>
<tr>
<td>5700</td>
<td>TGC</td>
<td>TGA</td>
<td>23604</td>
<td>TCC</td>
<td>TCA</td>
</tr>
<tr>
<td>6235</td>
<td>TGT</td>
<td>CGT</td>
<td>23929</td>
<td>CAA</td>
<td>TAA</td>
</tr>
<tr>
<td>6310</td>
<td>CAC</td>
<td>AAC</td>
<td>24432</td>
<td>ACA</td>
<td>ACT</td>
</tr>
<tr>
<td>6312</td>
<td>CAC</td>
<td>CAA</td>
<td>25528</td>
<td>CTT</td>
<td>TTT</td>
</tr>
<tr>
<td>6380</td>
<td>TCT</td>
<td>TTT</td>
<td>25563</td>
<td>CAG</td>
<td>CAT</td>
</tr>
<tr>
<td>6433</td>
<td>CAT</td>
<td>TAT</td>
<td>25673</td>
<td>CTT</td>
<td>CCT</td>
</tr>
<tr>
<td>6573</td>
<td>TTC</td>
<td>TTT</td>
<td>25785</td>
<td>TGG</td>
<td>TGT</td>
</tr>
<tr>
<td>6592</td>
<td>ACC</td>
<td>GCC</td>
<td>26060</td>
<td>ACT</td>
<td>ATT</td>
</tr>
<tr>
<td>6990</td>
<td>TTC</td>
<td>TTT</td>
<td>26730</td>
<td>CTG</td>
<td>CTA</td>
</tr>
<tr>
<td>Position</td>
<td>Ref Codon</td>
<td>Alt Codon</td>
<td>Position</td>
<td>Ref Codon</td>
<td>Alt Codon</td>
</tr>
<tr>
<td>----------</td>
<td>-----------</td>
<td>-----------</td>
<td>----------</td>
<td>-----------</td>
<td>-----------</td>
</tr>
<tr>
<td>7728</td>
<td>TTC</td>
<td>TTT</td>
<td>26735</td>
<td>ACA</td>
<td>ATA</td>
</tr>
<tr>
<td>8031</td>
<td>TAA</td>
<td>TAG</td>
<td>28081</td>
<td>ATG</td>
<td>GTG</td>
</tr>
<tr>
<td>8076</td>
<td>CGT</td>
<td>CGC</td>
<td>28144</td>
<td>TAC</td>
<td>CAC</td>
</tr>
<tr>
<td>8782</td>
<td>CCA</td>
<td>TCA</td>
<td>28311</td>
<td>ACC</td>
<td>ACT</td>
</tr>
<tr>
<td>8917</td>
<td>CTT</td>
<td>TTT</td>
<td>28541</td>
<td>AGC</td>
<td>AAC</td>
</tr>
<tr>
<td>9286</td>
<td>CAA</td>
<td>TAA</td>
<td>28725</td>
<td>TCC</td>
<td>TCT</td>
</tr>
<tr>
<td>10156</td>
<td>CGT</td>
<td>TGT</td>
<td>28854</td>
<td>TTC</td>
<td>TTT</td>
</tr>
<tr>
<td>10376</td>
<td>ACC</td>
<td>ATC</td>
<td>28878</td>
<td>CAG</td>
<td>CAA</td>
</tr>
<tr>
<td>11083</td>
<td>GTA</td>
<td>TTA</td>
<td>28881</td>
<td>TAG</td>
<td>TAA</td>
</tr>
<tr>
<td>11916</td>
<td>ATC</td>
<td>ATT</td>
<td>28882</td>
<td>GGG</td>
<td>AGA</td>
</tr>
<tr>
<td>12049</td>
<td>CAA</td>
<td>TAA</td>
<td>28883</td>
<td>GGG</td>
<td>GCG</td>
</tr>
<tr>
<td>13730</td>
<td>GCT</td>
<td>GTT</td>
<td>28975</td>
<td>GTC</td>
<td>TTC</td>
</tr>
<tr>
<td>14408</td>
<td>CCT</td>
<td>CTT</td>
<td>29179</td>
<td>GCA</td>
<td>TCA</td>
</tr>
<tr>
<td>14625</td>
<td>TGC</td>
<td>TGT</td>
<td>29353</td>
<td>CAA</td>
<td>TAA</td>
</tr>
<tr>
<td>14708</td>
<td>GCT</td>
<td>GTT</td>
<td>29679</td>
<td>ATC</td>
<td>ATT</td>
</tr>
<tr>
<td>14805</td>
<td>TAC</td>
<td>TAT</td>
<td>29692</td>
<td>GTA</td>
<td>TTA</td>
</tr>
<tr>
<td>15324</td>
<td>AAC</td>
<td>AAT</td>
<td>29742</td>
<td>ACG</td>
<td>ACA</td>
</tr>
<tr>
<td>15438</td>
<td>ATG</td>
<td>ATT</td>
<td>29755</td>
<td>GAG</td>
<td>TAG</td>
</tr>
<tr>
<td>16650</td>
<td>CTC</td>
<td>CTT</td>
<td>29764</td>
<td>GTG</td>
<td>TTG</td>
</tr>
<tr>
<td>17502</td>
<td>TTC</td>
<td>TTT</td>
<td>29779</td>
<td>GAG</td>
<td>TAG</td>
</tr>
</tbody>
</table>

| 29868    | ATG       | AGA       |

Table 5. Codon alterations due to mutations in SARS-CoV-2 sequences.

References


https://doi.org/10.25088/ComplexSystems.32.2.115


https://doi.org/10.25088/ComplexSystems.32.2.115


