An Algorithmic Approach for Signal Measurement Using Symbolic Dynamics of Tent Map

Rajlaxmi Basu, Dhrubajyoti Dutta, Soumitro Banerjee, *Fellow, IEEE*, Violeta Holmes, Peter Mather

***Abstract*—The symbolic time series generated by a unimodal chaotic map starting from any initial condition creates a binary sequence that contains information about the initial condition. A binary sequence of a given length generated this way has a one-to-one correspondence with a given range of the input signal. This can be used to construct analogue to digital converters (ADC). However, in actual circuit realizations, component imperfections and ambient noise result in deviations in the map function from the ideal, which, in turn, can cause significant error in signal measurement. In this paper, we propose ways of circumventing these problems through an algorithmic procedure that takes into account the non-idealities. The most common form of non-ideality—reduction in the height of the map function—alters the partitions that correspond to each symbolic sequence. We show that it is possible to define the partitions correctly if the height of the map function is known. We also propose a method to estimate this height from the symbolic sequence obtained. We demonstrate the efficacy of the proposed algorithm with simulation as well as experiment. With this development, practical ADCs utilizing chaotic dynamics may become reality.**

***Index Terms*—initial condition estimation, reduced height map, symbolic dynamics, signal measurement**

I. INTRODUCTION

CHAOS in nonlinear dynamical systems has been broadly studied over the past few decades to understand the underlying deterministic principles in the apparent randomness. If the factors governing chaotic dynamics are better understood, the information of the physical states of such a dynamical system can be retrieved consistently. Thus, the presence of determinism in chaos has led to many applications in a wide range of areas including, but not limited to, control and synchronization of systems, secured communication for cipher key encryption and data analysis to understand complex patterns and cycles. In this work, we are investigating the scope of signal measurement using chaotic dynamics as an approach for analogue-to-digital (A/D) conversion. We achieve this by using a unimodal chaotic map as a tool to generate a symbolic sequence corresponding to an unknown initial signal through the iterative dynamics produced by the map. Applying the dynamical principles of the chaotic map, we describe an algorithmic approach to compute the magnitude of the initial signal from the generated sequence. The system thus works as an A/D converter (ADC).

Currently there is a wide range of architectures available for A/D converters of which successive approximation, delta-sigma ($\Delta\Sigma$), pipelined and modified flash ADC types are the most commonly used architectures. Each of these architectures has some benefits along with some shortcomings. The selection criteria for any particular type depends on the application specifications such as precision, speed, chip area and power dissipation as each type can be analysed through several performance metrics as proposed by Walden [1] and Robert *et al*. [2]. As summarised by Bashir *et al*. [3], one has to choose a trade-off between resolution, power dissipation, and speed for a flexible design architecture. For successive approximation type ADCs, improved resolution is achieved through higher level of design complexity and resource consumption, but at the cost of reduced speed. $\Delta\Sigma$ ADCs are mostly preferred for better precision and low power consumption; however, due to oversampling, it offers moderate speed. Also, higher order system implementation requires a large amount of area and the stability factors are affected by the order of modulation.

In the flash type ADCs, the quantisation is mainly done through a parallel implementation of comparators and these are therefore well known for high speed operations; however, it is challenging to achieve higher bit resolution as the number of comparators double each time a bit resolution is increased by 1. Better resolution is achieved through several hybrids of flash architecture such as interpolation type which reduces the number of pre-amplification units and drastically reducing the chip area. Since the number of latches required is still the same as the classical flash architecture, additional folding stages are often incorporated to further reduce resources. Each folding stage includes a fine grain ADC and a coarse grain ADC with a folding circuit whose accuracy is critical and expensive. Another flash based architecture is pipelined ADC, which involves series implementation of quantisation blocks that are

R. Basu, D. Dutta, V. Holmes and P. Mather are with the Engineering and Technology Department, School of Computing and Engineering, University of Huddersfield, Queensgate, Huddersfield, W. Yorks., UK, HD1 3DH (email: rajlaxmi.basu@hud.ac.uk, dhruba.dutta@hud.ac.uk, v.holmes@hud.ac.uk, p.j.mather@hud.ac.uk).

S. Banerjee is with the Department of Physical Sciences, Indian Institute of Science Education & Research, Kolkata, Mohanpur Campus, Nadia-741246, India (email: soumitro@iiserkol.ac.in).

operated in parallel. Although these ADCs improve speed, the resources used for the pipelined architecture increases the chip area as well as the cost. Thus, it is evident that, for improved performance, most of the ADC architectures rely heavily on additional quantization blocks such as increased number of comparators or coarse ADC/DAC as well as folding circuits leading to increased resource consumption, which result in increased chip area with greater design complexity and high power consumption.

Recent advances in our understanding of chaotic dynamics has brightened the prospect of developing a feasible ADC that uses chaotic dynamics to convert a given input signal into a string of digital values [4,5,6], which may be able to overcome some of the difficulties mentioned above. Metropolis *et al*. [7] as well as Wang and Kazarinoff [8] have shown in theory that symbolic codes generated through unimodal chaotic maps can be ordered such that they have a correspondence to a point or an interval within a measurable state space. Therefore, if such codes can be obtained from a physically implemented chaotic system, those symbolic identities may be useful for applications like signal measurement. Due to the 'stretching and folding' nature of the unimodal functions, it is also possible to partition the phase space into intervals with unique symbolic signature. Since chaotic maps are simple mathematical functions which can be easily implemented physically, a single block of chaotic map can be reused iteratively to generate the dynamics and symbolic representations corresponding to an input signal.

In practice, the analogue signal to be measured is fed as the initial condition to a hardware-implemented chaotic circuit. As the dynamics evolves in time, a 'coarse grained' symbolic code is generated that holds the key information of the originating point, parametric factors, and the footprints of the dynamics. For our application, we have chosen the tent map as the suitable chaotic function to generate symbolic dynamics because it is simple to implement, and has no window of periodicity within the ergodic range of the map parameter [9]. The symbolic dynamics produced by the tent map are Gray codes. This code can be processed using a straightforward numerical exercise to estimate the initial condition. However, this is possible if the tent map generated by the circuit is 'ideal' i.e., its domain is exactly [0,1].

When we implement the map physically, it is subjected to several other factors such as component imprecision and noise, which may significantly reduce the parametric domain of the map. With a reduced height tent map, converting the generated Gray code sequences directly into the corresponding decimal values leads to an incorrect mapping and therefore measurement accuracy suffers, as observed in [4]. If one wants to circumvent the problem using a lengthier symbolic time series, it becomes demanding in terms of resource required. A similar analysis has been conducted theoretically in [5], which showed that the use of a map with ideal parameter is preferable. This, however, limits the applicability because in a physical implementation, deviation of the parameter is inevitable, as can be seen from the work of Kapitaniak et al. [6] who had also previously attempted to measure electrical signals in a similar way. They observed that the measured outcomes were greatly affected by the errors introduced due to the offsets and tolerances of the components used in the physically implemented map, which significantly reduced the parametric domain of the map.

In this paper, we develop an algorithm that enables one to measure a signal with reasonable accuracy even when the map parameter deviates from the ideal. For this, three problems had to be solved. Firstly, when the height of the map is less than 1, the map dynamics eventually gets confined within a range of the state space. Therefore, some sequences corresponding to the points that are not visited by the dynamics will not be appearing as well in the symbolic dynamics; those sequences are classified as forbidden sequences [10]. We have shown that, even though there are forbidden sequences, information regarding the points outside the bounding region can still be realised using the symbolic dynamics produced within the boundary. Secondly, due to such constraints on the dynamics, when sequences are converted back to the corresponding initial values, the intervals appear to be squeezed in with overlapping or colliding partitions within the bounded state space. For a reduced parameter, the partitions created on every stage of map operation, are shifted away from the ideal positions [11]. Since conventional techniques (while estimating the initial condition) assume that the subintervals created in each step of iteration to be equal and symmetric, correspondence between the symbolic sequence and the initial conditions appear to be lost. The algorithm we have developed accounts for this shift and enables one to estimate the initial condition in spite of the fact that the partitions of the phase space of the map are unequal in size. Thirdly, since in a practical scenario the map parameter can vary from time to time due to changes in the parameters of the circuit, we have developed an algorithm to estimate the map parameter from the obtained symbolic sequence. All such analysis can be carried out in the digital domain, thus making the potential system architecture less complicated at the hardware level.

To develop the initial condition estimation algorithm, the dynamical properties of both the full height and the reduced height tent map have been thoroughly studied and applied. We define tent map and its dynamical features that are relevant to our work in section II and III, followed by the general view of the dynamical attractor in case of reduced parameter in section IV, and describe ways to determine the parameter value from the symbolic sequences. Also, we show, how the apparent loss of definition of the points are still preserved through the symbolic dynamics. In section V, we show how the partitioning of the state space is carried out according to the orientation preserving and reversing nature of the map and how the size of the sub-interval is altered corresponding to the parametric value of a reduced height map. In section VI, we propose a suitable technique to deal with the initial condition estimation problem in the form of an algorithm based on interval arithmetic that can be implemented easily into processing devices like microcontrollers and field programmable gate arrays (FPGA). To evaluate the proposed method, mathematical simulations with detailed analysis are presented in section VII. In section VIII we show the results for a physically implemented system.

In section IX we summarize our work and include some comments regarding practical applications.

II. Context

We consider a class of chaotic maps called *unimodal maps*, $\mathcal{F}$. If a map $f \in \mathcal{F}$ such that $f : I \to I$ where $I = [a,b] \subset \mathbb{R}$, a < b satisfies the following conditions:

1. f has a unique maximum f_{max}, in the interval I,
2. $f_{max} = f(x_c)$ (where $x_c \in I$ is usually called the critical point),
3. f is monotonically increasing in the interval $[a,x_c]$ and monotonically decreasing in the interval $[x_c,b]$, then f is unimodal.

The class $\mathcal{F}$ consists of certain maps that can be defined using a *control parameter*, μ, such that $f_\mu(x) \in \mathcal{F}$ holds for $x \in I$, $\mu \in J \subset \mathbb{R}$ and $f_\mu(x)$ is a map on $I \times J$. In particular, the map considered for our application, the tent map $T \in \mathcal{F}$, belongs to a family of parametric self-maps $f_\mu : I \to I$ such that $I = [0,1]$, $J = [0,1]$, and $T(x)$ can be defined as

$$T(x) = f_\mu(x) = \begin{cases} 2\mu x & 0 \le x \le x_c \\ 2\mu(1-x) & x_c < x \le 1 \end{cases}. \quad (1)$$

where $x_c = 0.5 \in I$ is the critical point of the map. For the map to be chaotic, $J = (0.5,1]$. In the closed interval $I \subset \mathbb{R}$—also known as the state space of the map—the iterates of $T(x)$ is defined as $x_{i+1} = T(x_i)$, $i \in \mathbb{N}_0$ (where $\mathbb{N}_0 = \{0\} \cup \mathbb{N}$) such that,

1. $x_0 = T^0(x) = x$
2. $x_{i+1} = T^{i+1}(x_0) = T(T^i(x_0)) = T(x_i)$
3. $T(0) = T(1) = 0$
4. $T_{max} = T(x_c) \le 1$, where T_{max} is the maximum height of the map, for $0 \le \mu \le 1$
5. $T(T_{max}) = T^2(x_c) \ge 0$, where $T(T_{max})$ is the dynamic minimum of the long-term trajectory.

The collective set of n iterates, i.e. the set of n points visited by the trajectory of a tent map can be referred to as the orbit of an initial condition x_0 and is defined as $\mathcal{O}_T(x_0) = \{T^0(x_0), T^1(x_0), T^2(x_0), \ldots, T^n(x_0)\}$. In this work the *input signal* sets the *initial condition*, and so the two terms will be used interchangeably. Also, when $T_{max} = 1$ (for $\mu = 1$) we call it an "ideal" case, and maps with $T_{max} \le 1$ constitute the "non-ideal" case.

Given that chaotic maps are sensitive to initial conditions, an infinitesimally small change in the initial condition may result in substantially diverging trajectories and due to the folding nature of the map, points in the closed interval $I \subset \mathbb{R}$ will *eventually* map on to every other point in $I \subset \mathbb{R}$, or arbitrarily close to it [12]. Therefore, unique trajectories can be generated for any arbitrary point in $I \subset \mathbb{R}$. For the application of ADC, we choose to utilise symbolic dynamics to involve lesser resources. In the following section, the symbolic sequence generated by tent map has been described with its general features and functionalities that are relevant to our study.

III. Symbolic Dynamics and Coding of Intervals

The orbit of a tent map $\mathcal{O}_T(x)$, can be transformed into a symbolic sequence $\mathcal{S}_{n+1}$ of length $n+1$ where $\mathcal{S}_{n+1}(T,x) = s(x_0)s(x_1)s(x_2)\ldots s(x_n)$ such that $s : [0,1] \to \{0,1\}$ is defined as

$$s(x_i) = \begin{cases} 0 & x_i \le x_c \\ 1 & x_i > x_c \end{cases}. \quad (2)$$

Furthermore, it has been shown that, the symbolic sequences generated are Gray codes [13].

As per the desired system (Fig. 1), successful determination of an unknown voltage signal $x_0 \in I$ involves generating corresponding symbolic sequence $\mathcal{S}_{n+1}(T,x)$ through $T(x)$ in the analogue domain and the $\mathcal{S}_{n+1}(T,x)$ shall be further processed in a digital domain. The analogue circuit [14] of the tent map can be incorporated in the measurement system. On every i^{th} iteration, the state space I is partitioned into 2^{i+1} mutually exclusive sub-intervals I_j^i where $0 \le j \le (2^{i+1}-1)$ is the count of the sub-interval increasing from the left endpoint 0 to the right endpoint 1 within I and i is the iteration count [13]. The input signal to the function must therefore belong to any *one* of the sub-intervals. This reduces our problem down to identification of the correct sub-interval for the corresponding input signal through its symbolic signature. The following properties relate the symbolic sequence $\mathcal{S}_{n+1}(T,x)$ to the sub-intervals generated by the map.

1. Every $x \in I_j^i$ result into the same symbolic sequence $\mathcal{S}_{i+1}(T,x)$
2. If initial conditions $x \in I_j^i$ and $\hat{x} \in I_{j+1}^i$, then $\mathcal{S}_{i+1}(T,x)$ and $\mathcal{S}_{i+1}(T,\hat{x})$ differ by only one bit
3. $I_0^i \cup I_1^i \cup I_2^i \cup \ldots \cup I_{2^{i+1}-1}^i = I$
4. $I_j^i \cap I_k^i = \emptyset$ for $j \ne k$

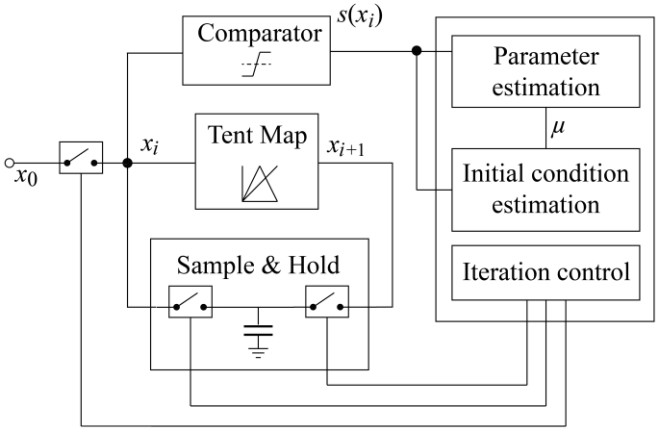


Fig. 1. The proposed block diagram of the measurement system using tent map implemented in analogue domain where unknown signal enters as an initial condition x_0 and output of each stage is fed back as an input for the next to complete iterations up to n times. The iterations can be executed through digitally controlled sample and hold technique. The comparator output (symbols) of each i^{th} stage should be received by the digital processing block for further processing.

TABLE I
CORRESPONDENCE BETWEEN SEQUENCES AND INPUT INTERVALS

j	$\mathcal{S}_3(T,x)$	Binary	GON
0	000	000	0
1	001	001	0.125
2	011	010	0.25
3	010	011	0.375
4	110	100	0.5
5	111	101	0.625
6	101	110	0.75
7	100	111	0.875

Therefore from the properties 1, 2 and 4, the symbolic sequence $\mathcal{S}_{n+1}(T,x)$ can be interpreted as $n+1$ bit long unique symbolic identity that corresponds to a sub-interval of the size I_j^n and so, the longer the symbolic sequence, the narrower will be the size of the intervals. Each such j^{th} interval can be identified by the corresponding symbolic sequence $\mathcal{S}_{n+1}$. The order of the symbolic sequences, as shown in [13], corresponds to the order $j = 1, 2, 3, \ldots, 2^{n+1}$ according to which the intervals I_j^n are ordered in I. For example, for all $\mathcal{S}_3(T,x)$, the order of the possible sequences corresponding to j can be seen from Table I. Therefore, for $\mathcal{S}_{n+1}(T,x)$, I_j^n can be written as $I_{\mathcal{S}_{n+1}}^n$ and can be used as a basis to identify the originating interval of an initial condition.

The initial conditions directly correspond to their originating intervals as long as the map retains the full height, when their symbolic signatures $\mathcal{S}_{n+1}(T,x)$ are converted to the corresponding binary codes $\mathcal{B} : b_0b_1b_2...b_n$

$$b_i = \begin{cases} s(x_i) & i = 0 \\ b_{i-1} \oplus s(x_i) & i > 0 \end{cases}. \tag{3}$$

$\mathcal{B}$ is further converted to the real values. This conversion from $\mathcal{S}_{n+1}(T,x)$ to real number is referred to as Gray Ordering Number (GON), given by the transformation

$$\mathrm{GON}(\mathcal{S}_{n+1}) = \sum_{i=0}^{n} b_i^{-(i+1)}. \tag{4}$$

and can be ordered by its magnitude as described in [13]. Table I shows GONs for a 3-bit sequence generated using $T^2(x_0)$ for inputs (x_0) with a step-size of 0.125. Considering a longer sequence will result in identification of input signals with a finer step size.

IV. DYNAMICS IN REDUCED PARAMETER CONDITIONS

In a realistic situation, it may not be possible to hold the parameter $\mu = 1$ constant. Under such non-deal condition, when the map height (parameter) is reduced i.e. $\mu < 1$, it undergoes certain changes in its dynamical characteristics. The property of the dynamical attractor of the tent map is related to its maximum height $T_{max} = T(x_c)$. Also, due to the folding nature of the tent map, the minimum value of the attractor can be determined as $T_{min} = T(T_{max}) = T(T(x_c))$. Thus $T_{max} = \mu$, and $T_{min} = 2\mu(1-\mu)$. Over a long term dynamics, it can be observed that points originating from arbitrary locations of I will *eventually* be attracted towards and be trapped within $I' = [T_{min},T_{max}] = [2\mu(1-\mu),\mu]$, where $I' < I$, when $\mu < 1$ (Fig. 2). The dynamics will continue as a never-ending process within I' since both fixed points 0 and $2\mu/(1+2\mu)$ are unstable for $\mu > 0.5$, and the orbit is chaotic. Also, from the available symbolic dynamics, the value of μ can be realised through the symbolic signature corresponding to the point T_{max} or T_{min}.

The properties of the dynamical maximum and minimum of the unimodal maps can also be studied symbolically through Milnor-Thurston Kneading Theory. When the critical point x_c is iterated through a tent map, the corresponding symbolic sequence $\mathcal{S}_{n+1}$ is known as the Kneading Sequence $\mathcal{K}$ [12] and can be expressed as

$$\mathcal{K} = \mathcal{S}_{n+1}(T,x_c) = s(T^0(x_c))s(T^1(x_c))s(T^2(x_c)) \ldots s(T^n(x_c)),\ n \in \mathbb{N}_0. \tag{5}$$

$\mathcal{K}$ can be a useful tool to realise the sequences corresponding to T_{min} and T_{max}. Considering a shift operator σ such that $s(T^{n+1}(x_0)) = \sigma(s(T^n(x_0)))$, operations of σ over $\mathcal{K}$ gives $\mathcal{S}_{max} = \sigma(\mathcal{K})$ and $\mathcal{S}_{min} = \sigma(\sigma(\mathcal{K}))$ which is *analogous* to the properties $T_{max} = T(x_c) = \mu$ and $T_{min} = T(T_{max}) = T(T(x_c)) = 2\mu(1-\mu)$ respectively. The parameter μ is thus recoverable using $\mathcal{S}_{min}$ or $\mathcal{S}_{max}$. For initial conditions x_0, we choose a certain number of $m \in \mathbb{N}$ transient iterations such that, for $i > m$, we get $x_i \in [T_{min},T_{max}]$. The choice of m is an empirical estimate, based on both the initial condition and the parameter of the map, and when both the factors remain to be unknown, m is chosen to be large enough to ensure that the subsequent iterates belong to $[T_{min},T_{max}]$. For $x_0 < T_{min}$, after m iterations when $x_{m+1} \geq T_{min}$, the corresponding symbolic sequence $\mathcal{S}_{n+1}(T,x_0)$ will be a string of m zeros followed by a sequence $\mathcal{S}_{n-m+1}(T,x_m) \in [\mathcal{S}_{min},\mathcal{S}_{max}]$. For $x_0 > T_{max}$, $x_0 \in [x_c,1]$, $s(x_0) = 1$, and therefore $T(x_0) < T_{min} \in [0,x_c]$ will continue with the aforementioned behaviour. Such cases will have $s(x_0) = 1$ leading a string of $m-1$ zeros followed by a sequence $\mathcal{S}_{n-m+1}(T,x_{m+1}) \in [\mathcal{S}_{min},\mathcal{S}_{max}]$. When the m transient symbols are discarded, the ordering of the GONs can be matched to the ordering of $\mathcal{S}_{min}$ through $\mathcal{S}_{max}$.

$$\mathrm{GON}_{min} \prec \ldots \prec \mathrm{GON}_{max} \equiv \mathcal{S}_{min} \prec \ldots \prec \mathcal{S}_{max} \tag{6}$$

Then, theoretically speaking, there will be no $\mathcal{S}_{n-m+1}(T,x_m)$ sequence appearing in the dynamics of $T(x_0)$ for a given μ whose corresponding GON can be found outside the range $[\mathrm{GON}_{min},\mathrm{GON}_{max}]$ and then any sequence $\mathcal{S}_{n+1}(T,x_0)$ outside $[\mathcal{S}_{min},\mathcal{S}_{max}]$, are treated as *forbidden sequences* [10] while all the

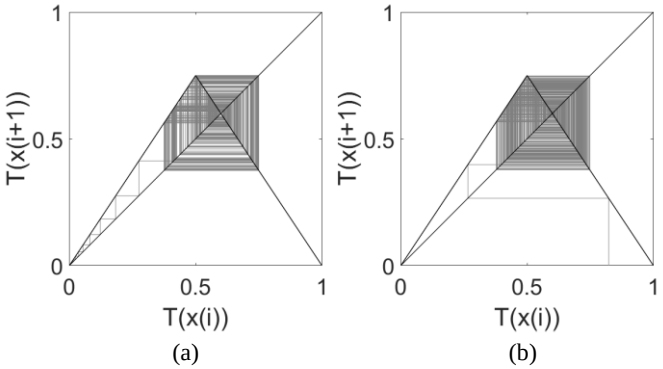


Fig. 2. Cobweb diagram of map attracted within $[T_{min},T_{max}]$ for $\mu = 0.75$; (a) $x_0 = 0.000124$ and (b) $x_0 = 0.823$; $n = 300$.

TABLE II.
COMPARISON OF IDEAL AND NON-IDEAL SEQUENCE

x_0	$\mathcal{S}_{16}$ for $\mu = 1$	$\mathcal{S}_{16}$ for $\mu < 1$	GON($\mathcal{S}_{16}$) for $\mu < 1$
0. 1951	0010100100001011	0011101101001110	0.17790
0. 1952	0010100100000100	0011101101001000	0.17796
0. 1953	0010100100000000	0011101101011010	0.17802
0. 1954	0010101100000111	0011101101011100	0.17808
0. 1955	0010101100001010	0011101101010110	0.17814

An arbitrary set of points with a step size of 0.0001, $\mathcal{S}_{16}$ generated using full ($\mu = 1$) and reduced ($\mu = 0.95$) $T(x_0)$ with GON calculated for the sequences generated with $\mu < 1$.

sequences within $[\mathcal{S}_{min},\mathcal{S}_{max}]$ are termed as *allowed sequence*. Thus discarding the transient m symbols from $\mathcal{S}_{n+1}(T,x_0)$ allows one to search for $\mathcal{S}_{min}$ or $\mathcal{S}_{max}$ in the allowed sequence domain, which in turn aids us to determine the reduced parameter μ from the symbolic trajectories [15,16]. However, in order to determine the initial condition x_0 successfully, the transient m symbols cannot be discarded because the information regarding a significant portion of the dynamic trajectory from the originating point of x_0 is contained in it. The originating interval can therefore be determined by realising the partitions generated by the map iterations on the state space I with respect to the 0's and 1's in the available symbolic sequence. Thus, we discard m transient symbols to determine μ, but do not discard it when determining the initial condition x_0.

When $\mu < 1$, as the partitioning of the state space continues, the sub-intervals created are unequal (Fig. 3). The partitioned sub-intervals are therefore squeezed in the state space towards x_c by a factor of μ (described in detail in section VI). The farther the parameter is away from its ideal value, the greater will be the amount of shift in the partitions from its ideal positions. Therefore, the initial points are redistributed to the adjacent intervals, causing the points to associate with a different symbolic sequence. If any sequence $\mathcal{S}_{n+1}$ generated with $\mu < 1$

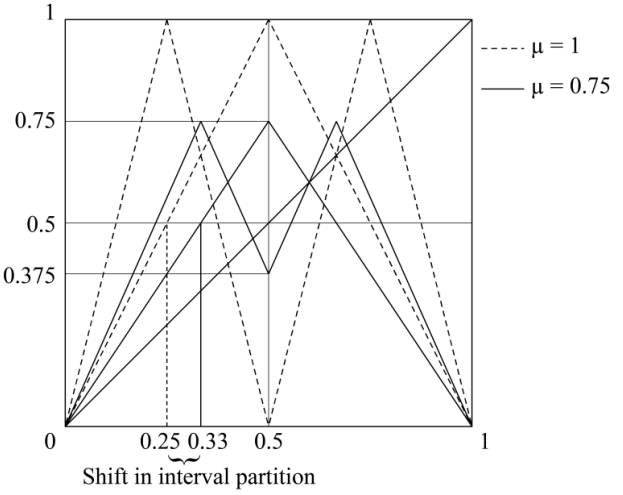


Fig. 3. The first two successive iterations of a reduced height (μ = 0.75) tent map are superposed over two iterations of a full-height (μ = 1) tent map. In the figure, one of the partitions creating the sub-intervals in the second iteration, can be seen to be shifted from 0.25 in case of the full-height map to 0.33 in case of the reduced height map. This results in uneven sub-intervals created in the state space.

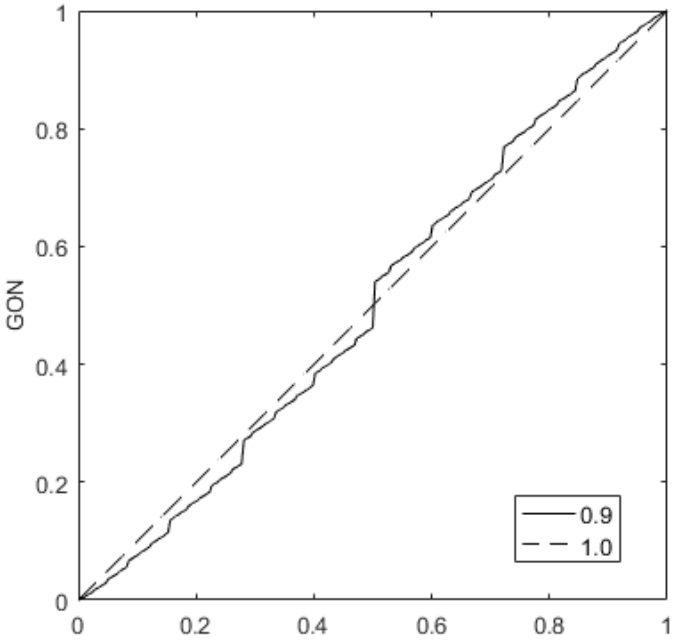


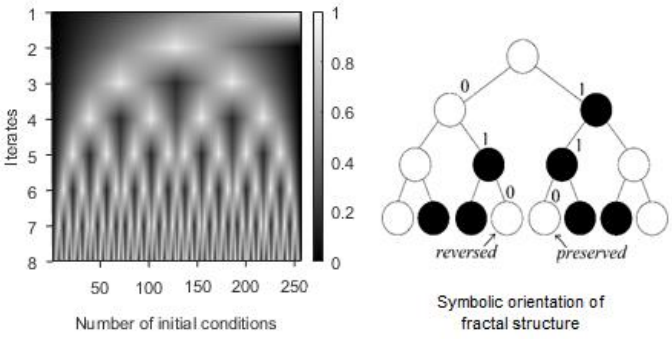
Fig. 4. A symbolic sequence is generated, once using μ = 0.90 (solid) and once using μ = 1 (dashed) . The GON for the sequence is calculated both times and plotted against the initial condition, showing signification deviations from the ideal, in case of μ = 0.90.

is converted back to the real value using base 2 (i.e. by calculating GON using (3) and (4)), it does not converge to a correct solution as can be verified by Table II. In fact, even for a small change in parameter (μ = 0.9), GONs deviate significantly from their ideal values, as can be seen in Fig. 4. However, the originating interval of x_0 can be recovered with respect to the partitions if the measure of the shift is considered for every symbol in $\mathcal{S}_{n+1}$. Also, many-to-one mapping on $\mathcal{S}_{n+1}$ within an interval is overcome by considering adequate number of samples, so that partitions of the sub-intervals are suitably generated and the codes differ by at least one symbol. The number of symbols to be considered will depend upon the desired resolution of x_0 which has been further elaborated in Section VII. As can be verified from Table II, a collection of points with fixed interval size for a given μ, have unique symbolic signatures. Therefore, as shown in the following sections, to correctly estimate the initial condition, the interval arithmetic needs to be modified and performed along the symbolic sequence so that the partitioning of the intervals can be traced with respect to μ.

V. INTERVAL ARITHMETIC

For any initial condition x_0 with symbolic sequence $\mathcal{S}_{n+1}$, we know that $x_0 \in I^n_{\mathcal{S}_{n+1}}$. Moreover, the i^{th} symbol in $\mathcal{S}_{n+1}$ indicates whether x_i belongs to the left or right of x_c, i.e., to I^0_0 or I^0_1. Thus, for every $s(x_i) \in \{0,1\}$, if $x_i = T^i(x_0) \in I^0_{s(x_i)}$, considering the inverse relation we get $x_0 \in T^{-i}(I^0_{s(x_i)})$. Therefore, for an n+1-bit sequence, combining this relation for every x_i, the originating interval $I^n_{\mathcal{S}_{n+1}}$ can be defined as

$$I^n_{\mathcal{S}_{n+1}} \equiv \bigcap_{i=0}^{n} T^{-i}(I^i_{s(x_i)}). \quad (7)$$



(a) (b)

Fig. 5. (a) Fractal nature exhibited by real dynamics of the map (shown up to 8 iterations). (b) The fractal orientation of the sub-intervals according to the symbols shown for up to three levels for up to 3 levels. Number of 1's determine the orientaion of the current sub-interval. Odd 1's result in orientation reversing, even 1's indicate orientation is preserved.

To illustrate, if $\mathcal{S}_{n+1}$ = 010...$s(x_n)$ is considered, the first symbol indicates that the initial condition, $x_0 \in I_0^0$. After applying the tent map function once, the iterate $T(x_0) \in I_1^0$, hence, $x_0 \in T^{-1}(I_1^0)$. So, considering the first two symbols, $x_0 \in I_0^0 \cap T^{-1}(I_1^0) \equiv I_{01}^1 \subset I_0^0$ [9,16]. In this manner, following all the symbols in the sequence, the originating sub-interval can be identified as

$$x_0 \in I_0^0 \cap T^{-1}\left(I_1^0 \cap T^{-1}(I_0^0 \ldots)\right) \equiv \ldots \subset I_{010}^2 \subset I_{01}^1 \subset I_0^0. \quad (8)$$

Since the map is non-invertible (every point has two inverses), the immediate question that arises from (7) is that when the inverse operation T^{-1} of the tent map function on an interval is performed, how the restriction for T^{-1} is chosen. This depends on the "orientation" of the map on the sub-interval that is created by the current iterate [9]. *Orientation* of an interval is determined by the slope of the function in that interval: a positive slope implies an orientation-preserving interval while a negative slope implies an orientation-reversing interval. The orientation reversal is responsible for the reversal of the lexicographic order of the symbolic signature (and thus generates a Gray code). Therefore, the orientation of the interval $I_{\mathcal{S}_{i+1}}^i$ can be determined from the sequence $\mathcal{S}_{i+1}$ associated with the i^{th} iterate.

In Fig. 5 (a) the fractal nature of the real iterates through a tent map can be observed, which when coarse-grained into symbols, result in mirroring orientation. As can be seen from Fig. 5 (b), if the i^{th} iteration of the tent map occurs on an orientation-reversing interval, the orientation of $I_{\mathcal{S}_{i+1}}^i$ gets reversed from that of $I_{\mathcal{S}_i}^{i-1}$. Hence, up to the i^{th} iteration, occurrence of the orientation-reversing iteration for an even number of times restores the orientation of $I_{\mathcal{S}_{i+1}}^i$, while an odd count of the same behaviour results in a reversal. Since each orientation-reversing iteration generates the symbol '1', we can determine the orientation of the interval $I_{\mathcal{S}_{i+1}}^i$ by checking if α_i is even (preserved) or odd (reversed), where α_i is given by (9).

$$\alpha_i = \alpha_{i-1} + s(x_i) \quad (9)$$

Therefore, the restrictions of the inverse operation T^{-1} of the tent map function can be chosen as

$$I_{\mathcal{S}_{i+1}}^i = T^{-1}\left(I_{\mathcal{S}_i}^{i-1}\right) = \begin{cases} \frac{I_{\mathcal{S}_i}^{i-1}}{2\mu} & \alpha \text{ is even} \\ 1 - \frac{I_{\mathcal{S}_i}^{i-1}}{2\mu} & \alpha \text{ is odd} \end{cases}. \quad (10)$$

In the following section, it is shown how the measure of shift in partitions is applied to the corresponding sub-intervals according to their orientation, given by each symbolic state in the sequence starting from $s(x_0)$ to $s(x_n)$, so that the originating interval of the initial condition x_0 can sharply be narrowed down from the state space I.

VI. Initial Condition Estimation

When the map height is reduced, the magnitude of inequality of the resulting asymmetric sub-intervals depends on the measure of the $\mu < 1$. Also, while narrowing down to the originating interval, the orientation of the current sub-interval determines whether the bigger or the smaller sub-interval needs to be chosen for the next step. Therefore, a decision needs to be taken regarding the direction in which the partition of the current state needs to shift (from the midpoint of the previous sub-interval) for each symbolic iterate in question.

As can be seen from Fig. 6, the asymmetrical partitions generated by the reduced height map are shifted towards the critical point x_c. Here, we show how the orientation of the forthcoming $I_{\mathcal{S}_{i+1}}^i$ can be used to determine which direction the partition on $I_{\mathcal{S}_i}^{i-1}$ shifts to, and whether the bigger or the smaller sub-interval contains the originating interval of x_0. Using $\mathcal{S}_{n+1}$ = 01…$s(x_n)$ this can be illustrated in the following manner. For $x_0 \in I$, $s(x_0) = 0 \Rightarrow x_0 \in I_0^0$. $s(x_1) = 1 \Rightarrow x_1 \in I_1^0$ and therefore, $x_0 \in I_0^0 \cap T^{-1}(I_1^0) \equiv I_{01}^1$, which lies to the *right* of the newly

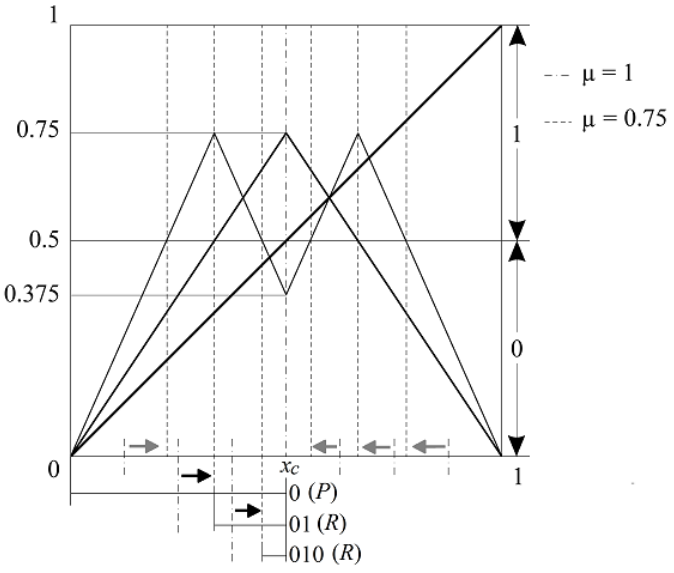


Fig. 6. The shift in the partition due to the reduced parameter is always towards the critical point x_c. The sub-interval I_{010}^1 has been narrowed down by the reduction in parameter from 1 to 0.75. The orientation of the interval is reversed by the 1 in the 2nd symbol, and the reversal is maintained by the 0 in the 3rd symbol.

generated partition as α_i is odd for I_{01}^1 (Fig. 6). Similarly, for $\mathcal{S}_{n+1} = 11\ldots s(x_n)$, despite $s(x_1) = 1$, α_i is even for I_{11}^1 and therefore $I_{11}^1 \ni x_0$ lies to the *left* of the newly generated partition. Continuing for n+1 symbols, the originating interval $I_{\mathcal{S}_{n+1}}^n \ni x_0$ can be obtained.

For the potential application of signal measurement from the symbolic sequence, the aforementioned approach has been formulated into a computational algorithm. The task of partitioning the intervals, and the subsequent choice of the sub-intervals has been adapted into a simplified numerical process for the ease of implementation in the digital processing domain. This is done by shifting one of the boundaries of the interval $I_{\mathcal{S}_i}^{i-1}$ towards the other, depending on the orientation of the resulting sub-interval $I_{\mathcal{S}_{i+1}}^i \ni x_0$, by a factor of μ, in such a way that the sub-interval that does not contain x_0 is eliminated, leaving the correct $I_{\mathcal{S}_{i+1}}^i$ behind thereby leading to the originating interval of x_0 on the n^{th} step.

For any given sequence $\mathcal{S}_{n+1}$, $s(x_0)$ is determined by $T^0(x_0)$ i.e. before any iteration through the map function, as the critical point x_c already divides the state space I into two equal halves. Hence the role of the first symbol $s(x_0)$ is simply to determine whether the algorithm must be performed on I_0^0 or I_1^0. Since I_1^0 is a mirror image of I_0^0 about $x_c = 0.5$, for any two symbolic sequences that differ only by their first symbol $s(x_0)$, their originating intervals also mirror each other exactly about x_c. Therefore, for simplicity, the computations for the symbolic

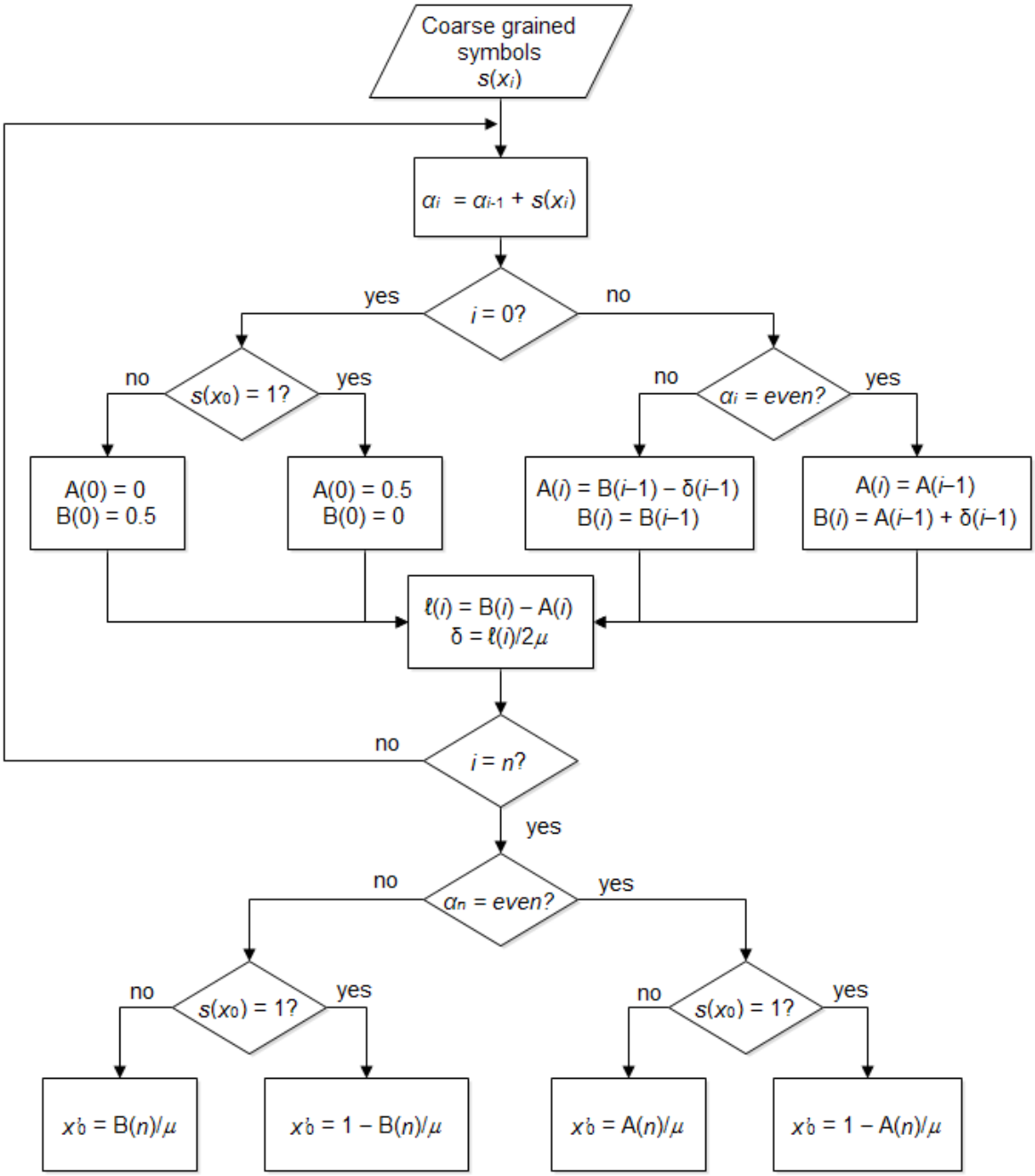


Fig. 7. The logical flow diagram of the proposed initial condition estimation (x'_0) algorithm from $\mathcal{S}_{n+1}$.

sequence beginning with $s(x_0) = 1$ is performed on the sub-interval that has a reverse orientation of $I_0^0 = [0,0.5]$.

From $s(x_1)$ onwards, let us denote the boundaries of the i–1^{th} sub-interval $I_{\mathcal{S}_i}^{i-1}$ as A(i–1) and B(i–1). Therefore, the corresponding length of the sub-interval is given by $\ell(i–1)$ = B(i–1) – A(i–1) and $\delta(i–1) = \ell(i–1)/2\mu$ determines by how much one boundary needs to be shifted towards the other for creating the i^{th} sub-interval. To evaluate the initial condition, the procedure is as follows:

1. For the interval I_0^0, i.e. for $T^0(x_0)$, the boundaries are referred to as A(0) = 0 and B(0) = 0.5 and $\ell(0)$ = B(0) – A(0) = 0.5 – 0 = 0.5. Similarly, by the previous proposition, the boundaries for I_1^0 is A(0) = 0.5 and B(0) = 0 and $\ell(0)$ = B(0) – A(0) = 0 – 0.5 = – 0.5. The negative value of the length is taken care of by the orientation of the symbols in the sequence.
2. From $s(x_1)$ onwards, the following step is repeated until $s(x_n)$. For i = 1, 2, …, n–1:
   - α_i is even, A(i) = A(i–1) and B(i) = A(i–1) + δ(i–1)
   - α_i is odd, A(i) = B(i–1) – δ(i–1) and B(i) = B(i–1)
3. When the operation is performed with μ < 1, the estimated initial condition x_0' is scaled by a factor of μ, resulting in $x_0' \in [0, \mu]$ which needs to be scaled back into $x_0' \in I = [0,1]$. Also, if $\mathcal{S}_{n+1}$ had $s(x_0) = 1$, the final sub-interval needs to be mirrored back into I_1^0 = [0.5,1]. Depending on the orientation of the sub-interval $I_{\mathcal{S}_{n+1}}^n$ of the n^{th} iteration, keeping the conditions in mind, we have four cases for determining the initial condition $x_0 \in I_{\mathcal{S}_{n+1}}^n$:
   - If α_n is even and $s(x_0) = 0$, $x_0' = A(n)/\mu$
   - If α_n is odd and $s(x_0) = 0$, $x_0' = B(n)/\mu$
   - If α_n is even and $s(x_0) = 1$, $x_0' = 1 – [A(n)/\mu]$
   - If α_n is odd and $s(x_0) = 1$, $x_0' = 1 – [B(n)/\mu]$

This has been summarised into a flow diagram in Fig. 7. The algorithm has been employed to confirm that the initial conditions originating from I for a range of control parameters is retrieved entirely as can be verified by the results in section VII.

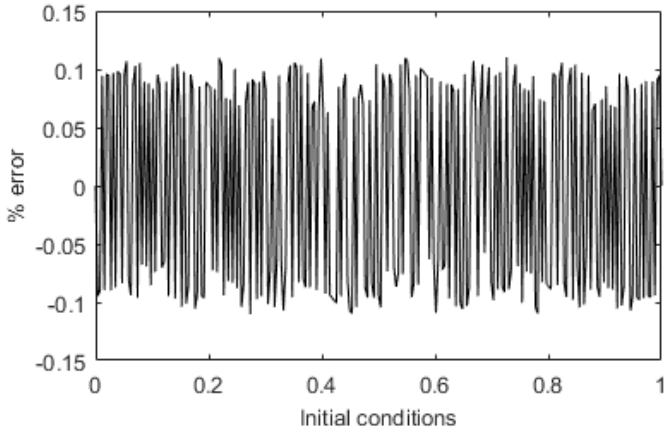
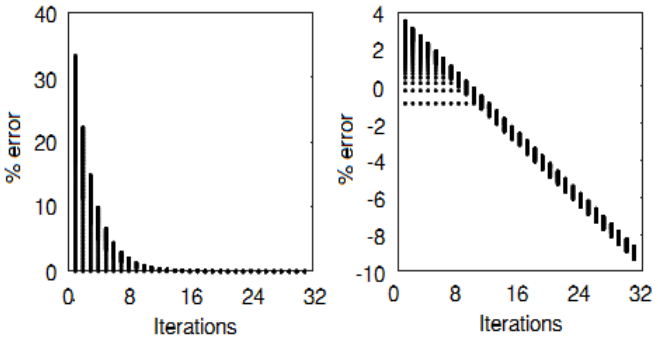


Fig. 8. The absolute error between the set of actual and the estimated initial conditions for μ = 0.75 using 16 symbols.

VII. Evaluation of the Algorithm



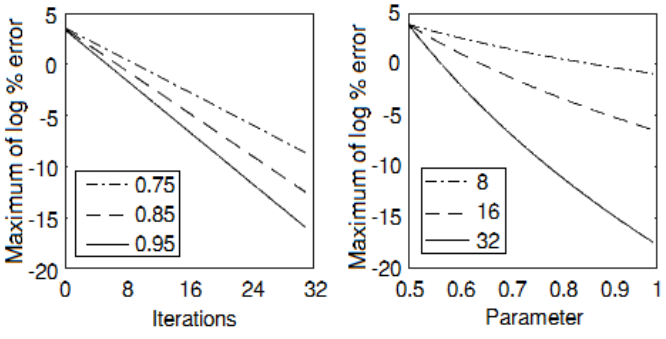
(a) (b)

Fig. 9. The (a) absolute error percentage $|\varepsilon_p|$ and (b) its $\log(|\varepsilon_p|)$ calculated for initial conditions with gradually increasing length of $\mathcal{S}$.

The functionality of the stated algorithm has been simulated and tested in a math processor (MatLab) with a set of initial conditions covering the entire state space I = [0,1]. A detailed insight regarding the behaviour of the algorithm in terms of number of symbols used, the resolution of the initial condition, and change in the parameter μ, are shown through various cases of test scenarios. The parameters for the tests are chosen in the range of μ = [0.75,1].

In order to generate the symbolic sequences for the tests, the selection of initial conditions from the state space I = [0,1] is done by dividing I into smaller fragments of size $1/2^\theta$ where $\theta \in \mathbb{N}_0$. The set of initial conditions $\mathcal{X}_\theta$ can be defined as $\mathcal{X}_\theta = \{x_0^p \mid x_0^p = p/2^\theta\} \subset \mathbb{R}$ where x_0^p is the p^{th} initial condition and index p = 0, 1, 2, …, 2^θ implying $\mathcal{X}_\theta$ contains 2^θ+1 elements. The set of initial conditions $\mathcal{X}_8$ is chosen for most of the cases which contains total of 257 real valued test points within I = [0,1].

The difference between the actual and the estimated initial condition is expressed as percent error $\varepsilon_p = [(x_0^p - \acute{x}_0^p)/I]\times 100 = 100(x_0^p - \acute{x}_0^p)$ where $x_0^p \in \mathcal{X}_\theta$ is the actual input initial condition and $\acute{x}_0^p$ is the corresponding estimated value of the initial condition form symbolic sequence. The elements of $\mathcal{X}_8$ are iterated to produce a symbolic sequence of length 16. The sequences are used to estimate the respective initial conditions



(a) (b)

Fig. 10. The change in logarithmic maximum percent error $\ln(|\varepsilon_{max}|)$ against (a) parameters 0.75, 0.85, 0.95 over a range of iterations (up to 32 symbols) and (b) iterations 8, 16, 32 over a range of parameters [0.75,1].

using the algorithm and each of the errors, ε_p is plotted against its corresponding x_0^p (Fig. 8).

To observe how the magnitude of error behaves when the number of symbols considered for estimation are increased, ε_p is calculated and plotted in Fig. 9 (a), at μ = 0.75 for all x_0^p, estimated for various length of symbols, where the length of symbols used for estimation was set each time from 1-32. The graph in Fig. 9 (a) shows an exponential drop in the maximum error $\varepsilon_{max} = max(|\varepsilon_p|)$ with the increase in number of iterations. To better observe the reduction in error over the count of symbols used, the logarithmic value of each $|\varepsilon_p|$, is plotted in (Fig. 9 (b)).

Predictably, the ε_p also depend on the parameter acting on the map during the generation of the symbolic sequences. The logarithmic ε_{max} of the state-space has been plotted and compared (in Fig. 10 (a)) for three different cases of parameters μ = 0.75, μ = 0.85 and μ = 0.95 across the number of symbols used. It can be seen that the maximum error decreases with increase in parameter value. If the number of iterations is fixed, then ε_{max} over a range of parameter [0.5,1], has a decreasing tendency. Three cases, with iterations 8, 16 and 32, is shown in Fig. 10 (b). The rate of decrement of ε_{max} increases with an increase in the number of symbol chosen. Thus, it is easily observed from the error trends that an optimum choice of iterations for an expected range parameter value can lead to a feasible system.

The accuracy of the estimated initial condition also depends on how fine the test points were chosen as a set of inputs in the first place. Performance of the algorithm is tested for nine sets of initial conditions with different resolutions, from $\mathcal{X}_8$–$\mathcal{X}_{16}$. Fig. 11 (a and b) shows the ε_{max} over the range of resolution for two cases of parameter values, μ = 0.75 and μ = 0.95, calculated using 16 symbols. The ε_{max} shows a marked increase for the resolutions higher than $\mathcal{X}_8$. In Fig. 11 (c), ε_{max} for μ = 0.85 is also plotted, which shows similar trend. Using 32 symbols for calculating ε_{max} with μ = 0.85, as in Fig. 11 (d) however, shows that the accuracy improves for higher resolutions with increment in number of symbols considered.

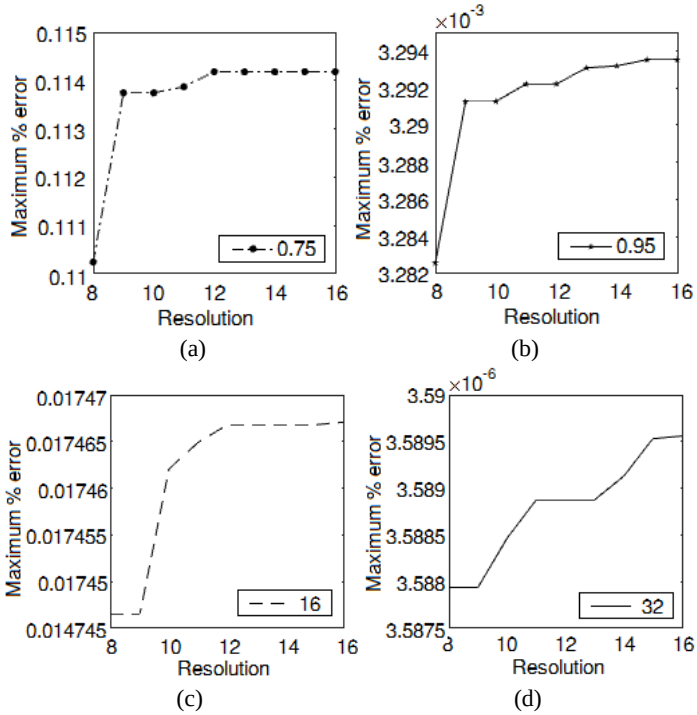


Fig. 11. The increase in ε_{max} over $\mathcal{X}_8$–$\mathcal{X}_{16}$ for (a) μ = 0.75 and (b) μ = 0.95. There is a marked increase in ε_{max} after $\mathcal{X}_8$ (symbols used: 16). The accuracy improves with increasing number of symbols, as seen in the plots generated for (c) 16 and (d) 32 symbols used for the estimation (μ = 0.85).

For any application, since it is expected that the number of iterations to be performed needs to be fixed, while choosing it, we must consider relating the error to the range within which the parameter of the implemented tent map circuit is likely to vary as well as the maximum resolution desired for the converted outcome. In the practical implementation, for an expected range of parameters and a fixed resolution, the choice of the number of iterations is directly related to the desired level of accuracy of the estimated outcome. The observations described until now can be utilised for software based applications. However, in a hardware oriented physical system, such fine structuring may not be observed because of the noise in the map circuit. The ε_{max} might measure higher due to the signal offsets depending on the component specifications.

VIII. Hardware Implementation

To evaluate the proposed algorithm on a physical system, we have implemented a tent map hardware adapted from the circuit implementation described in [14] (Fig. 12). A single tent map unit is used for iterations with the help of a two-stage sample-and-hold (S/H) feedback loop with a comparator in the input stage to generate the symbol for each iteration. The S/H loop is driven with anti-phase clocks using a microcontroller. A ramp function of 0–1 V is chosen as the set of input points to the circuit. We are taking symbols up to 16 bits for the estimation of each initial input point and operated the algorithm on the collected symbols.

In order to determine the map parameter, an input signal of 500 mV is separately iterated through the map up to 16 bits and parameter has been estimated to be approximately 0.90502 using the algorithm given in IV.

The 0–1 V ramp is input to the circuit and the estimation algorithm is executed using both 8 and 16 bit long sequences.

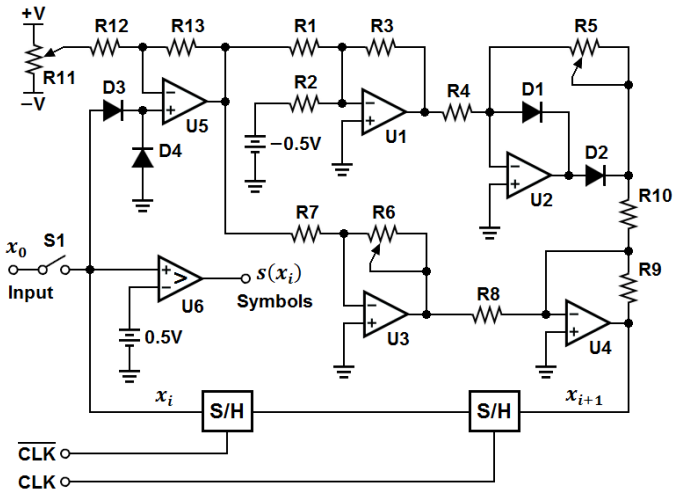


Fig. 12. Hardware implementation of tent map, as adapted from the circuit given in [14].

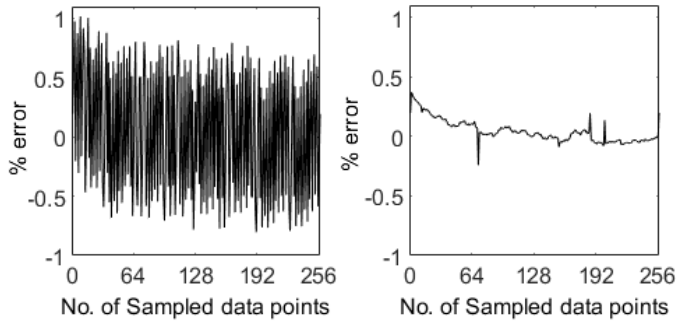


Fig. 13. The difference between the estimated initial conditions and the actual initial conditions are shown for (a) 8 and (b) 16 symbols.

The difference (ε_p) between the algorithm estimated outcomes and the actual input signals are calculated and plotted for 8 and 16 symbols. The results are shown in Fig. 13 (a and b).

To understand the errors better, the presence of noise in the physical system and the affected behaviour of the symbolic dynamics can be observed in further detail. When noise in the circuit gets multiplied over the iterations the signatures are affected causing the symbols to constantly flip about the critical point x_c. A range of 0−40dB signal-to-noise-ratio (SNR) is studied to see how noise affect the symbolic dynamics. For simulation, noise is added to every iterate of an orbit $\mathcal{O}_T$ using additive white Gaussian Noise (awgn(.)) function. The normalised mean of the number of symbols that flipped per iteration per decade change in noise in the entire state space I for $\mathcal{X}_8$ is shown in Fig. 14. It can be noticed that almost 50% of the orbits in $\mathcal{X}_8$ experience bit flipping due to noise, with gradually decreasing trend as the SNR improves. However, length of the noise free sequences can still be improvised with the aid of suitable filtering algorithms [18,19], which, at the moment, is beyond the scope of this article.

IX. Conclusion

In this paper, we have proposed a method of analogue-to-digital conversion using chaotic dynamics. In this approach, the signal to be measured is fed as initial condition to a unimodal chaotic map, and the resulting dynamical evolution contains the information about the initial condition. This is extracted using the symbolic sequence generated by the dynamics. So far, the main problem in practical adoption of this approach was that non-idealities in the map function in any practical implementation would make the measurement erroneous. In this paper, we have reported an algorithmic procedure to solve the critical problems posed by non-idealities of the map function used to generate the symbolic sequence.

We have shown that the information regarding the initial condition can still be recovered through a suitable interval arithmetic that employs non-uniform partitioning. The algorithm computes the shift in the partitions based on the actual value of the parameter of the map. Intervals visited by the dynamics could be traced back across the symbolic sequence and initial condition could be estimated with a reasonable accuracy. Along the way we had to solve the problem posed by the fact that while tracing back one has to apply the inverse of the map, but for each value two possible inverses exist. Since the parameter of the map is likely to fluctuate in a practical implementation, our algorithm also estimates the map parameter from the symbolic sequence and then uses that to compute the shift in the partition boundaries.

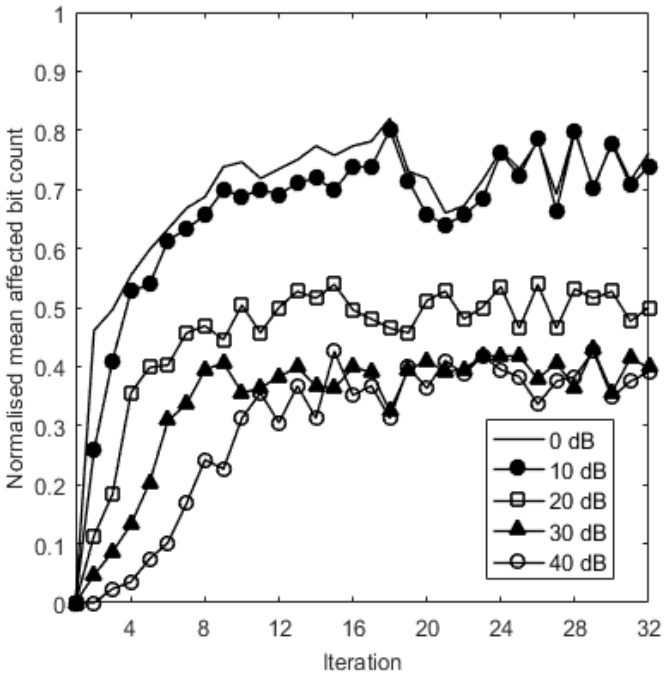


Fig. 14. The number of symbols flipped by the noise for every iteration over the state space I is counted and normalised (by dividing the count by the number of initial conditions considered in $\mathcal{X}_8$). The graph shows the normalized mean count of the number of bits flipped across I for every length of symbolic sequence for a range of SNR 0-40dB.

The accuracy of measurement was broadly studied through both simulation and hardware implementation. The precision is found to be dependent on the map parameter and number of iterations considered. There is an optimum range for the number of iterations if one knows the range within which the map parameter is expected to vary. For the conditions prevailing in practical implementations, a symbol length of 16 proved to be adequate for an effective estimation. Although noise in the system reduces a substantial number of meaningful symbols that can be accessed for processing, appropriate treatment for the noise can be incorporated to improve the quality of the symbolic trajectory. Performance can be further improved through optimization of the resource consumption, which offers scope of future work in this direction.

In practical terms, signal measurement using chaotic maps will have several benefits over the conventional ADC techniques in terms of resource consumption. Given the present-day ADC techniques, a large number of comparators are involved corresponding to the bit precision requirement and the structure gets doubled each time a bit is increased, as seen in the flash ADCs. Also, other techniques make use of additional digital to analogue converters (DACs) to compare the output of each stage of conversion and feedback the difference for subsequent stages as seen in sigma-delta, pipelined and successive-approximation ADCs. Chaotic measurement approach however saves resources considerably as it uses the same structure iteratively and bits are generated about the map threshold involving a single comparator.

REFERENCES

[1] R. H. Walden, "Analog-to-digital converter survey and analysis," in *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 4, pp. 539–550, Apr 1999. doi: 10.1109/49.761034.

[2] M. Robert, Y. Savaria and C. Wang, "Analysis of metrics used to compare analog-to-digital converters," *The 2nd Annual IEEE Northeast Workshop on Circuits and Systems, 2004. NEWCAS 2004.*, 2004, pp. 301–304. doi: 10.1109/NEWCAS.2004.1359091.

[3] S. Bashir, S. Ali, S. Ahmed and V. Kakkar, "Analog-to-digital converters: A comparative study and performance analysis," *2016 International Conference on Computing, Communication and Automation (ICCCA)*, Noida, 2016, pp. 999–1001. doi: 10.1109/CCAA.2016.7813861.

[4] V. Litovski, M. Andrejevic and M. Nikolic, "Chaos Based Analog-to-digital Conversion of Small Signals," *2006 8th Seminar on Neural Network Applications in Electrical Engineering*, Belgrade, Serbia & Montenegro, 2006, pp. 173–176. doi: 10.1109/NEUREL.2006.341205.

[5] C. Xi, G. Yong and Y. Yuan, "A Novel Method for the Initial-Condition Estimation of a Tent Map," *Chinese Physics Letters*, vol. 26, no. 7, pp. 078202 - 1–3, 2009. doi: 10.1088/0256-307X/26/7/078202.

[6] T. Kapitaniak, K. Zyczkowski, U. Feudel and C. Grebogi, "Analog to digital conversion in physical measurements," *Chaos, Solitons & Fractals*, vol. 11, no. 8, pp. 1247–1251, 2000. doi: 10.1016/S0960-0779(99)00003-X.

[7] N. Metropolis, M. L. Stein, P. R. Stein, "On finite limit sets for transformations on the unit interval," *Journal of Combinatorial Theory*, series A, vol. 15, no. 1, pp. 25–44, 1973. doi: 10.1016/0097-3165(73)90033-2.

[8] L. Wang and N. D. Kazarinoff, "On the Universal Sequence Generated by a Class of Unimodal Functions", *Journal of Combinatorial Theory*, series A, vol. 46, no. 1, pp. 39–49, 1987. doi: 10.1016/0097-3165(87)90075-6.

[9] R. Gilmore and M. Lefranc, "Discrete Dynamical Systems: Maps," *The Topology of Chaos*, 1st ed. New York, NY, USA: JW&Sons, 2002, pp. 40–53.

[10] J. M. Amigó, S. Elizalde, and M. B. Kennel, "Forbidden patterns and shift systems," *Journal of Combinatorial Theory*, series A, vol. 115, no. 3, pp. 485–504, 2008. doi: 10.1016/j.jcta.2007.07.004.

[11] E. M. Bollt, T. Standford, Y. Lai and K. Życzkowski, "What symbolic dynamics do we get with a misplaced partition? On the validity of threshold crossings analysis of chaotic time-series," *Physica D*, vol. 154, no. 3, pp. 259–286, 2001, doi: 10.1016/S0167-2789(01)00242-1.

[12] P. Collet and J. P. Eckmann, "Typical Behavior for One Map," *Iterated Maps on the Interval as Dynamical Systems*, 1st ed. Boston, MA, USA: Birkhäuser Basel, 1980, pp. 7–22.

[13] G. Alvarez and D. Arroyo, "Application of Gray codes to the study of the theory of symbolic dynamics of unimodal maps," *Communications in Nonlinear Science and Numerical Simulation*, vol. 19, no. 7, pp. 2345–2353, 2014. doi: 10.1016/j.cnsns.2013.11.005.

[14] I. Campos-Cantón, E. Campos-Cantón, J.S. Murguía and H.C. Rosu, "A simple electronic circuit realization of the tent map," *Chaos, Solitons & Fractals*, vol. 42, no. 1, pp. 12–16, 2009. doi: 10.1016/j.chaos.2008.10.037.

[15] E. Ott, "Strange attractors and chaotic motions of dynamical systems," *Reviews of Modern Physics*, vol. 53, no. 4, pp. 655–671, 1981.

[16] X. Wu, H. Hu, and B. Zhang, "Parameter estimation only from the symbolic sequences generated by chaos system," *Chaos, Solitons and Fractals*, vol. 22, no. 2, pp. 359–366, 2004. doi: 10.1016/j.chaos.2004.02.008

[17] D. Arroyo, G. Alvarez and J.M. Amigó, "Estimation of the control parameter from symbolic sequences: Unimodal maps with variable critical point," *Chaos*, vol. 19, no. 2, pp. 023125 - 1–9, 2009. doi: 10.1063/1.3155072.

[18] J. Schweizer and T. Schimming, "Symbolic dynamics for processing chaotic signals. I. Noise reduction of chaotic sequences," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 48, no. 11, pp. 1269–1282, Nov 2001. doi: 10.1109/81.964416.

[19] P. F. Marteau and H. D. I. Abarbanel, "Noise Reduction in Chaotic Time Series Using Scaled Probabilistic Methods," *Journal of Nonlinear Science*, vol. 1, no. 3, pp. 313–343, 1991. doi: 10.1007/BF01238817.

Rajlaxmi Basu received her B.Tech degree in Applied Electronics and Instrumentation Engineering from West Bengal University of Technology, India, in 2011 and M.Sc. in Embedded Systems from the University of Huddersfield, UK, in 2012. She is currently pursuing the Ph.D. degree from the same institute.

She has been teaching Electronics subject areas at the University of Huddersfield since 2014. Her research interests include mathematical modelling, statistical data analysis, algorithms for optimisation and control, signal processing and non-linear dynamical systems.

Ms. Basu was awarded the Vice Chancellor Scholarship by the University of Huddersfield to pursue her doctoral research.

Dhrubajyoti Dutta received his B.Tech degree in Instrumentation and Electronic Engineering from Biju Patnaik University of Technology, Orissa, India, in 2008 and the M.Sc. degree in Embedded Systems from the University of Huddersfield, UK, in 2012. He is currently pursuing his Ph.D. degree in electronics engineering from the same institute.

As an entrepreneur, he had developed instrumentation and control systems for biomedical, power and automation industries in collaboration with Jadavpur University, West Bengal, India, between 2008 and 2011. He has been teaching Electrical and Electronics subject areas at the University of

Huddersfield since 2014. His research interests include algorithms for optimisation and control, signal processing, mixed sensor applications, numerical modelling and data analysis, and estimation theories in non-linear dynamical systems.

Mr. Dutta had been awarded the Vice Chancellor Scholarship by the University of Huddersfield to pursue his doctoral research.

Soumitro Banerjee (born 1960) did his B.E. from the Bengal Engineering College (Calcutta University) in 1981, M.Tech. from IIT Delhi in 1983, and Ph.D. from the same Institute in 1987.

He was in the faculty of the Indian Institute of Technology, Kharagpur, since 1986, and has moved to the Indian Institute of Science Education & Research, Kolkata, in 2009. He has published three books: "Nonlinear Phenomena in Power Electronics" (Ed: Banerjee and Verghese, IEEE Press, 2001), "Dynamics for Engineers" (Wiley, London, 2005), and "Wind Electrical Systems" (Oxford University Press, New Delhi, 2005). His areas of interest are the nonlinear dynamics of power electronic circuits and systems, and bifurcation theory for non-smooth systems.

Dr. Banerjee served as Associate Editor of the IEEE Transactions on Circuits & Systems II (2003-05), and as Associate Editor of the IEEE Transactions on Circuits & Systems I (2006-2007). He is a recipient of the S. S. Bhatnagar Prize (2003), and was recognized as a "Highly Cited Author" by Thomson Reuters from 2004 to 2014. He is a Fellow of the Indian Academy of Sciences, the Indian National Academy of Engineering, the Indian National Science Academy, The World Academy of Sciences, and the IEEE.

Violeta Holmes obtained her B.Eng. in Physics and Electronics from the ETF Electronic and Technical Physics Faculty (University of Belgrade) in 1982, M.Sc. in Control Engineering from the University of Bradford in 1986, and Ph.D. from the University of Huddersfield in 1994.

She worked as a Systems Engineer for computer control of hot and cold steel rolling mills in Smederevo Steel Mill Yugoslavia. She has been serving as the faculty at the University of Huddersfield since 2007. In 2010, she co-founded the High Performance Computing (HPC) Research Group which has established links with the UK National e-Infrastructure Service (NES) as an affiliate member and has worked with 3M Business and Innovation Centre at the University of Huddersfield, fulfilling their needs of HPC.

Dr. Holmes was awarded the status of Chartered Engineer (IET) and Fellowship of Higher Education Academy (HEA) and is a member of the Institute of Engineering and Technology (IET) and the British Computer Society (BCS). She participated in The North West Universities Mentoring Scheme for women in Science, Engineering and Technology SET (MENWU) as a mentor in 2004/2005.

Peter Mather completed his BEng in Electrical and Electronics at the University of Huddersfield in 1990 and PhD in 1995.

After completing post-doctoral research into mixed signal test techniques, he was a senior mixed signal design engineer in the integrated circuit (IC) industry. He is currently a senior lecturer at the University of Huddersfield; current research interests are in mixed signal systems, primarily in the area of chaos based measurement; co-authoring a patent application filed in 2014, in the area of small signal measurement using chaos based systems.

Dr. Mather is a fellow of Higher Education Academy (HEA) and is a member of the Institute of Engineering and Technology (IET).